Bochumer Linguistische Arbeitsberichte 16

Proceedings of the 13th Conference on Natural Language Processing (KONVENS)
Bochum, Germany
September 19–21, 2016

Bochumer Linguistische Arbeitsberichte

Herausgeberin: Stefanie Dipper

Die online publizierte Reihe „Bochumer Linguistische Arbeitsberichte“ (BLA) gibt in unregelmäßigen Abständen Forschungsberichte, Abschluss- oder sonstige Arbeiten der Bochumer Linguistik heraus, die einfach und schnell der Öffentlichkeit zugänglich gemacht werden sollen. Sie können zu einem späteren Zeitpunkt an einem anderen Publikationsort erscheinen. Der thematische Schwerpunkt der Reihe liegt auf Arbeiten aus den Bereichen der Computerlinguistik, der allgemeinen und theoretischen Sprachwissenschaft und der Psycholinguistik.

The online publication series “Bochumer Linguistische Arbeitsberichte” (BLA) releases at irregular intervals research reports, theses, and various other academic works from the Bochum Linguistics Department, which are to be made easily and promptly available for the public. At a later stage, they can also be published by other publishing companies. The thematic focus of the series lies on works from the fields of computational linguistics, general and theoretical linguistics, and psycholinguistics.

Band 16 (September 2016)

Herausgeberin: Stefanie Dipper
Sprachwissenschaftliches Institut
Ruhr-Universität Bochum
Universitätsstr. 150
44801 Bochum

Erscheinungsjahr 2016
ISSN **2190-0949**

Stefanie Dipper, Friedrich Neubarth
and Heike Zinsmeister (Eds.)

Proceedings of the 13th Conference on Natural Language Processing (KONVENS) Bochum, Germany September 19–21, 2016

2016

Bochumer Linguistische Arbeitsberichte

(BLA 16)

Contents

Preface

This volume contains the proceedings of the 13th KONVENS (*Konferenz zur Verarbeitung natürlicher Sprache*), which is organized by DGfS-CL and hosted by the Linguistics Department at the Ruhr-University Bochum. The conference takes place September 19–21, 2016 in Bochum, Germany.

KONVENS has been held biennially since 1992 and is organized in turn by the scientific societies DGfS-CL (German Society for Linguistics, SIG Computational Linguistics), GSCL (Society for Language Technology and Computational Linguistics) and ÖGAI (Austrian Society for Artificial Intelligence).

This year's special theme is "Processing non-standard data – commonalities and differences". A wide range of data can be considered "non-standard" because it deviates in one way or the other from standard written data such as newspaper texts. Examples include data produced by language learners, historical data, dialect data, data from social media, or (transcriptions of) spoken data. We especially encouraged the submission of contributions comparing different types of non-standard data and their properties, focussing on their impact for natural language processing. For example, a feature common to many types of non-standard data is the use of non-standard spelling. However, spelling variation in learner data as compared to historical data is due to very different reasons and thus most likely results in very different types of non-standard spellings.

We are very proud to have two invited speakers who address the special theme in particular: John Nerbonne (University of Groningen & Universität Freiburg), who will give a talk on "Data from Non-standard Varieties", and Barbara Plank (University of Groningen) on "What to do about non-standard (or *non-canonical*) language in NLP". They both also contributed papers to this volume.

In total we received 58 submissions from authors from 14 different countries. 38 papers were accepted for either oral or poster presentations. The papers represent a broad selection of recent research in natural language processing, addressing topics ranging from measuring inter-annotator agreement to spelling normalization and sense induction.

The program also includes the ceremony for the GSCL doctoral thesis award in memory of Wolfgang Hoeppner and a talk by this year's recipient.

KONVENS 2016 is accompanied by several workshops and tutorials on September 22, 2016:

- NLP4CMC III: 3rd Workshop on Natural Language Processing for Computer-Mediated Communication, organized by Michael Beißwenger and Torsten Zesch (both Duisburg-Essen)
- IGGSA Workshop on German Sentiment Analysis 2016 (tutorial and shared task), organized by Melanie Siegel (Darmstadt, tutorial), Josef Ruppenhofer (Hildesheim, shared task), Julia Maria Struß (Hildesheim, shared task), Michael Wiegand (Saarbrücken, shared task)
- Visual Analytics for Computational Linguistics (tutorial), organized by Annette Hautli-Janisz and Dominik Sacha (both Konstanz)
- Symbolic Distributional Semantics with the JoBimText Framework (tutorial), organized by Martin Riedl, Eugen Ruppert and Chris Biemann (all Darmstadt)

We would like to thank DGfS-CL for supporting the conference and all participants for making this a great conference. We would also like to thank the members of the program committee for providing detailed reviews in due time and the local organizers. Finally, we are very grateful to our two invited speakers, John Nerbonne and Barbara Plank, for accepting our invitation.

These proceedings are available electronically at
https://www.linguistics.ruhr-uni-bochum.de/bla/.

Bochum, September 2016
The editors

Chairs

Stefanie Dipper (DGfS-CL/Ruhr-Universität Bochum, Germany)
Friedrich Neubarth (ÖGAI/OFAI Wien, Austria)
Heike Zinsmeister (GSCL/Universität Hamburg, Germany)

Local Organizers

Stefanie Dipper
Marcel Bollmann
Katharina Bort

Jessica Ernst
Julia Krasselt
Maurice Langner
Florian Petran
Claudia Roch
Adam Roussel
Fabian Simonjetz

Program Committee

Adrien Barbaresi (BBAW/ÖAW)
Chris Biemann (Technische Universität Darmstadt)
Felix Bildhauer (Freie Universität Berlin)
Andre Blessing (Universität Stuttgart)
Fabienne Braune (Universität Stuttgart)
Ernst Buchberger (ÖGAI/Medizinische Universität Wien)
Stephan Busemann (DFKI, Saarbrücken)
Miriam Butt (Universität Konstanz)
Özlem Çetinoğlu (Universität Stuttgart)
Simon Clematide (Universität Zürich)
Berthold Crysmann (CNRS – Laboratoire de Linguistique Formelle)
Markus Dickinson (Indiana University)
Martin Forst (NetBase Solutions GmbH)
Anette Frank (Universität Heidelberg)
Erhard Hinrichs (Universität Tübingen)
Manfred Klenner (Universität Zürich)
Roman Klinger (Universität Stuttgart)
Valia Kordoni (Humboldt-Universität zu Berlin)
Brigitte Krenn (OFAI, Wien)
Udo Kruschwitz (University of Essex)
Sandra Kübler (Indiana University)
Ekaterina Lapshinova-Koltunski (Universität des Saarlandes)
Florian Laws (Universität Stuttgart)
Anke Lüdeling (Humboldt-Universität zu Berlin)
Alexander Mehler (GSCL/Goethe-Universität Frankfurt)
Wolfgang Menzel (Universität Hamburg)
Detmar Meurers (Universität Tübingen)
Preslav Nakov (Qatar Computing Research Institute)
Günter Neumann (DFKI, Berlin/Saarbrücken)
Sebastian Padó (Universität Stuttgart)

Alexis Palmer (Universität Heidelberg)
Johann Petrak (University of Sheffield)
Manfred Pinkal (Universität des Saarlandes)
Hannes Pirker (ACDH, ÖAW)
Barbara Plank (University of Groningen)
Simone Ponzetto (Universität Mannheim)
Michael Pucher (ARI, ÖAW)
Uwe Quasthoff (Universität Leipzig)
Ines Rehbein (Universität des Saarlandes)
Georg Rehm (DFKI, Berlin)
Josef Ruppenhofer (Universität Hildesheim)
Felix Sasaki (DFKI, Berlin)
Dietmar Schabus (OFAI, Wien)
Roland Schäfer (Freie Universität Berlin)
Yves Scherrer (Université de Genève)
Helmut Schmid (Ludwig-Maximilians-Universität München)
Thomas Schmidt (IDS, Mannheim)
Sabine Schulte Im Walde (Universität Stuttgart)
Wolfgang Seeker (Universität Stuttgart)
Rico Sennrich (University of Edinburgh)
Marcin Skowron (OFAI, Wien)
Caroline Sporleder (Universität Göttingen)
Manfred Stede (Universität Potsdam)
Angelika Storrer (Universität Mannheim)
Olga Uryupina (University of Trento)
Tim vor der Brück (Hochschule Luzern)
Thomas Weskott (Universität Göttingen)
Ernesto William de Luca (Georg-Eckert-Institut, Braunschweig)
Andreas Witt (IDS, Mannheim)
Magdalena Wolska (Universität Tübingen)
Feiyu Xu (DFKI, Berlin/Saarbrücken)
Amir Zeldes (Georgetown University)
Torsten Zesch (Universität Duisburg-Essen)
Desislava Zhekova (Ludwig-Maximilians-Universität München)

Data from Non-standard Varieties

John Nerbonne
Dept. Informatiekunde
Rijksuniversiteit Groningen
&
Germanistische Linguistik
Albert-Ludwigs-Universität Freiburg
j.nerbonne@rug.nl

Abstract

The most important reasons for examining "non-standard data" with CL methods are the facts that this data represents a great deal of language behavior and that it serves as an object of scientific study in linguistics as a whole. This is true of the syntax of non-native second-language learners, the accents of non-native speakers, and the vocabularies of different dialect speakers.

Computational linguists have a good deal to offer to the various subfields of linguistics studying non-standard data. By automating steps in analysis we make the analyses replicable and also modifiable, we improve opportunities for comparing similar analyses, and perhaps most importantly, we enable the analyses of large amounts of data, providing more comprehensive views.

The data itself can be tricky to work with, however, as scientists in other fields are often specialized in a single language or language pair, which means that their data will not be varied enough to support all the research questions one would like to ask, e.g., the question of the generality of the techniques for a particular purpose. In other cases, the data simply won't have been collected with an eye to answering some interesting questions, which may mean that important parameters haven't been recorded. Finally, we note that non-automated analyses do not impose expectations that data be commensurate to the same strict degree (as automated ones), meaning that surprises can be in store even in well-studied data sets. This paper provides some concrete examples and discussion of these potential pitfalls.

One can protect oneself from some of these risks by seeking collaboration with domain experts, which is to be recommended in any case, as a way of making the work richer and better informed. Further, it makes sense to approach novels sorts of data — and even novel sources of data of a sort one suspects is familiar — with a broad range of potential research questions. There is an awful lot of interesting work still to be done!

1 Introduction

The theme of this year's KONVENS is non-standard data, and it's a great choice as computational linguistics (CL) ventures into areas of linguistics it's traditionally shied away from! I interpret the shyness incidentally not as a lack of CL interest in areas such as spoken language, historical data, second language learning, etc. (topics mentioned in the call for papers), nor as disregard for non-standard varieties, and certainly not as indifference to unedited prose in general, but rather as a wish to concentrate on honing technique and a wish to obtain results that allow interpretation with a focus on technique. From those points of view it makes sense to limit other parameters from varying too much. But I also agree emphatically that the time is ripe to widen CL's purview to include language from these other areas.

There are several reasons why we as computational linguists should work more with less standard data. First, most language is produced without any editing, and therefore without any effort to put it into a standard form. If we're going to deal with language of a wide variety of sorts, it will be difficult to avoid non-standard data. Second, there are important contributions CL is poised to make and which are simply required to make progress in this area. Some recent papers that illustrate this are Eisenstein, O'Connor, Smith, and Xing (2014) and Jurafsky, Chahuneau, Routledge, and Smith

(2014). Nguyen, Doğruöz, Rosé, and de Jong (Accepted to appear) provides a survey of CL work related to sociolinguistics, a large part of which involves the analysis of social media, usually in rather spontaneous, i.e. non-standard form.

I want to contribute to this theme by relating some of my experience with working with non-standard data, in particular data from non-standard varieties — dialects, "regiolects" (intermediate between dialects and stand languages, see Auer and Hinskens (1996)), and language in situations where there is contact (second language learner situations). After relating some of this experience, I'll close with some reflection on these. My intent is to be encouraging, but I'll note pitfalls as well as opportunities. There's every reason to be keen, but also to be cautious.

2 Contact syntax, aggregate distance, and detecting differences

Some colleagues in Linguistics at the University of Oulu were eager to collaborate on the *Finnish Australian English Corpus* (Watson, 1996). They'd worked on the corpus before, but never using language technology. The corpus consists of transcriptions of conversations held with Finnish emigrants to Australia. The 60-member group we'll focus on were adults on emigrating in the 1960s (around 30 yr. old), and they were interviewed after 30 years in Australia. They had working-class backgrounds, and "very few could speak any English at all on arrival to Australia" (Watson, 1996, p.45). The English was indeed quite rough, as expected, and this of course posed the technical challenge. To get a flavor of this consider the following excerpt from the corpus, elicited by asking participants to describe what a soldier would need to do to complete an assault course they were shown in a sketch:

> The soldier first have to go climb, climb to tree. Then uh, I don't know how they call that but uh, I, I call um, walkin' by hands, hangin' by hands or walkin' hands to other tree, come down to ground, walkin', um, uh, not walkin' but climbin' over brick wall, come dine..., do..., down other side, then have to go to ground by knees, goin' under some or, or whatever it is, climbin' up by ladders to other bick..., brick wall and jump down to ground on other side + um, there is, then have to go tunnel, maked from brick, come out on other end and ju..., jump to river, swim cross to finish line.

2.1 Theoretical goals

There is and was scientific consensus that one should expected to find Finnish-like elements in the speech of these emigrants (Opas-Hänninen, Hirvonen, Juuso, & Lauttamus, 2005), but I felt especially challenged first by a remark by the great theoretician on language contact, Uriel Weinreich:

> No easy way of measuring or characterizing the total impact of one language on another in the speech of bilinguals has been, or probably can be devised. The only possible procedure is to describe the various forms of interference and to tabulate their frequency (Weinreich, 1968, p.63)

Second, Ellis (1994), De Bot, Lowie, and Verspoor (2005) and other theorists of second-language acquisition have emphasized that it is not enough to catalog the "errors" of second-language users, because non-native speech often differs not in errors, but rather by overusing and by underusing specific linguistic items, where easier elements are typically overused and tougher ones underused. So the initial goal was to develop an aggregate measure of syntactic difference that was sensitive to overuse and underuse.

We settled on looking at part-of-speech (POS) tags, focusing on the frequency distributions of trigrams of POS-tags. We deliberately did not include lexical information in order to focus on syntax, and we decided to use trigrams in order to make the measure sensitive to context. Of course we were aware that looking at ordered sequences of syntactic categories might not be a general solution, but we were looking at English, where order is quite important, and we were interested in language behavior, not language competence.[1] By examining frequencies, we automatically gauged overuse and underuse, and by examining the entire distribution of POS-tag sequences we could claim to be contributing to Weinreich's goal of providing information on the total impact of the first language on the second (albeit only for syntax). We set our initial goals quite high.

[1] Sanders (2007) extended the work under description by examining leaf-path ancestors in parse trees.

2.2 Results

But would it work technically? We trained Thorsten Brant's TnT tagger (Brants, 2000) on the British part of the ICE corpus using the 270-element TOSCA-ICE tag set (Nelson, Wallis, & Aarts, 2002). We were naturally concerned with tagging accuracy, so we manually evaluated tagging accuracy on 1,000 randomly chosen words. The tagger was correct 81.2% of the time for single tags dropping to 56.1% for trigrams (Brants's tagger is 96.7% accurate when applied to the Penn Treebank) . See Wiersma, Nerbonne, and Lauttamus (2011). We also experimented with a smaller tag set, which naturally improved performance, but we decided to use the larger set for its more sensitive reflection of syntax.

We also tagged a corpus of speakers who had emigrated at 17 years of age and younger, because the material was most commensurable to the transcripts of the older emigrants. The speech of the younger emigrants was native-like, and we used this to identify particularly deviant POS trigrams.

We ignored very infrequent POS-trigrams (nearly 40,000 trigrams with frequency less than five in either corpus) so as not to be misled by what might be coincidence, and then compared the relative frequencies in the two corpora under comparison. Wherever the relative frequencies differ a good deal, we suspect that we are seeing contact-induced effects.

With respect to developing a syntactic distance measure, a rigorous validation would have to compare several data sets, ideally involving different target and different source languages, as well as several degrees of non-nativelike syntactic behavior.[2] So the best we can say on this score is that we've introduced a technique, but not that we've shown it to be probative, and certainly not for a range of languages and different degrees of contact-induced "contamination".

Close collaboration with the domain expert, Timo Lauttamus, was absolutely essential in applying this work to the question of detecting differences automatically. He examined a random sample of 137 of the 300 most divergent POS-trigrams and showed that most of them — all but 24 — were interpretable as the result of second-language disfluencies and a Finnish "substrate" in the emigrants' English. These include problems with (in)definite articles (Finnish has none), with the copula *be* (missing in Finnish), with the expletive *there* (likewise missing), and difficulties with contractions and auxiliary verb sequences, both of which led to underuse. See Lauttamus, Nerbonne, and Wiersma (2007) for a detailed presentation. From the point of view of identifying deviant syntax, the work was successful.

[2]For the sake of completeness I'll add that we *could* test for overall differences — in line with Weinreich's goal — by applying a permutation test to the table with two varieties and 8,300 instantiated POS-trigrams. This involved a tricky normalization. Di Buccio, Nunzio, and Silvello (2014) have suggested using vector space techniques to compare the trigram frequencies, and this seems more straightforward.

2.3 Reflection

It should be clear that we cannot claim to have solved the problem of measuring overall syntactic differences in varieties. We developed a measure and showed that it could be put to good exploratory use, but we certainly do not claim to have validated it rigorously. We just showed that the software helped in looking for differences in language use — in spite of the fact that the data was certainly noisy and the computational tool suboptimal.

Collaboration with Lauttamus, an expert on the English Finns acquire naturally, was essential to the success of the project, as was the fact that we eschewed a narrow focus on developing a measure of aggregate syntactic difference. I think we pushed the envelope a bit on that score, but, as I've emphasized, it would be rash to claim success on that point. It was essential that we aimed rather broadly in dealing with this data.

3 Accents and a caution on theory

There is a well-established line of research in which CL techniques are applied in dialectology, and Nerbonne (2009) motivates this theoretically. For the most part, this line of research has applies edit-distance measures (Kruskal, 1983) to phonetic transcriptions, and the work has established itself at least in areas where transcriptions are the primary recordings of pronunciations (most data collections more than about fifty years old). We have experimented with various modifications to the basic edit distance algorithm, where Heeringa, Kleiweg, Gooskens, and Nerbonne (2006) gives a flavor of the range of these modifications we've experimented with. Currently we prefer a version in which segment distances are weighted by the (inverse) frequency of their chance of corresponding in alignments. Because we gauge this frequency in-

formation theoretically, using pointwise mutual information, we refer to this as PMI-LEVENSHTEIN (Wieling, Margaretha, & Nerbonne, 2012).

But it has always been clear that reliable procedures for assaying the degree of difference in pronunciation would be useful for other reasons. Kondrak and Dorr (2006) demonstrate that this sort of procedure, applied to candidate drug names against the background of a data set of existing names, can identify potentially confusing name candidates, a circumstance that has been shown to have occasionally fatal consequences. In information retrieval, it is often difficult to find references to people whose names are normally spelled in a different writing system (Nabende, Tiedemann, & Nerbonne, 2010), such as Cyrillic, Arabic, Urdu or Japanese. One example of such a name is 'Musharraf', which sometimes occurs as 'Musharrav', 'Musharaf', etc. While there are often established conventions for TRANSLITERATION, few writers obey these, so a common technique is to attempt to identify alternative spellings that are likely to have the same pronunciation. This makes procedures for measure pronunciation differences useful in this context as well. Yet another, third area of application is in the diagnosis of speech problems, and Sanders and Chin (2009) have indeed applied an edit distance measure of the speech of cochlear implant bearers with some success.

3.1 Accents

It had occurred to me and to others that measuring how strong foreign accents are might be a fourth area of linguistics where a measure of pronunciation differences might be of interest, in particular to researchers in second-language learning. So I was very pleased when one of my collaborators, Martijn Wieling, noticed the Speech Accent Archive at George Mason University (Weinberger & Kunath, 2011). It contained then the recordings of over 800 non-native speakers of American English together with their phonetic transcriptions. By organizing a web-based judgment task[3] we were able to validate the PMI-based edit distance for this slightly different task — that of judging how non-native a speech sample sounded. The computational measures correlate very strongly ($r = 0.81$) with the judgment of native speakers with respect to how native-like the recorded passages were (Wieling, Bloem, Mignella, Timmermeister, & Nerbonne, 2014).[4] So the effort of moving into a new field led to a valuable new validation of technique, and this is worthwhile!

We were also able to explore the scientific issues a bit, investigating factors influencing the quality of the non-native accent, both the age at which English was learned and also the number of years resident in an English speaking country. These "insights" are almost proverbial — amounting to "the early bird catches the worm" and "practice make perfect", so we certainly don't claim any scientific breakthrough here, but our sample was large enough to let us catch a non-linear interaction between the two sorts of influences. As Figure 1 demonstrates, the two factors interact in a complex way. The "contour lines" on the regression surface are not evenly spaced as one moves up the age of learning onset; instead, lines are further and further away from one another, meaning that it becomes harder and harder to compensate for a late start with a long residence. I think we are the first to show this (Wieling, Bloem, Baayen, & Nerbonne, 2014).

3.2 Overreaching theoretically

So far, my report on this foray into a new sort of data makes it sound like an unqualified success, but there is more to tell. In a further step we tried to apply our "insights" to illuminate a famous issue in language acquisition and cognitive science, namely the CRITICAL AGE HYPOTHESIS. The idea is straightforward. We let the the distance of the learner's speech from the native pronunciation stand proxy for the success of language acquisition in general, and take the age of learning onset at face value. We can then plot the distance of the learner's speech from the native pronunciation as a function of the age of learning onset to get an idea if whether the decline in ultimate attainment is smooth, or whether there is a point — sometime before eighteen years or so — where ability sharply decreases. There's a nice paper by Jan Vanhove reminding us that PIECEWISE REGRESSION is the right technique to apply statistically (Vanhove, 2013), and Figure 2 shows the result of applying piecewise regression to the accent data.

Figure 2 breaks the data down into speakers of Indo-European languages (IE) and non-IE languages, which was not part of an initial hypothesis

[3] We're grateful to Mark Liberman for announcing this on *Language Log*, which is why so many subjects joined in.

[4] The native speaker judges only agreed with each other to a slightly greater degree ($r = 0.84$)

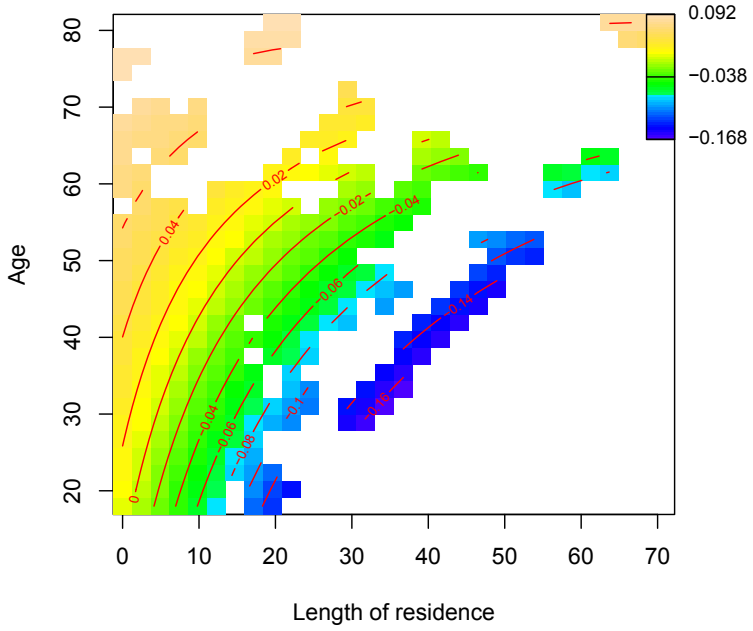

Figure 1: Accent quality is shown by the color, ranging from dark blue (quite native-like) to light yellow (distinctly foreign accent). Note that, in general, a long residence leads to better accents (darker blue) as does an early age at which English was learned. White areas indicate combinations of parameters for which little data was available.

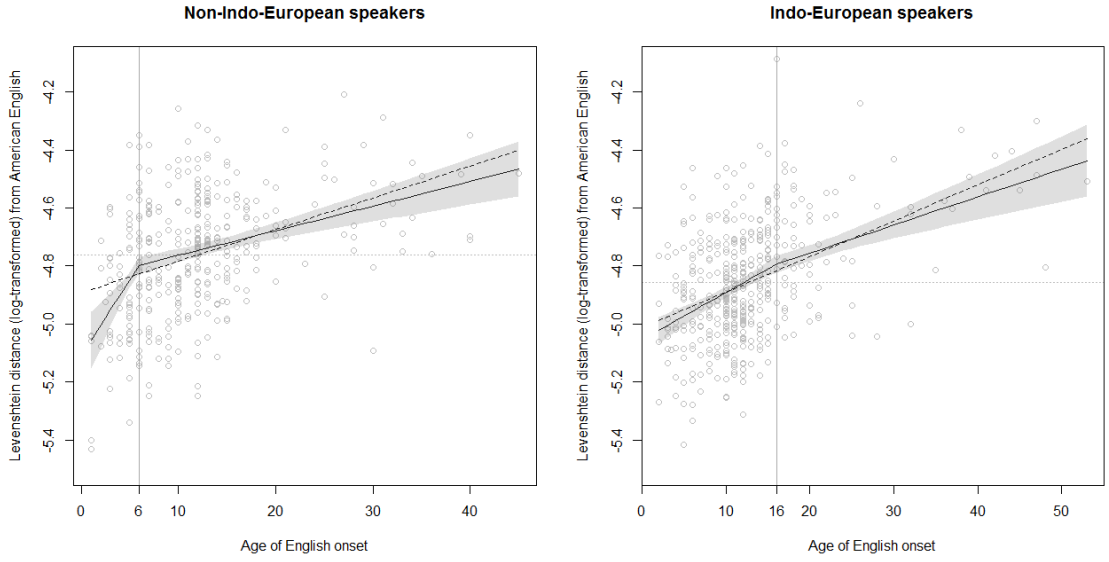

Figure 2: Accent quality (non-nativelikeness) deteriorates monotonically as a function of the age at which English was first learned. Moreover, there appears to be a sharp break around six years of age for speakers of non-IE languages (left), which would be compatible with the critical age hypothesis.

we brought to the data, so clearly **not** a hypothesis we might claim to confirm based on the analysis, but it's quite intriguing, as it suggests that the native language of learners might be a confound in studies of ultimate attainment of second-language learners. Speaker of languages related to English deteriorate throughout their lifetimes in their approximation to English pronunciation, but the deterioration is fairly constant. In contrast, the non-IE speaker's decline changes abruptly, even though, curiously, the rate of decline decreases after a critical point.

The referees varied in their reactions. One was definitely positive about the novelty of the finding and correctly chided us for failing to acknowledge potential biases in the data set (selection!), which the others likewise saw, but we were chastised to the point of emphatic rejection for not being *au courant* in the literature on second language learning vis-à-vis the critical age hypothesis. We'd read up on what we could, but there's an enormous literature, and it is difficult to get a sense of all that specialists hold dear. We definitely failed to distinguish IMPLICIT learners – those with no formal language training – from EXPLICIT learners, while the field has turned to seeing only implicit learners as interesting. In fact, however, there is no good way to operationalize this distinction in the Speech Accent Archive (Weinberger & Kunath, 2011) — those compiling the data set simply didn't include this information. So this aspect of the work simply failed to show what we originally claimed.

3.3 Reflection

Just as in the experiences with the non-native syntax, this line of research achieved some success, and for that it was again crucial that we had aimed broadly — both at validation of PMI-Levenshtein as a measure of pronunciation difference and at characterizing the role the age of learning onset and the length of residence plays. We failed to contribute to the discussion on the critical age for language acquisition due to our not knowing the literature sufficiently.

In retrospect we became convinced that there was no way to use the data to say much of anything about the critical age hypothesis, and it would have been prudent to seek collaboration with a language acquisition specialist before developing that aspect of the work. So the point for computational linguists interested in non-standard data is just that there is often a body of theory and and a literature that one simply has to command sufficiently in order to contribute.

4 Lexical variants and incommensurate data

Working with non-standard data entails surprises, at least occasionally. Working with standard data — say the Penn Treebank, the BNC or CELEX — means building on the work of others and (normally) relying on the intelligent choices of predecessors. Leaving this well-trodden path means that one occasionally has to think through the whole process of what it means to draw inferences with respect to a given hypothesis based on an unfamiliar data source. It often entails working with data that has previously only been analyzed manually and perhaps only examined for key features, so that there may be no experience of automatic processing, which means in turn that some problems — including missing data, confounds and unexpected distributions — may arise for the first time.

4.1 Dialect variation in vocabulary

It is interesting to examine the degree to which different linguistic levels correlate in their geographic distribution, e.g., pronunciation, lexical choice and syntax (Spruit, Heeringa, & Nerbonne, 2009), so we have looked at lexical and syntactic variation as well the pronunciation variation that we've mostly concentrated on. One such study involved the *Linguistic Atlas of Middle and South Atlantic States* (LAMSAS, Kretzschmar Jr. (1993)). In comparison to applying appropriate edit distance measures (to non-standard transcriptions), the vocabulary task sounded simple. Varieties should count as the same to the degree that they use the same words in response to fieldworkers' questions.

Looking at the data convinced us of two things, first, that simple string identity was likely to be too rough a measure to be useful. See Table 1, and see `http://www.let.rug.nl/~kleiweg/lamsas/overview/lex.txt` for a complete listing of all the responses in the data set.

Nerbonne and Kleiweg (2007) present a range of techniques that have been proposed for detecting similarity in lexical data, including approaches that use Porter stemming, edit distance, and inverse frequency weighting (Goebl, 1984) in (five) various combinations. This paper is written in the usual style of computational linguistics (CL), where several techniques are compared with respect to the

clearing up (435), clearing off, clearing, fairing off, clear up (50), fairing up, clear off, cleared up, fair off, clearing away (28), cleared off, breaking off, faired off, breaking away, fair up (18), break off, breaking, going to clear up, clear, fairing (9), ..., clouds is breaking (3), ..., ceasing, changing, fair, ..., held up, is broke, weather's going to break (2), a Dutchman's britches (1), ..., a-fairing, ..., a settling off, ..., blow off, blue sky enough to, ..., brightening, ..., make a Dutchman's pants, ..., moderating, ..., slacked up, ...

Table 1: Selection of responses to the question "If the sun comes out after a rain, you say the weather is doing what?" in decreasing order of frequency. 1516 response tokens, including 81 singletons (hapax legomena).

performance on an object measure. In the interest of space, I will not not repeat the presentation here, but I will note that it demonstrates the usefulness of CL techniques on non-standard data.

Second, we noted that field workers had often recorded multiple responses. Since this gives a flavor of working with non-standard data, I'll summarize the treatment here. For example, there were 1516 responses to the question of how to describe weather when rain was giving way to fairer skies — coming from only 1162 informants. Given that the data included multiple responses, we had to develop a generalization of the simple identity criterion for scoring responses. After all, the distance between $\{a,b\}$ and $\{a\}$ ought to be larger than the distance between $\{a\}$ and itself but smaller that the distance between $\{a\}$ and $\{b\}$:

$$d(\{a\},\{a\}) < d(\{a\},\{a,b\}) < d(\{a\},\{b\})$$

One might think of simply using the mean distance of the cross product between sets A and B of responses, but would make the distance between the $\{a,b\}$ and $\{b,a\}$ non-zero, so we developed a measure that is slightly more abstract, arriving at the following definition:

$$d(A,B) \doteq \frac{1}{|C|} \text{ Min } d(C), \qquad \text{where } C \text{ covers } A \times B$$

We stipulate that a set of ordered pairs C COVERS $A \times B$ as long as every element in A occurs as the first element of some pair in C and likewise every element of B occurs as a second element in a pair in C. $d(C)$ is just the sum of the distances in the set of ordered pairs. Note that this definition has the consequence that $d(\{a,b\},\{b,a\}) = d(\{a,b\},\{b,a\}) = 0$ The minimum cost cover in this case is $\{< a,a >, < b,b >\}$, whether the distances sum to 0.

4.2 Surprising preliminary analyses

After working out potential solutions to these two issues, we proceeded to first analyses, and were surprised when we clustered the aggregate lexical distances to obtain the result on the left in Figure 3, which doesn't correspond in the least to anything we'd read on American dialect areas! Given how instable clustering sometimes can be, we verified the analysis by applying multi-dimensional scaling (MDS, Nerbonne, Heeringa, and Kleiweg (1999)) to the aggregate distance table, but the impossible division cannot be blamed on clustering.[5] As the middle map shows in Figure 3 shows, the picture is even more incoherent when we include larger numbers of clusters.

After a good deal of exploration in the LAMSAS, including analysis of the various questionnaires used the years in which interviews were held, Peter Kleiweg noticed that the field workers differed enormously in the number of responses they recorded. Figure 4 shows that while Lowman was remarkably consistent in recording about the same number of responses in each interview, the other field workers were much less consistent. We also considered trying to use only the first response provided, but it wasn't clear that the first response provided represented the preferred response of the informant. The fact that the fieldworkers had collected essentially incommensurable sets of responses hadn't handicapped earlier, manual work with the data set, but I think that we were the first to point out that the discrepancies existed. Fortunately, Lowman's consistency meant that we could conduct and publish an analysis on a substantial subset of the LAMSAS data (Nerbonne & Kleiweg, 2003). Figure 5 show the areal division arising from the treatment sketched here; it corresponds

[5] Leinonen, Çöltekin, and Nerbonne (2016) present an MDS check on clustering results into the *Gabmap* web application for the analysis of language variation.

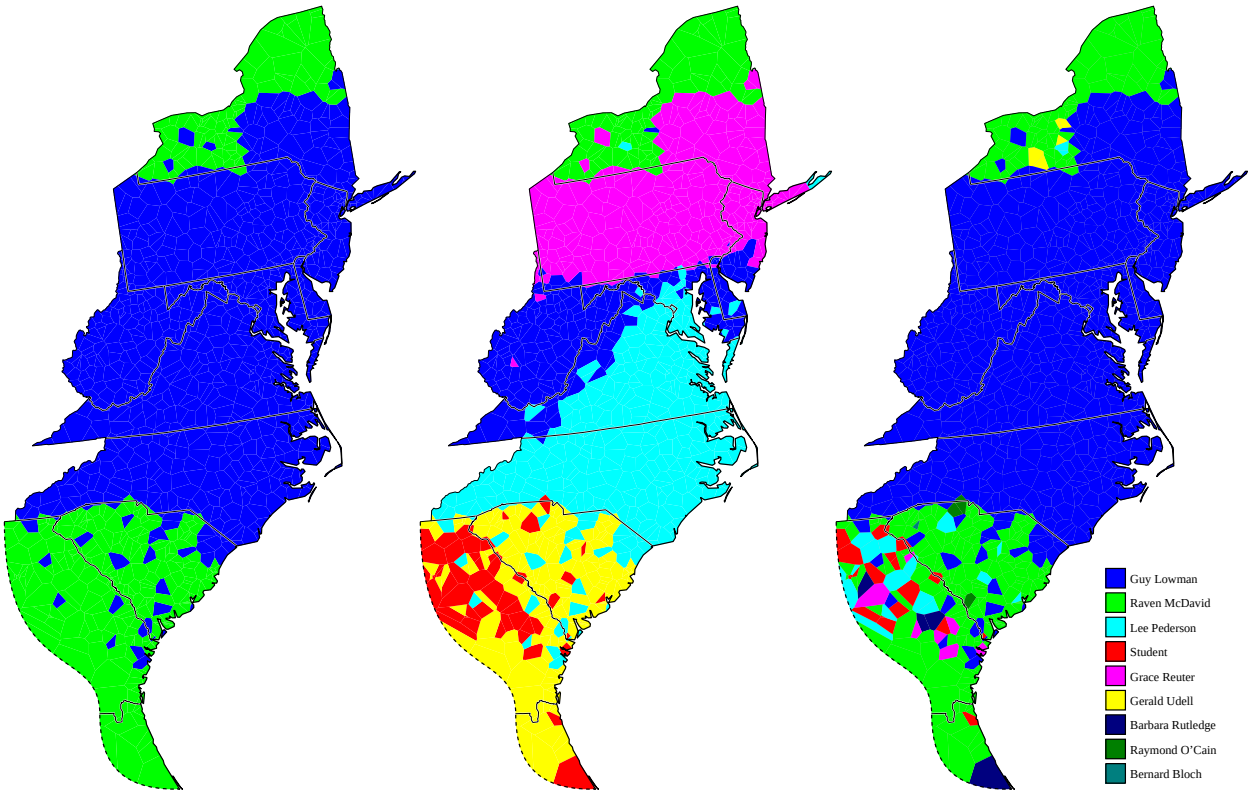

Figure 3: Preliminary results of clustering based on lexical choice in LAMSAS, where the legend on the right, showing the fieldworkers responsible for the data collection, provides an interpretation only for the rightmost map — i.e., where the fieldworker collected data. Presented at *Methods in Dialectology X*, Joensuu, but not included in the black and white publication.

closely to a controversial division originally proposed by Kurath (1949).

It turns out, by the way, that we were able to correct for the differences between the two fieldworkers at an aggregate level, essentially using standardized scores determined for each fieldworker in turn (z-scores), which we then used for comparisons. Wieling and Nerbonne (2011) use a correction on transcription practices to deal with a similar problem involving comparison in a data set where two transcription teams disagreed.

4.3 Reflections

So the degree of success in the work on lexical overlap among the LAMSAS sites is mixed. We were able to compare different standard CL techniques — stemming, and edit distance – as well as inverse frequency weighting (appealed to in particular as a means of detecting historical affinity) in order to make sense of a difficult data set. Further, we were able to extend the normal comparison of categorical data (same vs. different) to situations in which multiple responses are found.

But we were nonetheless taken aback by how incommensurable the data was with respect to the different field workers. Using what are common CL techniques (with the extensions mentioned) enabled a lexical analysis of the full set of responses for about 70% of the data, but the differences in the number of responses collected per field was never settled satisfactorily (*pace* the remarks above). It was a lucky coincidence that one fieldworker had collected 70% of the data, that he was very consistent in the number of responses he elicited, and that the area he worked in was geographically coherent. This meant that an analysis of his data alone was worthwhile.

Overall the exercise was successful, but it certainly illustrates how easily one can be surprised by non-standard data.

5 Final reflections

The most important reasons for examining non-standard data with CL methods are the fact that non-standard data represents a great deal of language behavior, and that it serves as the object of scientific study in linguistics as a whole. This is true of the syntax of non-native second-language learners, the accents of non-native speakers, and the vocabularies of different dialect speakers.

Computational linguists have a good deal to offer to the various subfields of linguistics studying non-standard data. By automating steps in analysis we make the analyses replicable (and modifiable), we improve opportunities for comparing similar analyses, and perhaps most importantly, we enable the analyses of large sets of data, providing more comprehensive views.

The data itself can be tricky to work with, however, as scientists in other fields are often specialized in a single language or language pair, as we saw in the case of the work on the syntax of Finnish emigrés to Australia, and this means that the data will not be varied enough to support all the research questions one would like to ask — in this case the question of the generality and validity of the techniques for a range of cases. In other sub-disciplines, the data simply won't have been collected with an eye to answering some interesting questions, as we saw in the case of the foreign accents, where, we hasten to add, the restriction might have been obvious to researchers who had familiarized themselves with the theoretical discussion beforehand. Finally, we note that non-automated analyses do not impose expectations that data be commensurate to the same strict degree (as automated ones), meaning that surprises can be in store even in data sets that are respected as standards in the field. The LAMSAS data provides an example of this.

One can protect oneself from some of these risks by seeking collaboration with domain experts, which is to be recommended in any case, as a way of making the work richer and better informed. Further, it makes sense to approach novels sorts of data — and even novel sources of data of a sort one suspects is familiar — with a broad range of potential research questions.

There is an awful lot of interesting work still to be done!

References

Auer, P., & Hinskens, F. (1996). The convergence and divergence of dialects in Europe. New and not so new developments in an old area. *Sociolinguistica*, *10*, 1–30.

Brants, T. (2000). TnT: A statistical part-of-speech tagger. In *Proceedings of the Sixth Conference on Applied Natural Language Processing* (pp. 224–231).

De Bot, K., Lowie, W., & Verspoor, M. (2005). *Second language acquisition: An advanced resource book*. Psychology Press.

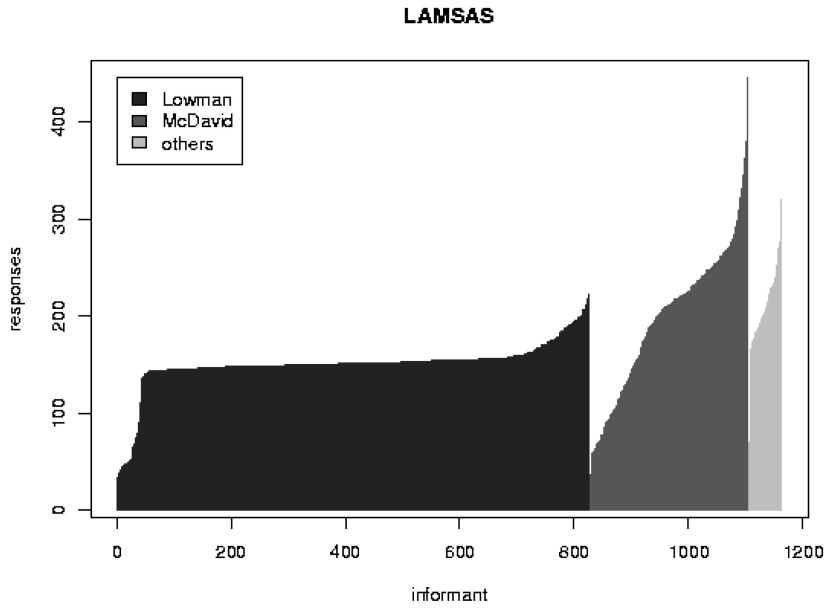

Figure 4: The number of responses per interview broken down by fieldworker. Lowman's interviews were remarkably consistent, allowing comparative interpretations, while others were not. From Nerbonne and Kleiweg (2003)

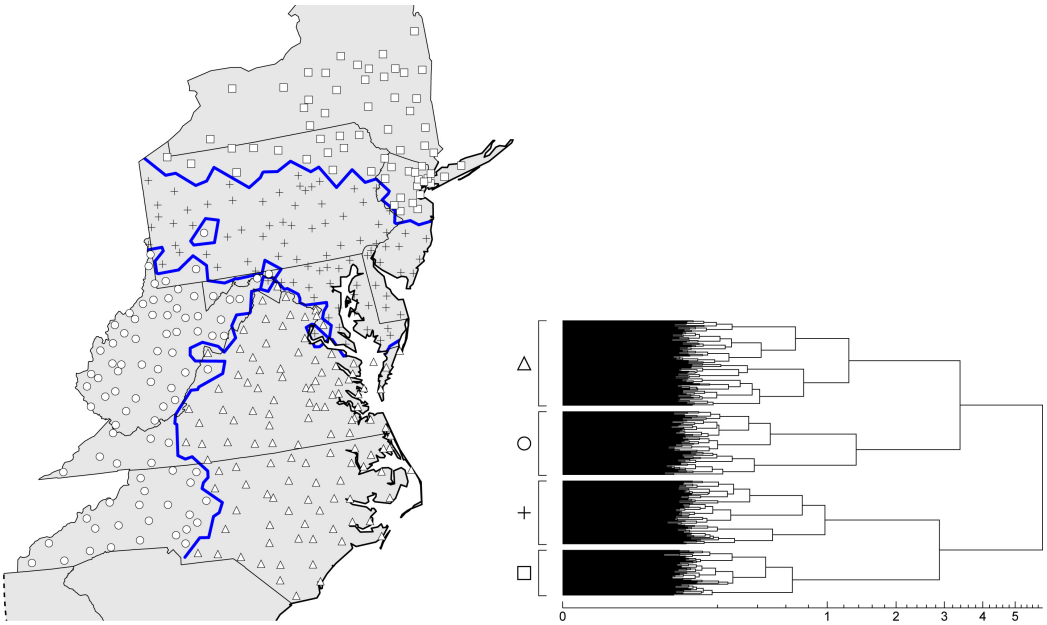

Figure 5: The final analysis of Lowman's lexical data, which, incidentally jibes well with Kurath's division. From Nerbonne and Kleiweg (2003)

Di Buccio, E., Nunzio, G. M. D., & Silvello, G. (2014, May). A vector space model for syntactic distances between dialects. In N. Calzolari et al. (Ed.), *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14).* Reykjavik, Iceland: European Language Resources Association (ELRA).

Eisenstein, J., O'Connor, B., Smith, N. A., & Xing, E. P. (2014). Diffusion of lexical change in social media. *PloS ONE*, *9*(11), e113114.

Ellis, R. (1994). *The study of second language acquisition*. Oxford University.

Goebl, H. (1984). Dialektometrische Studien: Anhand italoromanischer, rätoromanischer und galloromanischer Sprachmaterialien aus AIS und ALF. 3 Vol. *Tübingen: Max Niemeyer*.

Heeringa, W., Kleiweg, P., Gooskens, C., & Nerbonne, J. (2006). Evaluation of string distance algorithms for dialectology. In *Proceedings of the Workshop on Linguistic Distances* (pp. 51–62).

Jurafsky, D., Chahuneau, V., Routledge, B. R., & Smith, N. A. (2014). Narrative framing of consumer sentiment in online restaurant reviews. *First Monday*, *19*(4). doi: http://dx.doi.org/10.5210/fm.v19i4.4944

Kondrak, G., & Dorr, B. (2006). Automatic identification of confusable drug names. *Artificial Intelligence in Medicine*, *36*(1), 29–42.

Kretzschmar Jr., W. A. (1993). *Handbook of the Linguistic Atlas of the Middle and South Atlantic States*. University of Chicago Press.

Kruskal, J. B. (1983). An overview of sequence comparison: Time warps, string edits, and macromolecules. *SIAM Review*, *25*(2), 201–237.

Kurath, H. (1949). *A word geography of the Eastern United States*. Ann Arbor: University of Michigan Press.

Lauttamus, T., Nerbonne, J., & Wiersma, W. (2007). Detecting syntactic contamination in emigrants: the English of Finnish Australians. *SKY Journal of Linguistics*, *20*, 273–307.

Leinonen, T., Çöltekin, Ç., & Nerbonne, J. (2016). Using Gabmap. *Lingua*, *178*, 71–83.

Nabende, P., Tiedemann, J., & Nerbonne, J. (2010). Pair hidden Markov models for named entity matching. In T. Sobh (Ed.), *Innovations and advances in computer sciences and engineering* (pp. 497–502). Springer.

Nelson, G., Wallis, S., & Aarts, B. (2002). *Exploring natural language: Working with the British component of the International Corpus of English*. Amsterdam: John Benjamins Publishing.

Nerbonne, J. (2009). Data-driven dialectology. *Language and Linguistics Compass*, *3*(1), 175–198.

Nerbonne, J., Heeringa, W., & Kleiweg, P. (1999). Edit distance and dialect proximity. In D. Sankoff & J. Kruskal (Eds.), *Time warps, string edits and macromolecules: The theory and practice of sequence comparison* (2nd ed., p. i-iv). Stanford: CSLI.

Nerbonne, J., & Kleiweg, P. (2003). Lexical distance in LAMSAS. *Computers and the Humanities*, *37*(3), 339–357.

Nerbonne, J., & Kleiweg, P. (2007). Toward a dialectological yardstick. *Journal of Quantitative Linguistics*, *14*(2-3), 148–166.

Nguyen, D., Doğruöz, A. S., Rosé, C. P., & de Jong, F. (Accepted to appear). Computational sociolinguistics: A survey. *Computational Linguistics*. Retrieved from `arXivpreprintarXiv:1508.07544`

Opas-Hänninen, L. L., Hirvonen, P., Juuso, I., & Lauttamus, T. (2005). Happen I not talking good English: The progressive aspect in the English of Finnish Australians. In *Methods XII: Twelfth international conference on methods in dialectology* (pp. 1–5).

Sanders, N. C. (2007). Measuring syntactic difference in British English. In *Proceedings of the ACL 2007 Student Research Workshop* (pp. 1–6). Prague, Czech Republic: Association for Computational Linguistics. Retrieved from `http://www.aclweb.org/anthology/P07-3001`

Sanders, N. C., & Chin, S. B. (2009). Phonological distance measures. *Journal of Quantitative Linguistics*, *16*(1), 96–114.

Spruit, M. R., Heeringa, W., & Nerbonne, J. (2009). Associations among linguistic levels. *Lingua*, *119*(11), 1624–1642.

Vanhove, J. (2013). The critical period hypothesis in second language acquisition: A statistical critique and a reanalysis. *PLoS ONE*, *8*(7), e69172.

Watson, G. J. (1996). The Finnish-Australian

English corpus. *ICAME Journal*, *20*, 41–70.

Weinberger, S. H., & Kunath, S. A. (2011). The Speech Accent Archive: Towards a typology of English accents. In J. Newman, R. H. Baayen, & S. Rice (Eds.), *Corpus-based studies in language use, language learning, and language documentation* (pp. 265–281). Amsterdam: Rodopi.

Weinreich, U. (1968). *Languages in contact*. The Hague: Mouton.

Wieling, M., Bloem, J., Baayen, R. H., & Nerbonne, J. (2014). Determinants of English accents. In J. Wahle, M. Köllner, H. Baayen, G. Jäger, & T. Baayen-Oudshoorn (Eds.), *Proceedings of the 6th Conference on Quantitative Investigations in Theoretical Linguistics.* (Data package in *The Mind Research Repository*, Potsdam) doi: http://dx.doi.org/10.15496/publikation-8628

Wieling, M., Bloem, J., Mignella, K., Timmermeister, M., & Nerbonne, J. (2014). Automatically measuring the strength of foreign accents in English. *Language Dynamics and Change*, *4*(2), 253–269.

Wieling, M., Margaretha, E., & Nerbonne, J. (2012). Inducing a measure of phonetic similarity from pronunciation variation. *Journal of Phonetics*, *40*(2), 307–314.

Wieling, M., & Nerbonne, J. (2011). Measuring linguistic variation commensurably. *Dialectologia: revista electrònica*, *Special issue II*, 141–162.

Wiersma, W., Nerbonne, J., & Lauttamus, T. (2011). Automatically extracting typical syntactic differences from corpora. *Literary and Linguistic Computing*, *26*(1), 107–124. doi: 10.1093/llc/fqq017

What to do about non-standard (or *non-canonical*) language in NLP

Barbara Plank
University of Groningen
b.plank@rug.nl

Abstract

Real world data differs radically from the benchmark corpora we use in natural language processing (NLP). As soon as we apply our technologies to the real world, performance drops. The reason for this problem is obvious: NLP models are trained on samples from a limited set of *canonical varieties* that are considered *standard*, most prominently English newswire. However, there are many dimensions, e.g., sociodemographics, language, genre, sentence type, etc. on which texts can differ from the standard. The solution is not obvious: we cannot control for all factors, and it is not clear how to best go beyond the current practice of training on homogeneous data from a single domain and language.

In this paper, I review the notion of canonicity, and how it shapes our community's approach to language. I argue for leveraging what I call *fortuitous data*, i.e., non-obvious data that is hitherto neglected, hidden in plain sight, or raw data that needs to be refined. If we embrace the variety of this heterogeneous data by combining it with proper algorithms, we will not only produce more robust models, but will also enable adaptive language technology capable of addressing natural language variation.

1 Introduction

The publication of the Penn Treebank Wall Street Journal (WSJ) corpus in the late 80s has undoubtedly pushed NLP from symbolic computation to statistical approaches, which dominate our field up to this day. The WSJ has become the NLP benchmark dataset for many tasks (e.g., part-of-speech tagging, parsing, semantic role labeling, discourse parsing), and has developed into the *de-facto* "standard" in our field.

However, while it has advanced the field in so many ways, it has also introduced almost imperceptible biases: why is newswire considered more standard or more canonical than other text types? Journalists are trained writers who make fewer errors and adhere to a codified norm.[1] But let us pause for a minute. If NLP had emerged only in the last decade, would newswire data still be our canon? Or would, say, Wikipedia be considered canonical? User-generated data is less standardized, but is highly available. If we take this thought further and start over today, maybe we would be in an 'inverted' world: social media is standard and newswire with its 'headlinese' is the 'bad language' (Eisenstein, 2013). It is easy to collect large quantities of social media data. Whatever we consider canonical, all data comes with its biases, even more democratic media like Wikipedia carry their own peculiarities.[2]

It seems that what is considered canonical hitherto is mostly a historical coincidence and motivated largely by availability of resources. Newswire has and actually still *does* dominate our field. For example, in Figure 1, I plot domains versus languages for the treebank data in version 1.3 of the on-going Universal Dependencies[3] project (Nivre et al., 2015). Almost all languages include newswire, except ancient languages (for obvious reasons), English (since most data comes from the Web Treebank) and Khazak, Chinese (Wikipedia). While including other domains and languages is highly desirable, it is impossible

[1] We do not explicitly concern us here with issues of language prescription, but rather on the assumption-heavy perceptions of some instances of language as 'more normal'.

[2] For instance, the demographics of Wikipedia shows that mostly young single men aged 18-30 contribute, see https://strategy.wikimedia.org/wiki/Wikimedia_users#Demographics

[3] http://universaldependencies.org/

to find unbiased data.[4] Let's be *aware* of this fact and try to collect enough biased data.

Processing non-canonical (or non-canonical) data is difficult. A series of papers document large drops in accuracy when moving across domains (McClosky, 2010; Foster et al., 2011, inter alia). There is a large body of work focusing on correcting for domain differences. Typically, in domain adaptation (DA) the task is to adapt a model trained on some source domain to perform better on some new target domain. However, it is less clear what really folds into a *domain*. In Section 5, I will review the notion of domain and propose what I call *variety space*.

Is the *annotation* of non-canonical also more difficult, just like its processing appears to be? Processing and annotating are two aspects, and the difficulty in one, say processing, does not necessarily propagate the same way to annotation (Plank et al., 2015). However, very little work exists on disentangling the two. The same is true for examining what really constitutes a domain. What remains is clear: the challenge is all about *variations* of data. Language continuously changes, for various reasons (different social groups, communicative purposes, changes over time), and so we will continuously face interesting challenges, both for processing and annotation.

In the remainder I will look at the NLP community's approach to face these challenges. I will outline one potential way to go about it, arguing for the use of *fortuitous data*, and end by returning to the question of domain.

2 What to do about non-standard data

There are generally three main approaches to go about non-standard data.

2.1 Annotate more data

Annotating more data is a first and intuitive solution. However, it is naïve, for several reasons.

Domain (whatever that means) and *language* (whatever that comprises) are two factors of text variation. Now take the cross-product between the two. We will never be able to create annotated data that spans all possible combinations. This is the problem of *training data sparsity*, illustrated in Figure 1. The figure only shows a tiny subset of

[4] This is related to the problem of overexposure in ethics, e.g., (Hovy and Spruit, 2016).

	news	*fiction*	*nonfict.*	*blog*	*bible*	*legal*	*medical*	*social*	*spoken*	*wiki*	*web*	*reviews*
Anc. Greek		✓	✓		✓							
Arabic	✓											
Basque	✓	✓										
Bulgarian	✓	✓				✓						
Catalan	✓											
Chinese										✓		
Croatian	✓									✓		
Czech	✓		✓			✓	✓					✓
Danish	✓	✓	✓						✓			
Dutch	✓						✓			✓		
English		✓	✓	✓				✓	✓		✓	✓
Estonian	✓	✓										
Finnish	✓	✓		✓		✓				✓		
French	✓			✓						✓		✓
Galician	✓		✓			✓	✓					
German	✓									✓		✓
Gothic					✓							
Greek	✓								✓	✓		
Hebrew	✓											
Hindi	✓											
Hungarian	✓											
Indonesian	✓			✓								
Irish	✓	✓				✓					✓	
Italian	✓					✓				✓		
Kazakh		✓								✓		
Latin		✓	✓		✓							
Latvian	✓											
Norwegian	✓		✓	✓								
O.Slavonic					✓							
Persian	✓	✓	✓			✓	✓	✓	✓			
Polish	✓	✓	✓									
Portuguese	✓			✓								
Romanian	✓	✓	✓			✓	✓			✓		
Russian	✓	✓	✓							✓		
Slovenian	✓	✓	✓						✓			
Spanish	✓			✓						✓		✓
Swedish	✓	✓	✓						✓			
Tamil	✓											
Turkish	✓		✓									

Figure 1: The problem of *training data sparsity* illustrated for parsing: available annotated data in languages and domains; subset of syntactically-annotated treebanks from Universal Dependencies v1.3 for which domain/genre info was available.

the world's languages, and a tiny fraction of potential *domains* out there. The problem is that most of the data that is available out there is unlabeled. Annotation requires time. At the same time, ways of communication change, so what we annotate today might be very distant to what we need to process tomorrow. We cannot just "annotate our way out" (Eisenstein, 2013). Moreover, it might not be trivial to find the right annotators; annotation schemes might need adaptation as well (Zinsmeister et al., 2014) and tradeoffs for doing so need to be defined (Schneider, 2015).

What we need is *quick ways to semi-automatically gather annotated data*, and use more unsupervised and weakly supervised approaches.

2.2 Bring training and test data closer to each other

The second approach is based on the idea of making data resemble each other more. The first strategy here is *normalization*, that is, preprocess the input to make it closer to what our technology expects, e.g. Han et al. (2013). A less known but similar approach is to artificially corrupt the training data to make it more similar to the expected target do-

main (van der Plas et al., 2009). However, normalization implies "norm", and as Eisenstein (2013) remarks: whose norm are we targeting? (e.g., *labor* vs *labour*). Furthermore, he notes that it is surprisingly difficult to find a precise notion of the normalization task.

Corrupting the training data is a less explored endeavor. This second strategy though hinges on the assumption that one knows what to expect.

What we need are models that do provide nonsensical predictions on unexpected inputs, i.e., models that include *invariant representations*. For example, our models should be capable of learning similar representations for the same inherent concept, e.g., *kiss* vs *:** or *love* vs *<3*. Recent shifts towards using sub-token level information can be seen as one step in this direction.

2.3 Domain adaptation

There is a large body of work on adapting models trained on some source domain to work better on some new target domain. Approaches range from feature augmentation, shared representation learning, instance weighting, to approaches that exploit representation induced from general background corpora. For an overview, see (Plank, 2011; Weiss et al., 2016). However, what all of these approaches have in common is an unrealistic assumption: *they know the target domain*. That is, researchers typically have a small amount of target data available that they can use to adapt their models.

An extreme case of adaptation is cross-lingual learning, whose goal is similar: adapt models trained on some source languages to languages in which few or no resources exist. Also here a large body of work assumes knowledge of the target language and requires some in-domain, typically parallel data. However, most work has focused on a restricted set of languages, only recently approaches emerged that aim to transfer from multiple sources to many target languages (Agić et al., 2016).

What we need are methods that can adapt quickly to unknown domains and languages, without much assumptions on what to expect, and use multiple sources, rather than just one. In addition, our models need to *detect* when to trigger domain adaptation approaches.

In the next parts I will outline some possibilities to address these challenges. However, there are other important areas that I will not touch upon here (e.g., evaluation).

Side benefit of:	availability	readiness
User-generated content	+	+
Annotation	-	+
Behavior	+	-

Table 1: Typology of fortuitous data.

3 Fortuitous data

What we need are models that are more robust, work better on unexpected input and can be trained from semi-automatically or weakly annotated data, from a variety of sources. In order to build such models, I argue that the key is to look for *signal* in non-obvious places, i.e., *fortuitous data*.[5]

Fortuitous data is data out there *that just waits to be harvested*. It might be in plain sight, but is neglected (available but not used), or it is in raw form and first needs to be refined (almost ready but needs refinement). *Availability* and *ease-of-use* (or *readiness*) are therefore two important dimensions that define fortuitous data. Fortuitous data is the unintended yield of a process, a promising by-product or *side benefit*.

In the following I will outline potential sources of fortuitous data. An overview is given in Table 1.

Side benefit of user-generated content This is data of high availability and high readiness, but it is often not used or "preprocessed away". This source of fortuitous data includes user-generated content like webpages, social media posts, community-efforts like Wikipedia or Wiktionary. Concrete examples include hyperlinks that can be used to build more robust named entity and part-of-speech taggers (Plank et al., 2014a), or HTML markup for parsing (Spitkovsky et al., 2010). Similarly, Wiktionary can be used to mine large pools of data for unambiguous instances (Hovy et al., 2015), or can guide constrained inference like in type-constrained POS tagging (Täckström et al., 2013; Plank et al., 2014b). Broadly speaking, exploiting the web to process the web.

Side benefit of annotation Another yield that is often disregarded is annotator disagreement. Such data has high readiness, but low availability. It is still rare for annotation efforts to release intermediate or preliminary stages of the annotation project, but such data contains precious signal.

[5]Thanks to Anders Johannsen for suggesting *fortuitous* when I was in search for a name for *serendipitous casual* data.

In fact, instead of adjudicating annotator decisions, we should embrace it. Annotator disagreement contains actual signal informative for a variety of tasks, including tagging, parsing, supersense tagging and relation extraction, e.g., (Plank et al., 2014b; Aroyo and Welty, 2015).

Side benefit of behavior When people produce or read texts, they produce loads of by-product in form of behavior data. Examples here include click-through data, but also more distant sources such as cognitive processing data like eye tracking or keystroke dynamics. In a pilot study, I found keystroke logs carry signal that can be used to inform NLP. Such data represents a potentially immense resource (imagine logging devices build into online editors or mobile phones, or eye tracking build into mobile devices). However, only very little work explored this source yet, e.g., (Barrett and Søgaard, 2015; Klerke et al., 2016). It is also the "most distant" fortuitous source, having high availability and low readiness, as data often first needs to be *refined*.

Using fortuitous data can thus be seen as a way to quickly obtain semi-automatically labeled data, from a variety of sources. If we pair fortuitous data with appropriate learning algorithms (transfer/multi-task learning), this will enable language technology that can adapt quickly to new language varieties. However, one question remains.

4 But what's in a domain?

As already noted earlier (Plank, 2011), there is *no common ground on what constitutes a domain*. Blitzer et al. (2006) attribute domain differences mostly to differences in vocabulary, Biber (1988) explores differences between corpora from a sociolinguistics perspective. McClosky (2010) considers it in a broader view: "By domain, we mean the style, genre, and medium of a document." Terms such as genre, register, text type, domain, style are often used differently in different communities (Lee, 2002), or interchangeably.

While there exists no definition of domain, work on domain adaptation is plentiful but mostly focused on assuming a dichotomy: source versus target, without much interest in *how* they differ. In fact, there is surprisingly little work on how texts *vary* and the consequence for NLP. It is established that out-of-vocabulary (OOV) tokens impact NLP performance. However, what are other factors?

Interest in this question re-emerged recently. For example, focusing on annotation difficulty, Zeldes and Simonson (2016) remark "that domain adaptation may be folding in *sentence type* effects", motivated by earlier findings by Silveira et al. (2014) who remark that "[t]he most striking difference between the two types of data [Web and newswire] has to do with imperatives, which occur two orders of magnitude more often in the EWT [English Web Treebank]." A very recent paper examines word order properties and their impact on parsing taking a control experiment approach (Gulordava and Merlo, 2016). On another angle, it has been shown that tagging accuracy correlates with demographic factors such as age (Hovy and Søgaard, 2015).

I want to propose that 'domain' is an overloaded term. Besides the mathematical definition, in NLP it is typically used to refer to some coherent data with respect to topic or genre. However, there are many other (including yet unknown factors) out there, such as demographic factors, communicational purpose, but also sentence type, style, medium, technology/medium, language, etc. At the same time, these categories are not sharply defined either. Rather than imposing hard categories, let us consider a Wittgensteinian view.

5 The variety space

I here propose to see a domain as *variety* in a high-dimensional *variety space*. Points in the space are the data instances, and regions form domains. A dataset $\mathscr{D}$ is a sample from the *variety space*, conditioned on latent factors V:

$$\mathscr{D} \sim P(X,Y|V)$$

The variety space is a unknown high-dimensional space, whose dimensions (latent factors V) include (fuzzy) aspects such as language (or dialect), topic or genre, and social factors (age, gender, personality, etc.), amongst others. A domain is a *variety* that forms a region in this complicated network of similarities, with some members more prototypical than others. However, we have neither access to the number of latent factors nor to their types. This vision is inspired by the notion of prototype theory in Cognitive Science and Wittgenstein's *graded notion* of categories. Figure 2 shows a hypothetical example of this variety space.

Our datasets are subspaces of this high-dimensional space. Depending on our task, in-

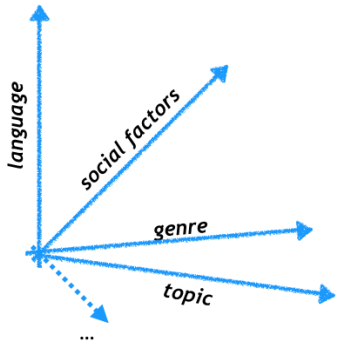

Figure 2: What's in a *domain*? Domain is an overloaded term. I propose to use the term *variety*. A dataset is a sample from the *variety space*, a unknown high-dimensional space, whose dimensions contain (fuzzy) aspects such as language (or dialect), topic or genre, and social factors (age, gender, personality, etc.), amongst others. A domain forms a region in this space, with some members more prototypical than others.

stances are sentences, documents etc. In the following I will use POS tagging as a running example to analyze what's in a domain, by referring to the datasets with the typically used categories.

Some empirical evidence - Taggers and Data Let us examine two POS taggers representative for different tagging approaches and evaluate them on several varieties. We use TNT,[6] an HMM-based tagger, and BILTY, a bidirectional LSTM tagger (Plank et al., 2016). Both taggers are trained on the WSJ training portion converted to Universal POS tags (Petrov et al., 2012). As test sets we consider parts of the Web Treebank (emails and answers), two Twitter datasets (FOSTER and GIMPEL/OCT27, Twitter sample 1 and 2 respectively), review data from two different age groups (Hovy and Søgaard, 2015), above 45 and below 35 years, and data from the CoNLL-X dataset from other Indogermanic languages.[7] These datasets were chosen to represent different varieties.

Results Table 2 shows POS tagging accuracies. First, as is well known, we see that all taggers suffer when applied to other domains. However, models trained on WSJ fare worse on data from the younger age group, thus age is a covariate. This confirms the age bias reported in (Hovy and

[6] http://www.coli.uni-saarland.de/~thorsten/tnt/

[7] http://ilk.uvt.nl/conll/free_data.html except Dutch because of joined MWU units.

VARIETY	SAMPLE	TNT	BILTY$_{\vec{w}}$	BILTY$_{\vec{w}+\vec{c}}$	OOV
(in-dom.)	wsj.test	96.63	97.25	97.85	20
domain	anwers	90.08	91.24	91.93	27
	emails	91.03	89.81	92.20	29
	Tw (foster)	90.25	92.47	92.26	28
	Tw (oct27)	65.98	66.37	67.16	52
age	U35	86.11	85.06	86.53	20
	O45	86.73	85.81	87.70	22
language	da	35.25	37.85	38.00	89
	pt	24.99	43.50	47.33	93
	sv	33.13	39.80	37.09	92

Table 2: Tagging accuracy on various test set varieties (domains, languages and age groups; Tw=Twitter), using coarse POS (Petrov et al., 2012). OOV: out-of-vocabulary rate wrt WSJ.TRAIN. Accuracy is significantly correlated with OOV rate (ρ = -0.70).

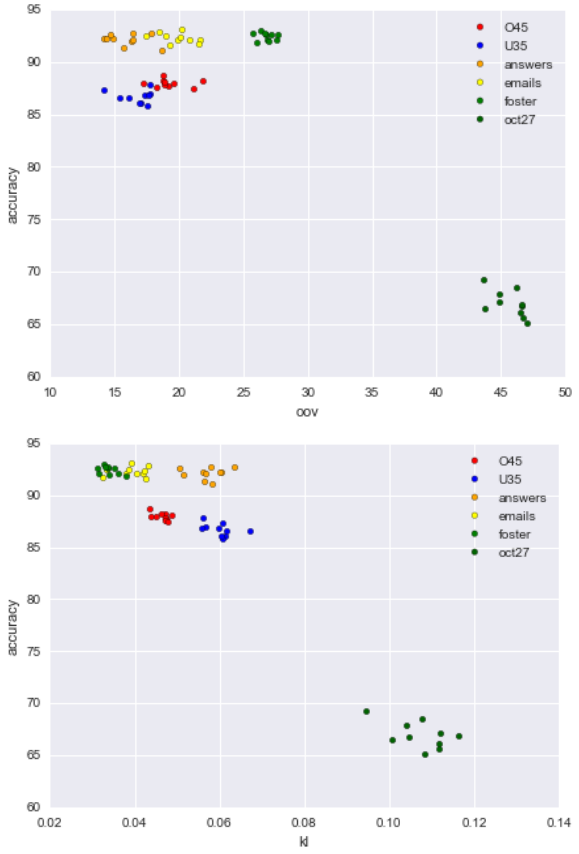

Figure 3: Accuracy of the WSJ tagger on 10 bootstrap samples ($k = 150$). Above: Accuracy versus OOV rate, Below: Accuracy vs KL divergence (src and trg gold POS bigram distributions). Different Twitter samples (green and darkgreen) exhibit very different behavior; oct27 has many OOVs *and* a high KL div; FOSTER is much closer to WSJ in terms of KL div.

Søgaard, 2015) for the same data but using different taggers. If we stretch the notion of variety to other languages, we see that performance unsurprisingly drops dramatically. Remember, we just apply an in-domain single language tagger to other languages, although only trained on WSJ here.[8] BILTY$_{\vec{w}+\vec{c}}$ performs much better on other languages than TNT. Although the neural network-based tagger that uses both word and character embeddings fares better overall, both taggers suffer similarly, their accuracy variation is highly correlated (ρ=0.95, $p < 0.01$ over all test sets; ρ=0.96 if we exclude the other languages, and ρ=0.94 if we also include OCT27).

While the two age samples have similar OOV rates, the two Twitter samples differ substantially. Twitter sample 1 (FOSTER) has an OOV rate close to others (28), while sample 2 has the highest OOV (52), every other token is an OOV word. Thus, although both come from the same medium (Twitter), they are very different samples. In general, OOV words are a major cause of performance drop. If we correlate all accuracies with OOV rate, we see a significant correlation (ρ = -0.70, p=0.02274). However, caution is needed here, the high correlation could be influenced by outliers. In fact, if we exclude the other Twitter sample (OCT27, which seems to form an outlier) and other languages, there is no significant correlation (ρ=0.23, p-val 0.6584), see Figure 3, explained next.

Rather than just inspecting numbers of single test sets, we will now plot data characteristics versus accuracy. In order to do so we take 10 bootstrap samples ($k = 150$ sentences) from the original test data, tag it with the best variant of BILTY, which uses word and character features, and evaluate it against gold POS. Figure 3 shows accuracy rates versus OOV rate (above) and accuracy vs KL-divergence between gold and predicted tag bigram distributions (lower plot). Each data point in the plot is a bootstrap sample.

The plots show that Twitter sample 2 (dark green, FOSTER) is similar in OOV rate to emails and answers; In fact, it is very close to the original dataset (WSJ), it differs the *least* from WSJ in terms of POS KL-divergence (lower plot). In contrast, Twitter sample 2 (green, OCT27) has not only high OOV rate, but it also differs highly in KL div from WSJ. The dataset contains many unusual POS sequences that are hard to predict. The same is true for age, the KL plot confirms that the tags of the younger group are harder to predict.

We see that performance varies greatly on different samples of Twitter data, as also reported earlier (Hovy et al., 2014). This suggest that Twitter is not a 'single domain'. It spans an entire range of varieties (social groups, agents, topics, even languages, etc.). Relating back to variety space, it seems that our two samples span different subspaces. Although the two samples used here do not resemble each other, they still share the commonality of being drawn from the same category (in this case, medium), mirroring Wittgenstein's theory of family resemblance, cf. (Givón, 1986). In fact, if we think about data from Twitter, we will have a prototypical member in mind, but members might vary highly. Whenever we build models for, say, Twitter, we need to be aware of these properties. The more the data varies, the more test samples we will need to achieve higher confidence in our models.

6 Conclusions

Current NLP models still suffer dramatically when applied to non-canonical data, where canonicity is a relative notion; in our field, newswire was and still often is the de-facto standard, the canonical data we typically train our models on.

While newswire has advanced the field in so many ways, it has also introduced almost imperceptible biases. What we need is to be aware of such biases, collect enough biased data, and model *variety*. I argue that if we embrace the variety of this heterogeneous data by combining it with proper algorithms, in addition to including text covariates/latent factors, we will not only produce more robust models, but will also enable adaptive language technology capable of addressing natural language variation.

Acknowledgments

I would like to thank the organizers for the invitation to the keynote at KONVENS 2016. I am also grateful to Héctor Martínez Alonso, Dirk Hovy, Anders Johannsen, Zeljko Agić and Gertjan van Noord for valuable discussions and feedback on earlier drafts of this paper.

[8] Subtoken representations are used train a single tagger for multiple languages (Gillick et al., 2015).

References

Željko Agić, Anders Johannsen, Barbara Plank, Héctor Martínez Alonso, Natalie Schluter, and Anders Søgaard. 2016. Multilingual projection for parsing truly low-resource languages. *Transactions of the Association for Computational Linguistics (TACL)*, 4:301–312.

Lora Aroyo and Chris Welty. 2015. Truth is a lie: Crowd truth and the seven myths of human annotation. *AI Magazine*, 36(1):15–24.

Maria Barrett and Anders Søgaard. 2015. Using reading behavior to predict grammatical functions. In *Proceedings of the Workshop on Cognitive Aspects of Computational Language Learning*, pages 1–5.

Douglas Biber. 1988. *Variation across speech and writing*. Cambridge University Press.

John Blitzer, Ryan McDonald, and Fernando Pereira. 2006. Domain adaptation with structural correspondence learning. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 120–128. Association for Computational Linguistics.

Jacob Eisenstein. 2013. What to do about bad language on the internet. In *Proceedings of the Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, pages 359–369, Atlanta.

Jennifer Foster, Özlem Cetinoglu, Joachim Wagner, Joseph Le Roux, Joakim Nivre, Deirdre Hogan, and Josef van Genabith. 2011. From news to comment: Resources and benchmarks for parsing the language of web 2.0. In *Proceedings of the 5th International Joint Conference on Natural Language Processing (IJCNLP)*.

Dan Gillick, Cliff Brunk, Oriol Vinyals, and Amarnag Subramanya. 2015. Multilingual language processing from bytes. *arXiv*.

Talmy Givón. 1986. Prototypes: Between Plato and Wittgenstein. *Noun classes and categorization*, pages 77–102.

Kristina Gulordava and Paola Merlo. 2016. Multilingual dependency parsing evaluation: a large-scale analysis of word order properties using artificial data. *Transactions of the Association for Computational Linguistics (TACL)*, 4:343–356.

Bo Han, Paul Cook, and Timothy Baldwin. 2013. Lexical normalization for social media text. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 4(1):1–27.

Dirk Hovy and Anders Søgaard. 2015. Tagging performance correlates with author age. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (ACL-IJCNLP)*, pages 483–488.

Dirk Hovy and Shannon L. Spruit. 2016. The social impact of natural language processing. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 591–598.

Dirk Hovy, Barbara Plank, and Anders Søgaard. 2014. When POS datasets don't add up: Combatting sample bias. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC)*, pages 4472–4475.

Dirk Hovy, Barbara Plank, Héctor Martınez Alonso, and Anders Søgaard. 2015. Mining for unambiguous instances to adapt part-of-speech taggers to new domains. In *Proceedings of the Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, pages 1256–1261.

Sigrid Klerke, Yoav Goldberg, and Anders Søgaard. 2016. Improving sentence compression by learning to predict gaze. In *Proceedings of the Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, pages 1528–1533.

David Lee. 2002. Genres, registers, text types, domains and styles: clarifying the concepts and navigating a path through the BNC jungle. *Language and Computers*, 42(1):247–292.

David McClosky. 2010. *Any domain parsing: automatic domain adaptation for natural language parsing*. Ph.D. thesis, Brown University.

Joakim Nivre, Željko Agić, Maria Jesus Aranzabe, Masayuki Asahara, Aitziber Atutxa, Miguel Ballesteros, John Bauer, Kepa Bengoetxea, Riyaz Ahmad Bhat, Cristina Bosco, Sam Bowman, Giuseppe G. A. Celano, Miriam Connor, Marie-Catherine de Marneffe, Arantza Diaz de Ilarraza, Kaja Dobrovoljc, Timothy Dozat, Tomaž Erjavec, Richárd Farkas, Jennifer Foster, Daniel Galbraith, Filip Ginter, Iakes Goenaga, Koldo Gojenola, Yoav Goldberg, Berta Gonzales, Bruno Guillaume, Jan Hajič, Dag Haug, Radu Ion, Elena Irimia, Anders Johannsen, Hiroshi Kanayama, Jenna Kanerva, Simon Krek, Veronika Laippala, Alessandro Lenci, Nikola Ljubešić, Teresa Lynn, Christopher Manning, Cătălina Mărănduc, David Mareček, Héctor Martínez Alonso, Jan Mašek, Yuji Matsumoto, Ryan McDonald, Anna Missilä, Verginica Mititelu, Yusuke Miyao, Simonetta Montemagni, Shunsuke Mori, Hanna Nurmi, Petya Osenova, Lilja Øvrelid, Elena Pascual, Marco Passarotti, Cenel-Augusto Perez, Slav Petrov, Jussi Piitulainen, Barbara Plank, Martin Popel, Prokopis Prokopidis, Sampo Pyysalo, Loganathan Ramasamy, Rudolf Rosa, Shadi Saleh, Sebastian Schuster, Wolfgang Seeker, Mojgan Seraji, Natalia Silveira, Maria Simi, Radu Simionescu, Katalin Simkó, Kiril Simov, Aaron Smith, Jan Štěpánek, Alane Suhr, Zsolt Szántó, Takaaki Tanaka, Reut Tsarfaty, Sumire Uematsu, Larraitz Uria, Viktor Varga, Veronika Vincze, Zdeněk Žabokrtský,

Daniel Zeman, and Hanzhi Zhu. 2015. Universal dependencies 1.2. LINDAT/CLARIN digital library at Institute of Formal and Applied Linguistics, Charles University in Prague.

Slav Petrov, Dipanjan Das, and Ryan McDonald. 2012. A universal part-of-speech tagset. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC)*.

Barbara Plank, Dirk Hovy, Ryan McDonald, and Anders Søgaard. 2014a. Adapting taggers to Twitter using not-so-distant supervision. In *Proceedings of the 25th International Conference on Computational Linguistics (COLING)*, pages 1783–1792.

Barbara Plank, Dirk Hovy, and Anders Søgaard. 2014b. Learning part-of-speech taggers with inter-annotator agreement loss. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, pages 742–751.

Barbara Plank, Héctor Martínez Alonso, and Anders Søgaard. 2015. Non-canonical language is not harder to annotate than canonical language. In *Proceedings of the 9th Linguistic Annotation Workshop (LAW IX)*, pages 148–151.

Barbara Plank, Anders Søgaard, and Yoav Goldberg. 2016. Multilingual part-of-speech tagging with bidirectional long short-term memory models and auxiliary loss. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 412–418.

Barbara Plank. 2011. *Domain adaptation for parsing*. Ph.D. thesis, University of Groningen.

Nathan Schneider. 2015. What I've learned about annotating informal text (and why you shouldn't take my word for it). In *Proceedings of the 9th Linguistic Annotation Workshop (LAW IX)*, pages 152–157.

Natalia Silveira, Timothy Dozat, Marie-Catherine De Marneffe, Samuel R Bowman, Miriam Connor, John Bauer, and Christopher D Manning. 2014. A gold standard dependency corpus for English. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC)*, pages 2897–2904.

Valentin I. Spitkovsky, Daniel Jurafsky, and Hiyan Alshawi. 2010. Profiting from mark-up: Hyper-text annotations for guided parsing. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 1278–1287.

Oscar Täckström, Dipanjan Das, Slav Petrov, Ryan McDonald, and Joakim Nivre. 2013. Token and type constraints for cross-lingual part-of-speech tagging. *Transactions of the Association for Computational Linguistics (TACL)*, 1:1–12.

Lonneke van der Plas, James Henderson, and Paola Merlo. 2009. Domain adaptation with artificial data for semantic parsing of speech. In *Proceedings of the Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, pages 125–128.

Karl Weiss, Taghi M Khoshgoftaar, and DingDing Wang. 2016. A survey of transfer learning. *Journal of Big Data*, 3(1):1–40.

Amir Zeldes and Dan Simonson. 2016. Different flavors of GUM: Evaluating genre and sentence type effects on multilayer corpus annotation quality. In *Proceedings of the 10th Linguistic Annotation Workshop (LAW X)*, pages 68–78.

Heike Zinsmeister, Ulrich Heid, and Kathrin Beck. 2014. Adapting a part-of-speech tagset to non-standard text: The case of STTS. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC)*, pages 4097–4104.

Bootstrapped OCR error detection for a less-resourced language variant

Adrien Barbaresi
Berlin-Brandenburg Academy of Sciences & Austrian Academy of Sciences
barbaresi@bbaw.de

Abstract

This study focuses on isolated error detection in a retro-digitized newspaper corpus published from 1946 to 1990 in the former German Democratic Republic. As there are OCR errors throughout the corpus but no clean reference for this variant of German, automatic OCR correction implies to overcome data sparseness and non-standard spelling, including compounds and inflected forms. The contributions of this paper are (1) a method to bootstrap detection of potential misspellings, (2) an assessment of several types of training data, and (3) an evaluation of several off-the-shelf candidate selection techniques. The chosen solution based on statistical affix analysis reaches an accuracy 10 points higher than existing morphological analysis systems on error detection, while a combination of fuzzy and approximate string search performs best for error correction. The criteria are met since it is possible to correct erroneous tokens without introducing too much noise.

1 Introduction

The study presented in this paper stems from a collaboration with historians to work on a diachronic newspaper corpus published in and at the time of the former German Democratic Republic (GDR/East Germany). The corpus has been digitized by a library consortium with limited resources, and the advertised quality is 95% error-free content. While no precise unit is given, it can be assumed this is on character level, which could qualify as average optical character recognition (OCR) accuracy (Holley, 2009), and which also leaves much room for improvement on token level. Numerous OCR errors can be expected throughout the texts, i.e. neither author ignorance, nor typographical errors on typing, but transmission and storage errors (Peterson, 1980).

To reduce the error rate, automatic post-processing of digitized documents is necessary. As the retro-digitized newspaper (*Neues Deutschland*) is a first attempt to grasp language use in the GDR on a large scale, there are no available corpora of this kind to train statistical models on or evaluate the results, although commonly used noisy channel models (Brill and Moore, 2000) work best on manually corrected training data, and system evaluations are performed on series of string pairs (Eger et al., 2016). In the absence of a gold standard, a bootstrap method has to be found in order to predict errors accurately without a reference. The overall precision has to be high, otherwise the correction process could degrade the corpus more than it improves it.

I focus on non-word misspellings, strings that are not found even in a large dictionary (Flor, 2012), and I develop a corrector, which implies detecting misspelled words and trying to find the most likely correct word (Peterson, 1980). This has to be done on a single OCR output, methods based on different OCR engines (Klein and Kopel, 2002) are not applicable. The contributions of this paper are as follows: (1) a corpus-based method to bootstrap detection of potential misspellings; (2) an assessment of several types of training data; and (3) an evaluation of several off-the-shelf candidate selection techniques.

2 Problem description

2.1 Error detection task

In the remainder of this article, emphasis lies on isolated non-word error correction (Kukich, 1992), also known as type-wise canonicalization techniques (Jurish, 2010) and single-token non-word OCR error correction using non-contextual algo-

rithms (Flor, 2012). Word segmentation issues are existent, but the way there are processed by such a component as well as others in a classical annotation toolchain is too difficult to benchmark, so that they have to be addressed separately. The problem tackled in this article can be split into three tasks: detection of an error, generation of candidate-corrections, and ranking of the corrections (Kukich, 1992).

Since progresses in hardware have been significant since the 90s, it is now possible to design an "ideal system" which involves "broad lexical coverage" and a lexicon as large as 100,000 words (Kukich, 1992). The task can be performed using a large database of token n-gram occurrences (Carlson and Fette, 2007). However, the context of this study is far from the "idealized conditions" described by Généreux et al. (2014), i.e. no more than two edit operations and a perfect dictionary. There are indeed substantial problems with error models driven by rules when the Levenshtein distance (Levenshtein, 1966) between error and correct string is higher than 2, and the least distant string is not necessarily the best candidate.

Since there is no proper dictionary to derive all correct word forms from, the task cannot be reduced to a normalization of out-of-vocabulary tokens to an in-vocabulary standard form, as commonly formulated (Han et al., 2013). More specifically, due to the diversity of morphology and flexion in German, rare forms potentially unknown to dictionaries may be correct (e.g. *Leninschem*, dative form "relative to Lenin", or *Spitzenlastfahrweise*, a technical term used for power plants), and keeping case markers intact is paramount.

Following from the differences listed above, the task differs from classical OCR-post-correction processes in the way that the tokens to be corrected are partly divergent but fully correct utterances, and partly OCR-related errors. The ratio between them is expected to be 95 to 5%, but it is impossible to assess with precision and it varies in time. In that sense, it is comparable to normalization of short text messages in that lexical variants may be intentionally generated (Han et al., 2013), and my goal is to overcome data sparseness.

2.2 Related results

Benchmarks are hard to come by since to my best knowledge there is no quantitative study on spell-checking for texts published in the GDR. Several methods tested on English in a seminal article (Kukich, 1992), with comparatively small dictionaries, yield top accuracies between 0.75 and 0.81. Regarding inflected languages, the TISC system for Dutch advertises a precision of 0.60, a recall of 0.67, and an F-measure of 0.63 on diachronic newspaper corpora (Reynaert, 2004), while its successor TICCL achieves a precision of 0.926, a recall of 0.894, and a F-measure of 0.910 when used without lexicon on contemporary parliament acts (Reynaert, 2011). Concerning language variants, character-level models on Egyptian Arabic dialect reach an accuracy of 0.805 on out-of-vocabulary and 0.946 on in-vocabulary words (Farra et al., 2014).

2.3 Characteristics of the corpora

The *Neues Deutschland*-corpus (ND) spans practically the time of existence of the GDR: it comprises 1.46 million articles published from 1946 to 1990, and about 444 million tokens in total. Its OCR quality varies significantly due to font changes and apparently uneven digitization.

To build a reference, two comparable corpora in size and time span are taken into consideration. Both were published in the Federal Republic of Germany (West Germany): (1) *Die Zeit* (*DZ*; 1946-2015; 1.12 million articles; 529 million tokens), and (2) *Der Spiegel* print edition (*SP*; 1947-2015; 324,000 articles; 246 million tokens). These corpora have been crawled from online archives, digitization has been undertaken by the publishers; the documents used to build a corpus are thus natively digital and they are practically exempt of OCR-related errors.

Comparison on type level shows significant discrepancies between the newspaper corpora, with a higher absolute number of types for ND, and low overlapping between the types: only 23.4% of ND's alphabetic types are found in a combination of DZ and SP. This indicates that while errors may have been contained on character level, the dispersion on type level is very high, meaning that there are a relatively high number of erroneous variants for each potential error-free token, and that dictionary coverage is low in any case.

2.4 Linguistic setting

Additionally, there are peculiarities of German as spoken in the GDR which need to be clarified. The newspaper uses a written standard so that in general no dialectal/regional variance is to be expected. However, there are a number of differences regard-

ing institutions, social roles, and words used in everyday life. This is particularly true for compound names, due to the flexibility of German: between both sides of the boundary, a high number of true lexical differences are to be found in (1) comparatively unusual but frequent compounds (e.g. *antiimperialistisch*, anti-imperialistic), (2) roots and compounds typical for systemic differences (e.g. *Kombinat* for business group or conglomerate in East Germany), and (3) rare compounds due to the focus on particular aspects (e.g. *Euterkontrolle*, udder control).

Proper nouns are also potentially an issue because of the diverging national and ideological references. Nonetheless, the difference seems to be of quantitative nature, since most person and place names used in the East appear in the West, albeit with a much lower frequency. This discrepancy indicates that frequency information in reference corpora may not be significant.

3 Method

The overlap between reference and correction corpora is low, so that working on improving dictionary coverage may not be the best approach. I use a corpus-based morphological analysis to find potential OCR-errors, whereas approximate matching (Hall and Dowling, 1980) and fuzzy search algorithms (Hauser et al., 2007; Généreux et al., 2014) based on character n-gram models are used to generate candidates for replacement and find the best one.

3.1 Error detection

Morphological analysis in German is performed by software such as SMOR (Schmid et al., 2004), which is suitable for texts of this period due to its training materials. It is expected that since it somehow reflects the logic of the language, it does not output any analysis for words which do not exist, whereas it would do so for rare compounds and even proper nouns.

The method introduced here is data-driven and grounds on affix analysis (Peterson, 1980). Relevant information is stored in a trie (Fredkin, 1960), a data structure allowing for prefix search and its reverse opposite in order to look for sublexicons, an approach used for instance in the case of agglutinative languages (Agirre et al., 1992). Compound splitting is highly necessary in morphologically rich languages (Reynaert, 2004), tokens are decomposed whether thex contain hyphens or not. The smallest possible token length for learning and searching is fixed to 4 characters. The affix and morpheme trees are learned from a types list. Simple rules are added to account for joins between compounds as well as inflection-related endings (-s, -en, etc.) in order to cope with rare phenomena which might not be present in the training data. The detection algorithm consists of one or two iterations of a search for the longest prefix and suffix as well as sanity checks to see if the rest could itself be an affix or a word of the dictionary.

3.2 Candidate selection

Candidates are found and ranked using bigram and trigram similarity (Zamora et al., 1981). On top of the similarily, fuzzy string matching already used for spelling-correction in historical texts (Hauser et al., 2007; Généreux et al., 2014) as well as approximate string matching are used. The approach tends to be conservative, nothing is modified if nothing is found within the bounds of a search space. Moreover, the agreement between both search algorithms is also evaluated. To account for inflexions, endings are normalized to the form of the original token in case a correction is suggested; capitalization is also restored to the original state.

4 Results

4.1 Evaluation data

The data for this experiment consist of a "difficult but realistic" (Kukich, 1992), "clear" set of string pairs, some misspellings and some correct but rare types; it contains a fair proportion of proper nouns as well as shorter items. The candidates have been found using frequency lists and morphological analysis tools, the list is designed to be difficult for the tools at hand. For the sake of evaluation, all cases can be considered to be unambiguous.

There are 500 non-word errors with corrections, with a Levenshtein distance comprised between 1 and 5 (mean 1.7, standard deviation 0.8): *Kriegsvqrhereitung*, *Sdiwermasdiinenbau*, *Tsdiiangkaischek*. On the other hand, there are 500 rare but correctly spelled words including inflected forms for the detection of false positives: *Komsomolzen*, *Plastfolie*, *Antiimperialistischen*, *CSSR-Mädchen*, *Kleinstübertrager*, etc. The dataset is available online.[1]

[1] http://clarin.bbaw.de/de/objects/dwds:7/

	Voc. size	Precision	Recall	F-score	Accuracy
Spellchecker					
hunspell (*de_DE*)	∼ 75,000	.583	1	.737	.643
Morphological analysis (no result for the word)					
ZMORGE	∼ 78,000	.630	.926	.750	.691
MORPHISTO	∼ 18,200	.638	.948	.763	.705
SMOR	∼ 50,000	.701	.946	.805	.771
Affix tries and composition rules					
Top-10% ND	725,995	.806	.406	.540	.654
Top-10% DZ+SP	596,984	.797	.924	.856	.844
Top-10% WEB	2,205,332	.855	.846	.850	.851
Intersection DZ+SP+KERN	757,953	.842	.904	.872	.867
Top-35% KERN	814,156	.837	.914	.874	.868
Top-10% DZ+SP+KERN	897,359	.842	.908	.874	.869
Intersection DZ+SP	1,620,976	.866	.890	.878	.876

Table 1: Evaluation of several error detection strategies, ordered by ascending accuracy

4.2 Error detection

I resort to morphological analysis to see if the words are to be corrected or not, the results are summarized in Table 1. My evaluation features the Enchant interface to the hunspell spell-checker[2] (de_DE-locale), common morphological analysis software such as *Morphisto* (Zielinski et al., 2009), *SMOR* (Schmid et al., 2004) and its enriched version based on the Wiktionary *Zmorge* (Sennrich and Kunz, 2014). The models used are the standard off-the-shelve ones, since no training material is available for the texts, and since standard training is assumed to be close enough to newspaper text.

My method uses affix trees induced as described above from West-German newspaper texts, on tokens with a minimum length of 4 characters. Additionally, the DWDS core corpus (Geyken, 2007), a balanced corpus for German in the 20th century (*KERN*; 1900-1999; 123 million tokens) is taken as an error-free reference. As it has been shown that web corpora could lead to better OCR correction (Strohmaier et al., 2003; Whitelaw et al., 2009), results based on frequent word forms extracted from a giga-token "clean" web corpus of German (Barbaresi, 2016) are referenced in the benchmark (*WEB*; 2002-2015; 2.1 billion tokens), although the corpus is neither geographically nor topically focused.

The results show that the efficiency of detection does not rely primarily on vocabulary size, the training corpora are preponderant for all tested solutions. The method introduced in this article works best in terms of precision, F-score, and accuracy, albeit with vocabularies sizes ten to twenty times larger than other tools. It cannot be trained on noisy corpus data since even a frequency filter cannot eliminate all OCR errors in the training with ND. Manual screening confirms that there are errors to be found in the top-10% of ND types, showing the extent of the problem to be treated.

Clean contemporary data from the DWDS-core corpus achieve good results even if the frequency range taken for the study is stretched toward less frequent types. The types extracted from the Web corpus are not optimal: since it does not cover the right text type and the right period, much more information is needed to achieve a similar result, thus introducing more noise. However, corpus size is not an issue with web corpora, and the results still are a positive indication as to their usefulness for general purposes, with a well-balanced ratio between precision and recall. The affix models based on contemporary West-German newspapers (DZ and SP) generally achieve better results; training data featuring not a frequency filter but an intersection (types present at least once in both newspapers) seem to eliminate potential noise due to *hapax legomena* while gathering enough information to provide a small boost concerning accuracy.

The output of morphological analysis based on the top-10% types of DZ+SP is used to discriminate between the tokens in the benchmark, since my method and this dataset provide the best F-measure as well as the best accuracy.

[2]http://www.abisource.com/projects/enchant/

Algorithm	Prec.	Rec.	F-sc.	Acc.
Approximate	.942	.524	.674	.746
Fuzzy	.922	.594	.723	.772
Combination	.949	.524	.675	.748

Table 2: Evaluation of error correction algorithms

4.3 Candidate selection

Due to the configuration of the data set the search space is limited to a maximum Levenshtein distance of 5. Whether candidates are ranked by distance or by frequency does not make noticeable changes because the algorithms already use a frequency measure internally. The parametrization of character n-grams does not bring a significant boost either: 2- or 3-grams achieve similar results. Punctuation and flexion rules yield small improvements. To replicate the results in order to make sure that no artifacts arise from a particular algorithm implementation, the method has been tested in Perl and Python using corresponding modules and packages[3], with similar results.

Due to the data used the maximum recall is 0.89. The results are summed up in Table 2. Approximate string search yields the best results in terms of precision while the fuzzy string search algorithm performs better in terms of recall, F-score, and accuracy. The best conservative approach seems to be a combination of fuzzy set and approximate string search (intersection). Although the recall values are low (between 50 and 60%), the accuracy on out-of-vocabulary tokens slightly falls short to the results of Farra et al. (2014) for Egyptian dialect, and this first experiment already meets the criteria for text correction, since erroneous tokens would be corrected without introducing too much noise.

Regarding qualitative evaluation, frequency-based error correction such as *Usa-Ausbeuter* (US-exploiter) in *Usa-Aushelfer* (US-aides, rare and generally used in a military context) would be grammatically correct but completely wrong as far as historical analysis is concerned. However, most recurring errors are of secondary importance as they deal with specialization (*Radialbohrmaschine* erroneously changed to *Spezialbohrmaschine*), or evolving normalization of proper nouns across time (*Bjelorußland* and *Belorußland*).

A way to address the mistakes may be to perform a proper candidate re-ranking (Flor, 2012), for instance by changing the costs for Levenshtein distance calculation (Hauser et al., 2007). First tests show two difficulties due to discrepancies and inflected forms: either the solution is not even in the candidate list or the distance costs do not perform evenly.

[3]Python: *marisa-trie*, *fuzzyset*, and *ngram* modules.
Perl: *Tree::Trie*, *Text::Fuzzy*, and *String::Approx*.

5 Conclusion

I have provided a method to bootstrap detection of potential misspellings in a language variant without existing standard data. Concerning error detection, morphological analysis trumps out-of-vocabulary methods as well as regular spell-checkers. Additionally, statistical affix analysis trumps morphological analysis, with accuracies up to 10 points higher than SMOR. Clean and if possible contemporaneous corpus data make a positive difference in the benchmark, and although GDR-specific vocabulary is rare in web corpora they seem to have potential as a supplementary resource. Error correction is best performed by a combination of off-the-shelf candidate selection techniques, in order to find the right balance between statistical and rule-based approaches. In both cases, results are in line with the criteria for the task, since they would correct erroneous tokens without introducing too much noise.

Acknowledgments

This work has been supported by a CLARIN-D special interest group dedicated to late modern and contemporary digital history (*FAG-9*).

References

Eneko Agirre, Inaki Alegria, Xabier Arregi, Xabier Artola, A Díaz de Ilarraza, Montse Maritxalar, Kepa Sarasola, and Miriam Urkia. 1992. XUXEN: A spelling checker/corrector for Basque based on Two-Level morphology. In *Proceedings of the 3rd conference on Applied Natural Language Processing*, pages 119–125. Association for Computational Linguistics.

Adrien Barbaresi. 2016. Efficient construction of metadata-enhanced web corpora. In *Proceedings of the 10th Web as Corpus Workshop*, pages 7–16. Association for Computational Linguistics.

Eric Brill and Robert C Moore. 2000. An improved error model for noisy channel spelling correction. In *Proceedings of the 38th Annual Meeting on Association for Computational Linguistics*, pages 286–293.

Andrew Carlson and Ian Fette. 2007. Memory-based context-sensitive spelling correction at web scale. In *Proceedings of ICMLA*, pages 166–171. IEEE.

Steffen Eger, Tim vor der Brück, Alexander Mehler, et al. 2016. A Comparison of Four Character-Level String-to-String Translation Models for (OCR) Spelling Error Correction. *The Prague Bulletin of Mathematical Linguistics*, 105(1):77–99.

Noura Farra, Nadi Tomeh, Alla Rozovskaya, and Nizar Habash. 2014. Generalized Character-Level Spelling Error Correction. In *Proceedings of the Annual Meeting of the ACL*, pages 161–167.

Michael Flor. 2012. Four types of context for automatic spelling correction. *TAL*, 53(3):61–99.

Edward Fredkin. 1960. Trie Memory. *Communications of the ACM*, 3(9):490–499.

Michel Généreux, Egon W Stemle, Verena Lyding, and Lionel Nicolas. 2014. Correcting OCR errors for German in Fraktur font. In *Proceedings of the First Italian Conference on Computational Linguistics CLiC-it 2014 & and of the Fourth International Workshop EVALITA 2014*, pages 186–190. Pisa University Press.

Alexander Geyken. 2007. The DWDS corpus: A reference corpus for the German language of the 20th century. *Collocations and idioms: Linguistic, lexicographic, and computational aspects*, pages 23–40.

Patrick AV Hall and Geoff R Dowling. 1980. Approximate String Matching. *ACM computing surveys (CSUR)*, 12(4):381–402.

Bo Han, Paul Cook, and Timothy Baldwin. 2013. Lexical normalization for social media text. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 4(1):5.

Andreas Hauser, Markus Heller, Elisabeth Leiss, Klaus U Schulz, and Christiane Wanzeck. 2007. Information access to historical documents from the Early New High German period. In *Dagstuhl Seminar Proceedings*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik.

Rose Holley. 2009. How good can it get? Analysing and improving OCR accuracy in large scale historic newspaper digitisation programs. *D-Lib Magazine*, 15(3/4).

Bryan Jurish. 2010. More than words: using token context to improve canonicalization of historical German. *JLCL*, 25(1):23–39.

Shmuel T Klein and Miri Kopel. 2002. A voting system for automatic OCR correction. In *Proceedings of the Workshop on Information Retrieval and OCR at SIGIR*, pages 1–21.

Karen Kukich. 1992. Techniques for automatically correcting words in text. *ACM Computing Surveys (CSUR)*, 24(4):377–439.

Vladimir I Levenshtein. 1966. Binary codes capable of correcting deletions, insertions, and reversals. In *Soviet physics doklady*, volume 8, pages 707–710.

James L Peterson. 1980. Computer programs for detecting and correcting spelling errors. *Communications of the ACM*, 23(12):676–687.

Martin Reynaert. 2004. Multilingual text induced spelling correction. In *Proceedings of the Workshop on Multilingual Linguistic Ressources*, pages 117–124. Association for Computational Linguistics.

Martin Reynaert. 2011. Character confusion versus focus word-based correction of spelling and OCR variants in corpora. *International Journal on Document Analysis and Recognition (IJDAR)*, 14(2):173–187.

Helmut Schmid, Arne Fitschen, and Ulrich Heid. 2004. SMOR: A German Computational Morphology Covering Derivation, Composition and Inflection. In *LREC*.

Rico Sennrich and Beat Kunz. 2014. Zmorge: A German Morphological Lexicon Extracted from Wiktionary. In *Proceedings of LREC*, pages 1063–1067. ELRA.

Christian M. Strohmaier, Christoph Ringlstetter, Klaus U Schulz, and Stoyan Mihov. 2003. Lexical postcorrection of ocr-results: The web as a dynamic secondary dictionary? In *Proceedings of ICDAR*, pages 1133–1137.

Casey Whitelaw, Ben Hutchinson, Grace Y Chung, and Gerard Ellis. 2009. Using the web for language independent spellchecking and autocorrection. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, volume 2, pages 890–899. Association for Computational Linguistics.

EM Zamora, Joseph J Pollock, and Antonio Zamora. 1981. The use of trigram analysis for spelling error detection. *Information Processing & Management*, 17(6):305–316.

Andrea Zielinski, Christian Simon, and Tilman Wittl. 2009. Morphisto: Service-oriented Open Source Morphology for German. In *State of the Art in Computational Morphology*, pages 64–75. Springer.

${}_t\gamma$ – Inter-annotator agreement for categorization with simultaneous segmentation and transcription-error correction

Fabian Barteld **Ingrid Schröder** **Heike Zinsmeister**
Institut für Germanistik
Universität Hamburg
firstname.lastname@uni-hamburg.de

Abstract

When annotating non-standard texts such as historical texts or spoken language, tasks that are normally considered to be pure categorization tasks such as part-of-speech tagging are often combined with correcting errors in the tokenization and even the transcribed text itself along the way. As a consequence, inter-annotator agreement measures are needed that measure agreement for categorization by also taking changes in segmentation and the underlying text into account. In this paper, we present the first inter-annotator measure of this kind, text-gamma (${}_t\gamma$). Based on γ (Mathet et al., 2015), the inter-annotator agreement is measured using an alignment of the annotations. For this, we consider alignments of the annotations that follow from optimal alignments of the underlying text sequences. Furthermore, we use a specialized function to measure the disorder of the alignment. For chance-correction, we introduce a method that takes the annotation bias introduced by pre-annotation into account when estimating the expected (dis)agreement between annotators.

1 Introduction

The annotation of non-standard texts such as historical texts, spoken language, or user-generated content poses specific problems for the annotation process. Even tasks as basic as segmenting a text into tokens for subsequent part-of-speech (POS) tagging become considerably harder for such data than for standard text since whitespace often does not coincide with the boundaries of syntactical words (Barteld et al., 2014). As a consequence, human annotators are sometimes asked to check and correct the underlying tokenization along the way when annotating this kind of data. Examples for annotation guidelines that address this task explicitly are Čibej et al. (2016), giving guidelines for the normalization of Slovene Tweets and the guidelines for HiTS (Dipper et al., 2013), a POS tagset developed for historical variants of German. Furthermore, when working with data that is not born-digital such as historical texts or data that is not written in nature such as spoken language, the textual representation of the data that is annotated is already an interpretation of the original data and might contain errors. This adds the necessity of correcting the text during the process of tagging. As an example two different transcriptions of the same text are shown in (1) where an "i" followed by an "n" was corrected to "m".

(1) sambt aller v**in̄**stendicheit vthgelacht /
sambt aller v**m̄**stendicheit vthgelacht /
with all circumstances construed
'construed extensively'
(Source: Verl. Sohn)

Annotation tools developed for the annotation of non-standard text such as CoBaLT (Kenter et al., 2012) and CorA (Bollmann et al., 2014) consequently allow the annotators to change the underlying text and the segmentation into tokens during the annotation process. Effectively, this is turning the annotation from a categorization task into a combination of string editing, segmentation, and categorization.

While the annotation tools exist, there is no chance-corrected inter-annotator agreement measure for this setting available. We address this issue by presenting *text-gamma* (${}_t\gamma$) the first measure for categorization that takes into account the possibility of correcting the segmentation and the text along the way. As the quality of the transcription and the segmentation presented to the annotators affects the expected number of corrections, we also introduce a method for determining

chance correction that takes the annotation bias introduced by pre-annotations into account when estimating the expected (dis)agreement between annotators.

While ${}_t\gamma$ is usable for all kinds of segments and categories – even multiple categorizations of a segment, e.g. assigning POS tags and lemmas to tokens – with simultaneous correction of the segmentation and the underlying text done by an arbitrary number of annotators, we exemplify and evaluate this measure on data as created in a setting of tokenization and POS tagging of an historical text by two annotators.

2 The annotation task

In this section, we present a formalization of the different types of categorization tasks: (a) *pure categorization*, the traditional task, where predefined segments are labeled with a category, (b) *categorization with segmentation correction*, the extension of pure categorization to born-digital, non-standard texts such as computer-mediated communication, where the segmentation is corrected by the annotators, and (c) *categorization with segmentation and text correction*, the extension of categorization to non-standard texts that are not born-digital such as historical texts where the digitized text might contain errors that are corrected by the annotators as well as the segmentation.

For the formalization, we combine the quite similar concepts that are introduced by Mathet et al. (2015) and used in GATE (Cunningham et al., 2014).[1] We define an **annotation** as an entity that has been created by a (human or automatic) annotator, that has a type (e.g. *token*, *sentence*) and a feature set realized as a set of attribute-value pairs (e.g. *POS=noun*). An annotation has a position on a continuum in terms of start and end offsets. The **continuum** can be continuous, e.g. in the case of a timeline where the offsets represent the points in time where an annotation starts and ends. We look at cases where the continuum is a text represented by a character string and the start and end points of annotations are given by character offsets, therefore the continuum is discrete. Furthermore, annotations that are attached to the same continuum can be combined in an **annotation set**. When the continuum is text, i.e., a character string, we mark this with the subscript *text* (**annotation set$_{\text{text}}$**).

Using this terminology, the traditional task of POS tagging – an example of pure categorization – can be modeled as an iterative creation of annotation sets on the same continuum. The first iteration, which is usually done automatically by a tokenizer, creates annotations of the type *token* with non-overlapping start and end offsets. The annotations cover the continuum completely, only whitespace characters may be left uncovered.[2] The resulting annotation set$_{\text{text}}$ is the input to the second iteration of the annotation procedure – this phase is traditionally seen as the annotation proper: In this second iteration, annotators are presented with the annotated text resulting from the first iteration and add new feature-value pairs (for POS) to the annotations of type *token*. For inter-annotator agreement experiments, iteration 2 is done independently by multiple annotators, resulting in multiple annotation sets$_{\text{text}}$. Fig. 1 illustrates the three types of categorization tasks introduced above.

Fig. 1a shows the traditional setting, pure categorization. In this setting, the annotators do not change the text or the token segmentation, i.e., in our terminology, the continuum and the offsets of the annotations, respectively. In this case, each resulting annotation set$_{\text{text}}$ contains the same number of annotations and for each annotation there is exactly one corresponding annotation in the other sets, which are easily identified by the offsets. The only possible difference is in the POS values. This setting allows for a straightforward comparison of the assigned categories.

Fig. 1b shows categorization with simultaneous segmentation correction, i.e., the annotators occasionally change the start and end offsets of annotations by merging or splitting them. This results in annotation sets$_{\text{text}}$ derived from the same input, which possibly differ in the number of annotations which again might also differ in their positions on the continuum. Therefore, it is not as straightforward to identify corresponding annotations for which the assigned categories have to be

[1] Both introduce similar concepts, treating annotations as spans over a continuum. However, there are differences. For example, the annotations as used in GATE are more general than the units introduced by Mathet et al. (2015), as annotations are typed and allow for more than one category by using feature sets.

[2] Note that in our formalization tokens are independent of whitespace in the underlying texts. E.g. the string 'New York' can be treated as a multi-word unit by creating an annotation that covers the whole sequence or as two tokens by creating two annotations that cover the first and the second part respectively, leaving the whitespace uncovered. Therefore, changing the segmentation does not affect the underlying continuum.

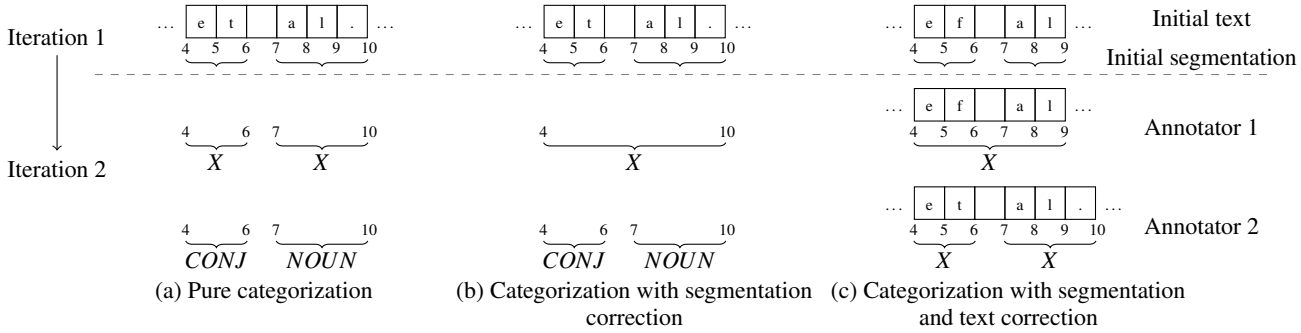

Figure 1: Different types of categorization tasks, using the universal POS tagset (Petrov et al., 2012)

compared. Still, the annotations are all attached to the same continuum.

Fig. 1c shows the case when the data is not born-digital and annotators are allowed to change the textual representation, i.e., the underlying continuum, as well as the segmentation. Textual changes can also affect the annotations, e.g., when inserting a character into the text, the offsets of all subsequent annotations need to be adapted. This is exemplified by the "." that the second annotator inserted. Therefore the last offset in the example is 10, while it is 9 for the first annotator. In the end, the resulting annotation sets might differ regarding the contained annotations. Furthermore, the annotations are attached to different continua.

This third annotation task could be split into three separate annotation processes where first, the text is corrected, then this text is segmented and in a third step the segments are labeled. Such a pipelined annotation setting would allow us to compute the inter-annotator agreement for each of the three steps independently using existing measures. However, it would introduce the need to fix the result of each step, e.g., errors in the segmentation and the transcription cannot be corrected when assigning labels. Our experience with the creation of a corpus with Middle Low German texts shows that many segmentation and/or transcription errors only become apparent while assigning POS tags. Consequently, we present the first measure for inter-annotator agreement that can be used when categorization is combined with segmentation and transcription-error correction.

3 Related work

There exist inter-annotator agreement (IAA) measures for each of the tasks described in Fig. 1 when performed individually. In wide use are measures like α (Krippendorff, 1980) and κ (Cohen, 1960) for categorization tasks. Artstein and Poesio (2008) give an overview of these and other measures for categorization tasks.

A commonly-used measure for the quality of a segmentation is *WindowDiff* (Pevzner and Hearst, 2002). However, this and related measures, are geared toward comparing an automatically created segmentation with a reference segmentation and therefore do not apply chance correction. For manually created segmentation, it is preferable to use measures that take chance correction into account like α_U (Krippendorff, 1995) that measures the degree to which segments overlap or B-based π^* (Fournier, 2013) that is designed for complete segmentation tasks where the annotations cover the whole continuum.

For a setting in which the two tasks of detecting units and categorizing them are combined, there exist only a few measures, among them different versions of ${}_u\alpha$ (Krippendorff, 2013; Krippendorff, 2015) and γ (Mathet et al., 2015). The latter is based on finding an optimal alignment between the annotations from a set of annotators, i.e., identifying the annotations that are most similar, aligning them, and then calculating the mean dissimilarity between them. For the task considered here such a measure has to be combined with a measure quantifying the dissimilarity between texts. There are a few attempts to measure the quality of transcriptions, e.g., Munyaradzi and Suleman (2013) using a normalized variant of the Levenshtein distance for manuscript transcriptions and Valenta et al. (2014) using word accuracy for speech transcription. Both do not apply chance correction. As using chance correction for IAA is considered state of the art (Artstein and Poesio, 2008), we aim to apply chance correction in our measurements. For our task the chance correction has to account

for the fact that, at least with transcription and segmentation, the annotators do not start from scratch but are presented with pre-annotations, i.e., they start with a tokenized transcription. When the impact of these pre-annotations on IAA (Fort and Sagot, 2010) is not considered, the actual agreement would be overestimated.

In the next section, we present a method to create alignments between the annotations from different annotators. Using these alignments the disagreement between aligned annotations can be calculated similarly to the way in which it is calculated in γ. However, differences in the underlying texts have to be included in the disagreement. Subsequently, we propose a method to estimate the expected agreement taking the pre-annotations into account. Finally, we evaluate our measure using corpus shuffling (Mathet et al., 2012).

4 Aligning annotations from different continua using sequence alignment

Gamma (γ) (Mathet et al., 2015) is calculated using the mean dissimilarity between aligned annotations taking the category and the position into account. The alignment used is the alignment with the lowest mean dissimilarity. For this, all possible alignments are considered in the original computation. Using this method directly is not possible in our situation, as differences in the position of units may result from different textual bases. For instance, the insertion of one letter by only one of the annotators shifts all following offsets of her annotations to the right. As a consequence, annotations that span only one letter would not overlap when comparing the texts of different annotators, leading to artificially high dissimilarities.

In example (1), the same part of the original texts is transcribed with two letters ("iñ") and with one letter ("ñ") in the two transcriptions. This influences the characeter offsets of all the following characters, e.g., the "/" starts at the position 38 in the first transcription and at position 37 in the second transcription. As it only has a length of one, there is no overlap between these two tokens when only considering their positions in the corresponding transcription.

Ignoring the position of annotations is not a solution here, since it would allow the alignment of annotations spanning the same sequence of characters even if they were from different ends of the text. Therefore, we apply a different method to find optimal alignments between annotations by using (multiple) sequence alignment (MSA). MSA is a common technique in analyzing genome sequences and an active research topic in bioinformatics (Chatzou et al., 2015). MSA has been used in natural language processing as well (Barzilay and Lee, 2002; Prokić et al., 2009; List, 2012; Kirschenbaum, 2013). Given n input sequences the result of a MSA is a set of n aligned sequences, i.e., the resulting sequences all have the same length and the characters at a given position in the sequences are aligned with each other. To accomodate differing lengths between the input sequences, gaps (represented by "_" in the examples) are inserted (cf. example 2).

(2) v o _ r w a h r
v o u r w a _ r
f _ u r w a h r

An optimal sequence alignment is one that minimizes the costs introduced by matches, mismatches and gaps. The basic algorithm to find an optimal alignment is a specialization of the algorithm described by Needleman and Wunsch (1970). Normally, the alignment of mismatches is allowed. However, then it is not always possible to perfectly align the annotations on the new sequences as can be seen from the following example:

(3) h a t s
h a t t

In (3), it is not possible to positionally align an annotation corresponding to *hat* in the first sequence with an annotation corresponding to *hatt* in the second sequence on the continuum created by sequence alignment. As such this is not a problem for aligning these annotations for the calculation of γ. However, when textual and positional dissimilarity are both integrated into the calculation of the alignments' dissimilarity, the dissimilarity between *hatt* and *hat* will be artificially high as the annotations differ both positionally and textually.

To not over-punish such settings, we do not include the position in the dissimilarity measure of ${}_t\gamma$. Furthermore, we do not consider all possible alignments of annotations but only alignments of annotations that have the same position in an optimal sequence alignment. This avoids the problem of aligning two annotations from different regions of the continuum as described above.

So far, we would not allow the alignment of *hat* and *hatt* in (3). To make this alignment pos-

sible, we only allow matches and gaps in the sequence alignment, e.g., by setting the cost for mismatches such that introducing gaps will always be preferred.

To create possible alignments of annotations, we introduce boundaries as elements into the sequences (denoted by "{" and "}" below). Now, aligned annotations can be read off directly from the aligned sequences.[3] The strings from example (3) lead to the following optimal sequence alignments in (4) and (5):

(4) { h a t } { s _ }
{ h a t _ _ _ t }

(5) { h a t _ } { s }
{ h a t t } _ _ _

Both alignments ore optimal sequence alignments even if in (4) no annotations are aligned and in (5) the annotations covering *hat* and *hatt* are aligned. In our approach, all alignments of annotations that result from an optimal sequence alignment are considered for finding the best alignment of annotations.

We want to point out the behavior of this alignment method for adjacent annotations that only partially overlap comparing two annotation sets. Take the artificial example of $\{a\}\{bbc\}$ and $\{abb\}\{c\}$. Examples (6) and (7) show two optimal alignments of these sequences:

(6) { a } { b b _ _ c }
{ a _ _ b b } { c }

(7) { a _ _ } { b b c }
{ a b b } { _ _ c }

In this case, aligning the annotations or not aligning them both result in optimal sequence alignments (both with a cost of $4 \times c_g$, where c_g denotes the cost of inserting a gap). However, in the examples (8) and (9) with the sequences $\{a\}\{bc\}$ and $\{ab\}\{c\}$, variant (9), in which the annotations are aligned is "cheaper" and hence is the only optimal sequence alignment:

(8) { a } { b _ _ c }
{ a _ _ b } { c }

(9) { a _ } { b c }
{ a b } { _ c }

In the examples (10) and (11) with sequences $\{a\}\{bbbc\}$ and $\{abbb\}\{c\}$, it is the other way round. Here the annotations are not aligned as only option (10) is an optimal sequence alignment.

(10) { a } { b b b _ _ c }
{ a _ _ b b b } { c }

(11) { a _ _ _ } { b b b c }
{ a b b b } { _ _ _ c }

For these examples, we assumed that gaps at textual positions (gap_t) have the same cost as gaps at boundary positions (gap_b). If we allow the setting of gap_b independently of gap_t a preference for or against aligning annotations that partially overlap can be chosen. Supposing that gap_t is set to 1, the following cases apply: (i) when two boundaries are less than $2 \times gap_b$ characters apart, they are always aligned, (ii) when two boundaries are exactly $2 \times gap_b$ characters apart, they can be aligned and, (iii) when two boundaries are more than $2 \times gap_b$ characters apart are never aligned. In our experiments, we set $gap_t = gap_b$.

There exist many algorithms for MSA differing in the computational complexity and the accuracy of the produced alignments. In principle all of these methods are usable to induce possible alignments of annotations. For the evaluation, where we aligned two versions of one text consisting of about 3,700 characters, we used the algorithm by Needleman and Wunsch (1970) but followed more than one path in the backtracking phase in order to obtain the different possible alignments.

Simply following all possible paths leading to optimal alignments of the sequences may be computationally intractable as the simple difference in example (1) already allows the three optimal alignments shown in (12).

(12) a. v _ i n s
v m _ _ s

b. v i _ n s
v _ m _ s

c. v i n _ s
v _ _ m s

As we are only interested in inducing alignments of annotations, the above differences do not influence the result. Hence, we only follow alternative paths when annotation boundaries are involved. Furthermore, we exploit inequality (1) (see Section 5) that holds for the dissimilarity measure that we use, and bias the alignments towards aligning annotations by aligning boundaries if possible. In (13) only the second alignment is produced.

[3] Note that this method requires the annotations of one annotator to be non-overlapping. Otherwise, the character denoting the end of an annotation can be ambiguous.

(13) a. { n e } { m a g }
{ - - - - m a g }

b. { n e } { m a g }
- - - - { m a g }

Aligning the text used for our experiments with its shuffled version (see Section 7), where the text, the segmentation and the categories are changed, and the magnitude was set to 1, leads to only one annotation alignment in the mean produced by this method (out of ten runs, only in one run two alignments were produced).

5 Calculating the observed disorder

As γ (Mathet et al., 2015), ${}_t\gamma$ is calculated based on the disorder of an optimal alignment ($\delta(a)$) between the annotations from different annotators. An alignment $\bar{a}$ is considered optimal when it minimizes the disorder. Unlike Mathet et al. (2015), we do not consider all possible alignments between annotations when looking for the optimal alignment but only the alignments that result from an optimal sequence alignment as described in the previous section. Annotations from different annotators are aligned when they cover the same span in the aligned sequences. Therefore, for each of the optimal sequence alignments exactly one alignment of annotations is defined consisting of unitary alignments ($\breve{a}$) between annotations or annotations and empty elements ($\emptyset$).

Following Mathet et al. (2015), the disorder of an alignment is defined as

$$\bar{\delta}(\bar{a}) = \frac{1}{\bar{x}} \sum_{i=1}^{|\bar{a}|} \breve{\delta}(\breve{a}_i)$$

where $\breve{\delta}$ is the **dissimilarity** between the aligned annotations. An alignment of an annotation with the empty element has a dissimilarity of $\Delta_\emptyset$ (cf. Mathet et al. (2015)).

We define the dissimilarity of an alignment of two annotations u and v as

$$d_t\gamma(u,v) = \frac{1}{n+1}(d_t(text(u),text(v)) + \sum_{i=1}^{n} d_i(feat_i(u),feat_i(v)))$$

where n is the number of features of the annotations (cf. Section 2). d_t is a dissimilarity measure between the texts covered by the annotations and the d_i are dissimilarity measures between the feature values. For the evaluation, we use the simple nominal dissimilarity measure which is 0 in the case of equality and $\Delta_\emptyset$ in the case of inequality for all d_x. Other d_x are usable as well, e.g., d_{cat} as described by Mathet et al. (2015), that takes overlaps between categories into account, or a string similarity metric such as the Levenshtein distance (Levenshtein, 1966) for textual differences.

Note that when using the dissimilarity measure exactly as described above, the following inequality holds:

$$d_t\gamma(u,v) \leq \Delta_\emptyset = \frac{1}{2}(d_t\gamma(u,\emptyset) + d_t\gamma(v,\emptyset)) \quad (1)$$

Therefore, the dissimilarity of an alignment is at least as high as the dissimilarity of an alignment where fewer annotations are aligned (i.e., it has more alignments with $\emptyset$). This means that many alignments created by optimal sequence alignments can be removed from the set of possible alignments for the calculation of ${}_t\gamma$.

As pointed out above, we do not consider positional differences in our dissimilarity measure. This is unproblematic since we do not align tokens that are not mapped to the same position by the sequence alignment process.

6 Calculating the expected disorder

For state-of-the-art IAA metrics, it is expected to take chance agreement into account (Artstein and Poesio, 2008). Our new measure ${}_t\gamma$ – like the original γ – measures disagreement between aligned annotations. The standard way of incorporating chance-correction to disagreement based measures is to use the ratio between the observed disagreement (D_o) and the expected disagreement (D_e), i.e. the disagreement that is expected when both of the annotators are guessing. Therefore, we define ${}_t\gamma$ exactly as γ as $1 - \frac{D_o}{D_e}$.

We follow Mathet et al. (2015) and compute D_e by sampling randomly generated annotation $\text{sets}_{\text{text}}$. Mathet et al. (2015) randomly create sets for which (i) the number of units per annotators, (ii) the categories, (iii) the length of the units of a given category, (iv) the length of gaps, and (v) overlaps between units of given categories are distributed as in the observed annotation set. Then they use these samples to estimate D_e.

This, however, estimates D_e when annotations are created without any pre-annotation which is

not the case for text corrections and tokenization in our case. Therefore, calculating D_e in this way would underestimate the actually expected disorder. Take for example two annotators annotating a text that was automatically tokenized with an error rate of 4% (Jurish and Würzner, 2013). In this case, only a small fraction of tokens needs to be changed. The expected agreement for two annotators highly disagreeing will still be substantially higher than the agreement to be expected when two tokenizations are created randomly. Therefore, we do not sample annotation sets that are randomly generated, but we create annotation sets by applying changes randomly to the pre-annotation.

Given the situation where tokenized transcriptions are annotated with POS tags, the creation of random annotation sets consists of three steps: Firstly, the text is changed, secondly the segmentation is changed, and thirdly the segments are annotated with POS tags. When modeling random annotations, we assume that all three steps are independent of each other. Further, we assume that the amount of changes (c_t and c_s for text and segmenation) the annotators perform follows a binomial distribution with the parameter n being the number of annotations. The parameter p can either be derived from the (known) quality of the pre-annotation, e.g., set to 0.04 for segmentation changes when the error rate of the tokenizer is 4%. Alternatively, it can be estimated from the observed differences between the annotation sets and the pre-annotation. Both methods can also be combined using maximum a posteriori (MAP) estimates for p (Manning and Schütze, 1999). Like Mathet et al. (2015), we use the annotations from all annotators for estimating distributions, i.e., treating annotators as interchangeable (Krippendorff, 2011).

Given the tokenized transcription, in the first step, we apply c_t text changes. For this c_t distinct annotations are chosen according to a uniform distribution. Then one of the three types of textual changes (insertion, deletion and substitution) is chosen from an equal distribution. For insertion and substitution a character is chosen based on the distribution of characters in the observed annotation set.

In a second step, the segmentation is changed by applying splits and mergers, i.e., adding or removing boundaries. This is done c_s times. For each change, one of the three operations (split, merge with left, merge with right) is chosen from a uniform distribution. Afterwards a segment is chosen again from an equal distribution, excluding the first segment for merge with left and the last segment for merge with right. Note that the annotations resulting from a split or a merger can be chosen for a subsequent change.

For the third step, i.e., labelling the tokens, in our case no pre-annotation is assumed. Therefore, we simply add labels to the tokens following the distribution of the labels in the observed annotation sets.

Using this method to create random annotation sets, we can estimate D_e by applying the same sampling method as Mathet et al. (2015).

7 Evaluation

For evaluating ${}_t\gamma$, we use the corpus shuffling method (Mathet et al., 2012). With this method a given reference annotation is changed randomly with a given magnitude m. Following Mathet et al. (2012), the shuffling is repeated with differing values of m (ranging between 0 and 1 with a stepsize of 0.05). For each of these values, the inter-annotator agreement is measured. These values show how the measure reacts to differences in two annotation sets of a specified magnitude. The values taken by the inter-annotator agreement measure should be (i) strictly decreasing with increasing magnitude m – i.e. reflect the increasing difference of the annotation sets and (ii) use the full range of possible values (Mathet et al., 2015).

We use a reference annotation set_{text} for the evaluation. The text has a length of 3,706 characters. The annotation set contains 608 tokens labeled with POS tags. We simulate a second annotation set_{text} by applying shuffling to this reference annotation. For the shuffling, three methods are applied: (i) textual change, (ii) segmentation change and (iii) label change. As shuffling all three types with the same magnitude is unrealistic (due to the pre-annotation bias), we calculate m_t for the magnitude of text changes, m_s for the magnitude of segmentation changes and m_l for the magnitude of labeling changes from a given m as follows: $m_t = 0.5 \times m$, $m_s = 0.1 \times m$ and $m_l = m$.

In each of the three steps, given a magnitude $0 \leq m_i \leq 1$, $c = m_i \times n$ changes are applied. For textual and category changes, the changes are applied to distinct annotations. As our parametrization of γ only measures if two aligned annotations

have the same text or not, each token is only considered once for text changes. The shuffling itself follows the same procedure as for the calculation of the expected disagreement.

We test three different settings that correspond to the three types of categorization tasks given in Fig. 1: (a) only categories are shuffled, (b) categories and segmentations are shuffled, and (c) categories, segmentations and the text is shuffled.

For the calculation of the expected disorder, we do not estimate the probabilities for the changes from the data to benchmark the influence of these parameters on the final agreement value. We evaluate three parameter settings: Firstly, we set the probability for text (p_t) and for segmentation changes (p_s) both to 0, i.e., the expected disorder is calculated for pure categorization (Cat). Secondly, we simulate the situation, where a text that is born-digital is automatically tokenized with an error rate of 4% (Jurish and Würzner, 2013), consequently p_t is set to 0 and p_s to 0.04 (Cat + Seg). Thirdly, we simulate the situation, where a text is automatically transcribed and tokenized afterwards with 25% of the tokens needing a textual correction. p_t and p_s are therefore set to 0.25 and 0.04 respectively (Cat + Seg + Text). Note, that the values for m_t and m_s limit the magnitude of the shuffling to approximately twice the expected error rate.

As both the shuffling and the calculation of the expected disorder is randomized, we repeat each step ten times. Figure 2 gives the mean values. The error bars denote the standard error.

For comparison, we used the DKPro Agreement package (Version 2.1.0) (Meyer et al., 2014) to compute α for the pure categorization setting and α_U with aggregation over categories for the categorization and segmentation setting. We also used the software supplied by the authors of γ[4] to calculate gamma for the categorization and segmentation settting. We only calculated γ for one shuffling, and only for magnitudes 0, 0.25, 0.5, 0.75 and 1.

As can be seen from Fig. 2, ${}_t\gamma$ shows an almost perfectly linear response to the increasing magnitude of the shuffling. The parametrizations expecting less change are always lower than the other parametrizations (except in the case of perfect agreement). This is expected as more agreement is attributed to chance.

[4] `https://gamma.greyc.fr` (Version 1.0).

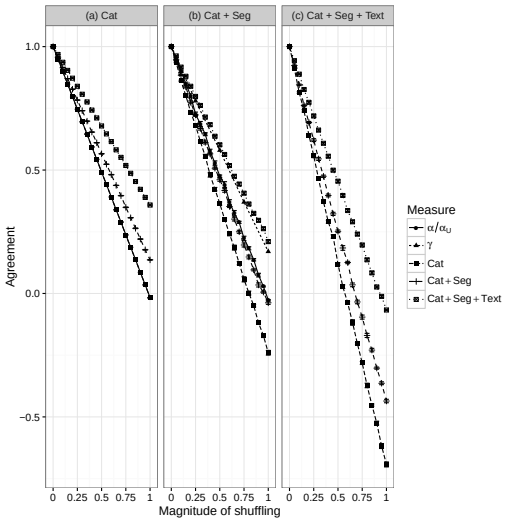

Figure 2: Evaluation results

When only the categories are shuffled, the parametrization of ${}_t\gamma$ for pure categorization covers the full range between 1 and 0, i.e., between perfect agreement and chance agreement. In this setting it behaves indistinguishably from α. When expecting errors in the transcription and segmentation, the agreement values stay above 0, reflecting the fact that the perfect agreement concerning the text and the segmentation is better than expected by chance. Consequently, the values of ${}_t\gamma$ can go below 0 in the other settings – as there are disagreements in the segmentation and/or the text not expected by chance. This differs from what Mathet et al. (2015) expect and is due to the fact that the parameters for the expected disorder calculation are not estimated from the observed annotation sets but are fixed. When expecting categorization and segmentation changes, ${}_t\gamma$ behaves similarly to α_U when categories and segments are shuffled. As expected, the original γ overestimates the amount of agreement as it does not take the pre-annotation into account.

The agreement value with settings for the expected agreement corresponding to the shuffling scenario is close to 0 when m is close to 1. The fact that it is slightly below 0 is due to the fact that $m_s = 0.1$ is slightly higher than $2 \times p_s = 0.08$.

8 Conclusion and further work

We presented text-gamma ${}_t\gamma$, a derivation of γ (Mathet et al., 2015), to measure inter-annotator agreement for categorization tasks where the annotators are allowed to change the underlying text and the segmentation during the annotation pro-

cess as it is done when annotating non-standard data that is not born-digital. The basis of our method is to align the texts using sequence alignment to create alignments of the annotations. The best of these alignments is chosen using a special dissimilarity measure. The inter-annotator agreement is measured on the basis of the mean dissimilarity between the aligned annotations. A practical point not addressed so far is that the resulting optimal alignment between the annotations can be used to show the annotators cases where they disagree and to analyze these deviations between the annotators.

For chance correction, we introduced a simple model to obtain expected disorders. To take the influence of the pre-annotation into account, our model does not model the creation of an annotation from scratch but starting with a given annotation set, random changes are applied.

Our evaluation using corpus shuffling showed that ${}_t\gamma$ reacts with a linear decrease to deviations between two annotation sets with increasing magnitude.

In its current form, ${}_t\gamma$ has some limitations. It assumes that the annotations cover the whole text as, e.g., with tokenization (with the possible exception of whitespace) and are not overlapping. While ${}_t\gamma$ is usable with annotation sets that do not cover the whole text, it is important to bear in mind that only annotations are compared. Textual changes outside of annotations have no influence on the agreement value. For non-overlapping unitizations, one possible way to take such changes into account would be to transform them into segmentations by treating gaps as annotations with the special type *gap* and ensure that gaps are not aligned with annotations of other types.

Changing the order of segments in the text is another point that ${}_t\gamma$ in its current form does not handle. This can appear, for example, when annotators disagree on the location where interlinear additions are added. The global sequence alignment used to infer possible alignments does not allow alignments between identical text segments to appear in different positions or – in other words – edges aligning annotations do not cross.

In the case of overlapping annotations of the same type, aligning annotations by inserting the annotation boundaries into the texts and aligning the text does not work as is since closing boundaries may be ambiguous in the case of overlaps.

Furthermore, our evaluation only took one type of annotation (tokens), categorization with one set of categories (POS) and two annotators into account and used a basic dissimilarity metric for nominal categories. It will be interesting to see how ${}_t\gamma$ behaves with more than two annotators, other dissimilarity metrics that take overlaps between categories into account, and with annotation sets containing multiple types of segments (e.g. *tokens* and *sentences* as in the annotation task described by Čibej et al. (2016)) and/or multiple labels for annotations (e.g. POS tag and lemma).

Regarding the chance correction, we introduced a simple model to randomly change the annotation. This model introduced some simplifications, for example, the three parts of the annotation process are modelled independently and only one edit operation is allowed for each token. Further work could introduce a more detailed model for chance correction, for example introducing further edit operations for a token with a decreasing probability.

Resources

We provide the following resources together with the paper:

(i) An implementation of the IAA measure described in the paper. The program takes CorA-XML-files, the output format of the annotation tool CorA[5], as input and outputs the IAA value and an alignment of the annotations for further analysis. It can be found at `https://github.com/fab-bar/TextGammaTool`.

(ii) An org-file[6] containing the paper and the complete code that was used to run the experiments, making the work reproducible. It can be found at `https://github.com/fab-bar/paper-KONVENS2016`.

Acknowledgements

The work of the first and second authors has been funded by the DFG. We would like to thank the anonymous reviewers for their helpful remarks and Piklu Gupta for improving our English. All remaining errors are ours.

[5] `https://www.linguistics.ruhr-uni-bochum.de/comphist/resources/cora/index.html`

[6] `http://orgmode.org/`

Primary data

Verl. Sohn *De parabell vam vorlorn Szohn*. Printed 1527 in Magdeburg by Burchard Waldis. Transcribed in the DFG-funded project "Referenzkorpus Mittelniederdeutsch / Niederrheinisch (1200 - 1650)".

References

Ron Artstein and Massimo Poesio. 2008. Inter-coder Agreement for Computational Linguistics. *Computational Linguistics*, 34(4):555–596.

Fabian Barteld, Renata Szczepaniak, and Heike Zinsmeister. 2014. The definition of tokens in relation to words and annotation tasks. In *Proceedings of the Thirteenth International Workshop on Treebanks and Linguistic Theories (TLT13)*, pages 250–258, Tübingen, Germany.

Regina Barzilay and Lillian Lee. 2002. Bootstrapping lexical choice via multiple-sequence alignment. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 164–171, Stroudsburg, PA. Association for Computational Linguistics.

Marcel Bollmann, Florian Petran, Stefanie Dipper, and Julia Krasselt. 2014. CorA: A web-based annotation tool for historical and other non-standard language data. In *Proceedings of the 8th Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities (LaTeCH) @ EACL 2014*, pages 86–90, Gothenburg, Sweden. Association for Computational Linguistics.

Maria Chatzou, Cedrik Magis, Jia-Ming Chang, Carsten Kemena, Giovanni Bussotti, Ionas Erb, and Cedric Notredame. 2015. Multiple sequence alignment modeling: methods and applications. *Briefings in Bioinformatics*, pages 1–15.

Jaka Čibej, Darja Fišer, and Tomaž Erjavec. 2016. Normalisation, Tokenisation and Sentence Segmentation of Slovene Tweets. In *Proceedings of the LREC-Workshop on Normalisation and Analysis of Social Media Texts (NormSoMe)*, pages 5–10, Portorož, Slovenia.

Jacob Cohen. 1960. A Coefficient of Agreement for Nominal Scales. *Educational and Psychological Measurement*, 20(1):37–46.

Hamish Cunningham, Diana Maynard, Kalina Bontcheva, Valentin Tablan, Niraj Aswani, Ian Roberts, Genevieve Gorrell, Adam Funk, Angus Roberts, Danica Damljanovic, Thomas Heitz, Mark A. Greenwood, Horacio Saggion, Johann Petrak, Yaoyong Li, Wim Peters, and Leon Derczynski. 2014. Developing Language Processing Components with GATE Version 8. University of Sheffield Department of Computer Science.

Stefanie Dipper, Karin Donhauser, Thomas Klein, Sonja Linde, Stefan Müller, and Klaus-Peter Wegera. 2013. HiTS: ein Tagset für historische Sprachstufen des Deutschen. *JLCL*, 28(1):1–53.

Karën Fort and Benoît Sagot. 2010. Influence of Pre-annotation on POS-tagged Corpus Development. In *Proceedings of the Fourth Linguistic Annotation Workshop, ACL 2010*, pages 56–63, Uppsala, Sweden. Association for Computational Linguistics.

Chris Fournier. 2013. Evaluating Text Segmentation using Boundary Edit Distance. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, pages 1702–1712, Sofia, Bulgaria. Association for Computational Linguistics.

Bryan Jurish and Kay-Michael Würzner. 2013. Word and Sentence Tokenization with Hidden Markov Models. *JLCL*, 28(2):61–83.

Tom Kenter, Tomaž Erjavec, Darja Fišer, and others. 2012. Lexicon construction and corpus annotation of historical language with the CoBaLT editor. In *roceedings of the 6th EACL Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities*, pages 1–6, Avignon, France. Association for Computational Linguistics.

Amit Kirschenbaum. 2013. Unsupervised Segmentation for Different Types of Morphological Processes Using Multiple Sequence Alignment. In Adrian-Horia Dediu, Carlos Martín-Vide, Ruslan Mitkov, and Bianca Truthe, editors, *Statistical Language and Speech Processing*, number 7978 in LNAI, pages 152–163. Springer, Berlin, Heidelberg.

Klaus Krippendorff. 1980. Chapter 12. In *Content Analysis: An Introduction to Its Methodology*, pages 129–154. Sage, Beverly Hills, CA, 1 edition.

Klaus Krippendorff. 1995. On the reliability of unitizing continuous data. *Sociological Methodology*, 25(47):47–76.

Klaus Krippendorff. 2011. Agreement and Information in the Reliability of Coding. *Communication Methods and Measures*, 5(2):93–112.

Klaus Krippendorff. 2013. Chapter 12. In *Content Analysis: An Introduction to Its Methodology*, pages 267–328. Sage, Thousand Oaks, CA, 3 edition.

Klaus Krippendorff. 2015. Replacement of Section 12.4 (revised version 2015.9.23). In *Content Analysis: An Introduction to Its Methodology*, pages 309–319. Sage, Thousand Oaks, CA, 3 edition. Available at http://web.asc.upenn.edu/usr/krippendorff/dogs.html (2015-08-05).

Vladimir Levenshtein. 1966. Binary codes capable of correcting deletions, insertions, and reversals. *Soviet Physics Doklady*, 10(8):707–710.

Johann-Mattis List. 2012. Multiple sequence alignment in historical linguistics. In Enrico Boone, Kathrin Linke, and Maartje Schulpen, editors, *Proceedings of ConSOLE XIX*, pages 241–260.

Christopher D. Manning and Hinrich Schütze. 1999. *Foundations of Statistical Natural Language Processing*. MIT Press, Cambridge, Massachusetts, London, England.

Yann Mathet, Antoine Widlöcher, Karën Fort, Claire François, Olivier Galibert, Cyril Grouin, Juliette Kahn, Sophie Rosset, and Pierre Zweigenbaum. 2012. Manual corpus annotation: Giving meaning to the evaluation metrics. In *Proceedings of COLING 2012: Posters*, pages 809–818, Mumbai, India.

Yann Mathet, Antoine Widlöcher, and Jean-Philippe Métivier. 2015. The Unified and Holistic Method Gamma (γ) for Inter-Annotator Agreement Measure and Alignment. *Computational Linguistics*, 41(3):437–479.

Christian M. Meyer, Margot Mieskes, Christian Stab, and Iryna Gurevych. 2014. DKPro Agreement: An Open-Source Java Library for Measuring Inter-Rater Agreement. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: System Demonstrations*, pages 105–109, Dublin, Ireland.

Ngoni Munyaradzi and Hussein Suleman. 2013. Quality Assessment in Crowdsourced Indigenous Language Transcription. In Trond Aalberg, Christos Papatheodorou, Milena Dobreva, Giannis Tsakonas, and Charles J. Farrugia, editors, *Research and Advanced Technology for Digital Libraries*, number 8092 in LNCS, pages 13–22. Springer, Berlin, Heidelberg.

Saul B. Needleman and Christian D. Wunsch. 1970. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *Journal of Molecular Biology*, 48(3):443–453.

Slav Petrov, Dipanjan Das, and Ryan McDonald. 2012. A Universal Part-of-Speech Tagset. In *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, pages 23–25, Istanbul, Turkey.

Lev Pevzner and Marti A. Hearst. 2002. A critique and improvement of an evaluation metric for text segmentation. *Computational Linguistics*, 28(1):19–36.

Jelena Prokić, Martijn Wieling, and John Nerbonne. 2009. Multiple sequence alignments in linguistics. In *Proceedings of the EACL 2009 Workshop on Language Technology and Resources for Cultural Heritage, Social Sciences, Humanities, and Education*, pages 18–25. Association for Computational Linguistics.

Tomáš Valenta, Luboš Šmídl, Jan Švec, and Daniel Soutner. 2014. Inter-Annotator Agreement on Spontaneous Czech Language. In Petr Sojka, Aleš Horák, Ivan Kopeček, and Karel Pala, editors, *Text, Speech and Dialogue*, number 8655 in LNCS, pages 390–397. Springer International Publishing, Switzerland.

Creating an extensible, levelled study corpus of Russian

Dolores Batinić
URPP Language and Space
University of Zurich
dolores.batinic@uzh.ch

Sandra Birzer
Institute of Slavonic Studies
University of Innsbruck
sandra.birzer@uibk.ac.at

Heike Zinsmeister
Institute of German Studies
University of Hamburg
heike.zinsmeister@uni-hamburg.de

Abstract

In this paper, we present first results of training a classifier for discriminating Russian texts into different levels of difficulty. For the classification we considered both surface-oriented features adopted from readability assessments and more linguistically informed, positional features to classify texts into two levels of difficulty. This text classification is the main focus of our Levelled Study Corpus of Russian (LeStCoR), in which we aim to build a corpus adapted for language learning purposes – selecting simpler texts for beginner second language learners and more complex texts for advanced learners. The most discriminative feature in our pilot study was a lexical feature that approximates accessibility of the vocabulary by the second language learner in terms of the proportion of *familiar* words in the texts. The best feature setting achieved an accuracy of 0.91 on a pilot corpus of 209 texts.

1 Introduction

Selecting texts of an appropriate difficulty level is a challenging task for both teachers of a second language (L2) as well as the learners themselves. This becomes particularly evident when learners are working with linguistic corpora which is part of many foreign language studies in the digital age (Römer, 2008; Steinbach and Birzer, 2011): Linguistic corpora do not normally differentiate between texts suitable for beginner and more advanced L2 learners.

One way to deal with text selection for L2 learning purposes is simplifying texts (Karpov and Sibirtseva, 2014; Vajjala and Meurers, 2014), another one is compiling texts selected for different proficiency levels as an additional resource for learners especially on a beginner and intermediate level (Cobb, 2007; Allan, 2009). This paper contributes to the second line of research. In this paper, we introduce our concept for creating a Levelled Study Corpus of Russian (LeStCoR) stratified into texts suitable for L2 learners of different proficiency levels. While the sampling and creation of LeStCoR is still work in progress, we will mainly focus on one aspect: the method of automatically classifying Russian texts according to the difficulty they pose for L2 learners. Since our goal is to provide an extensible study corpus of Russian, we need a tool that supports the classification of new texts in an efficient and consistent way. To this end, we train a classifier on manually labelled texts and use surface as well as linguistically motivated features to discriminate between simple (Class I) and more difficult texts (Class II).

It is important to note that in our approach automatic classification is used by the corpus creator – not the learners themselves – to identify texts with an appropriate difficulty level for integrating them into the corpus. The classification is seen as a preprocessing step followed by additional manual checking if deemed necessary. This means that the classification is performed 'behind the scenes' in terms of Aston (2000). It is not offered 'on stage' as a method for learners to identify appropriate texts by themselves (Vajjala and Meurers, 2013).

The paper is structured as follows. In Section 2, we introduce related work on classifying texts automatically according to their difficulty. Section 3 describes the target text selection. In Section 4, we introduce characteristics that are indicative for text difficulty and detail how we operationalized them as features. Section 5 describes the actual feature selection. In Section 6, we evaluate our approach by a pilot study performed on 209 texts that demonstrates the applicability of the classification method. We close with a discussion of the results and further work.

2 Related Work

There is a long tradition of assessing the difficulty of a text in terms of surface-oriented readability measures that allow the researchers to compare different texts in an objective way (see Dubay (2004) for a historical overview). In addition to the classical surface-oriented measures that mainly take simple word counts, word and sentence lengths etc. into account, other approaches integrate lexical, syntactic, and discourse features that address the lexical coverage of a text, its parts of speech, syntactic structures, and cross-sentential features like the referential overlap and relations between clauses triggered by discourse connectives (McNamara et al., 2014; Napolitano et al., 2015). Machine learning approaches make use of the fact that different measures quantify different aspects of the text difficulty characteristics (Schwarm and Ostendorf, 2005; Vajjala and Meurers, 2012; Karpov and Sibirtseva, 2014). Many related works focused on establishing the level of text difficulty for native speakers (Collins-Thompson and Callan, 2004; Schwarm and Ostendorf, 2005; Feng et al., 2010). However, studies of the difficulty level for L2 learners have also been conducted recently, with the underlying hypothesis that text comprehensibility is perceived very differently by L2 learners (François, 2014; Heilman et al., 2007; Xia et al., 2016).

3 Compilation of a seed corpus

LeStCoR is intended to grow over time by being extended with new texts. For the pilot study on text classification, we selected 209 texts from the Test of Russian as a Foreign Language (TORFL, Russian: TRKI) reading and listening tasks. The pilot corpus is stratified into two classes: Class I contains 136 texts that belong to beginners' or lower intermediate levels (TRKI levels elementary, basis and level 1), whereas Class II contains the other 73 texts of intermediate or advanced levels (TRKI levels 2, 3 and 4). Table 1 gives an overview of the text distribution across the TRKI levels and our text difficulty classes (I & II). We also provide the corresponding levels of the Common European Framework of References for Languages (CEFR) for comparison.

As shown in Table 1, the distribution of texts per class was not homogeneous, since we were able to provide more texts for Class I than for Class II. Some of the texts needed to be OCRed and manually corrected. All texts were part-of-speech tagged and lemmatized with TreeTagger (Schmid, 1994) using parameter files trained on the disambiguated version of the Russian National Corpus (Plungian, 2005; Plungian et al., 2009; Sharoff et al., 2008).

Class	TRKI	CEFR	Sem	#Texts
I	elementary	A1	1st	43
	basis	A2	2nd	43
	1	B1	2nd	50
II	2	B2	3rd	38
	3	C1	4th	30
	4	C2	indep	5

Table 1: TRKI proficiency levels and sampling of the pilot corpus (*#Text*: number of texts, *Class*: simple vs. difficult texts; *Sem*: Semester, *indep*: semester independent).

4 Candidates for features

In this pilot study, we mainly focused on surface features that are employed in traditional readability scores and linguistically motivated token-related lexical and morphosyntactic features. For the linguistic features we tested to what extent the proportion of 'familiar' words, the proportion of 'abstract' words and the proportion of different parts of speech in text may be indicative of the text difficulty.

Average readability score. For calculating the readability scores, we adapted the Python implementation of existing readability measures by Rik Goldman[1] to Russian and calculated an average grade score based on seven common measures (for an overview of most scores see DuBay (2004); the Coleman Liau Index Score is described in Coleman and Liau (1975)):[2]

- Flesch-Kincaid Grade Level
- Coleman Liau Index Score
- (Gunning) Fog
- SMOG Index
- Automated Readability Index
- New Dale Chall Adjusted Grade Level[3]
- Powers-Sumner-Kearl Grade Level

[1] Goldman's implementation: `https://github.com/ghoulmann/py-readability-statistics`.

[2] A demo-version of our text difficulty calculator can be accessed at `http://www.lestcor.com/`.

[3] Calculating the New Dale Chall Adjusted Grade Level makes use of the concept of *hard words*. For English this is done by counting words in text not belonging to the Dale Chall list of 3,000 frequent English words. In our adaptation to Russian, we defined 'hard words' in Russian texts as those having four or more syllables.

Readability scores can be interpreted as an estimation of the number of years of education a person has had. An average readability score of 5 indicates that the given text should be easily comprehensible for a fifth-grade student, whereas a score higher than 15 means that the text is best suited for college graduates.

Familiar words. This feature operationalizes the accessibility of the vocabulary by L2 learners. It measures how much of the text is covered by core vocabulary and other words that are easy to grasp by an adult learner. As *core vocabulary* we used the list of 5,000 most frequent Russian lemmas compiled by Sharoff (2002). A core vocabulary of 5,000 most frequent words is expected to enable the learner to understand about 80% of a text (Hiebert and Kamil, 2005). In addition, as familiar words we also considered numerals, proper names, pronouns, and internationalisms. The latter are treated as familiar words because adult learners of Russian may easily understand them without being familiar with the Russian vocabulary itself. Some examples are бокс 'box', бейсбол 'baseball', and телефон 'telephone'. The list of internationalisms was gathered from Wikipedia's list of internationalisms in the Russian language. We assumed that a high proportion of familiar words was indicative for texts with low difficulty.

Abstract words. We calculated the average occurrence of abstract words in sentences by counting the words in a text having typical abstract word endings, such as -изм '-ism', -ость '-ness', -ство '-ship', -ота '-ness', -ание / -ение (markers of nominalized verbs) and dividing it by the total number of sentences in a text. We also experimented with the proportion of abstract words in the whole text. We assumed that abstract words occurred more frequently in sentences from higher classes. We did not discriminate between internationalisms and abstract words so that there is a certain overlap and potential correlation.

Parts of speech. In order to verify if there is a prevalence of a particular part of speech in sentences of Class I and Class II, we considered the average occurrences of nouns, verbs, pronouns, adjectives, adverbs, adpositions, conjunctions, and particles. Relying on the study conducted by Feng et al. (2010), we expected nouns to have a higher predictive power than other parts of speech.

Syntactic and discourse features. With the idea that they could be discriminative for difficult texts, we studied the distribution of adverbial participles, perfect participles, and marking of conditional (чтобы 'in order to').

Content words. We calculated the proportion of nouns, adjectives, verbs, and adverbs in texts. We assumed that a high proportion of content words may be a good indicator of text difficulty: We expected that the more content words per text, the more difficult the text.

Type/token ratio. We calculated the ratio of unique words in texts (types) to the total number of word occurrences (tokens) in texts. A low ratio would indicate a more difficult text due to a high number of different words.

5 Feature selection

Before selecting the actual feature combinations for the classifier, we observed the differences in their distributions within texts of Class I and Class II. As shown in Figure 1, the average proportion of familiar words in texts of Class I differed from the one in Class II (an average text in Class I contained 94% of familiar words, whereas an average text from Class II contained 83% of familiar words). A difference in the two classes was also considerable for the features average readability (per text). Figure 2 shows that the average absolute frequencies of abstract words and nouns per sentence were also discriminative, followed by adjectives and adpositions. In order to find thresholds which would discriminate between Class I and Class II, we first calculated the average distribution of a given feature for each class. Then we experimented with the classification model by setting initially the two averages as thresholds and incrementing/reducing them until we reached the highest accuracy for the given model. We also investigated different readability measures and found that Flesch-Kincaid Score seemed to discriminate between the two groups more strongly than other readability measures, so we used it as a separate feature as well. The proportion of content words and type/token ratio did not prove to be discriminative for Class I and Class II. The same applies to our syntactic and discourse features, which were too infrequent in the selected TRKI texts to play a role in the classification process (for instance, the conditional marker чтобы 'in order to' occurs only four times in Class I and five times in Class II).

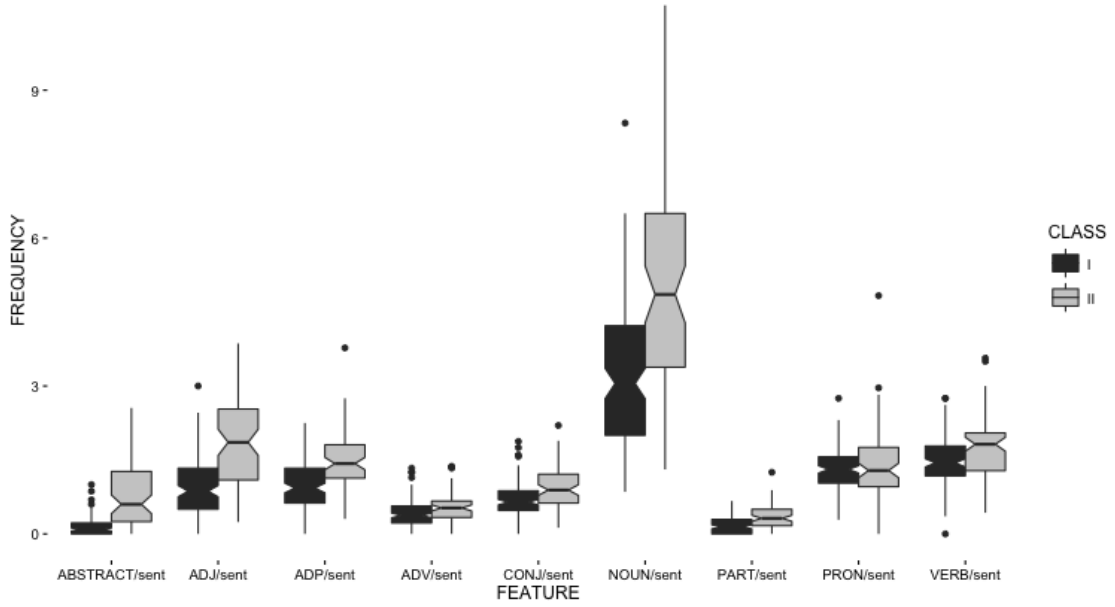

Figure 2: Boxplots for the absolute frequencies of abstract words and different parts of speech (per sentence) in Class I & II. Notches indicate medians and their 95% confidence intervals; dots mark outliers. (Created with R's ggplot2 package).

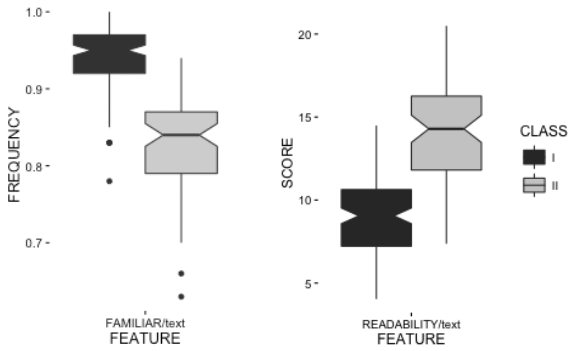

Figure 1: Boxplots for the relative frequencies of familiar words (to the left) and readability scores (to the right) per text in Class I & II.

6 Results and Discussion

We performed a classification with Naive Bayes (NLTK[4], Bird et al., (2009)) and 10-fold cross validation. As a baseline we assumed that the classifier would (randomly always) assign Class I which would result in 65% of the texts being correctly classified on average (136/209). The classifier achieved an accuracy of 0.91 by predicting the text difficulty level by combining the features average readability, familiar words, abstract words, nouns and adjectives. Contrary to our expectations, the average readability score alone did not prove to be discriminative enough (accuracy of 0.64). However, the models that combined average readability with other features reached an accuracy between 0.89 and 0.91 (see M3-M7 in Table 2). Familiar words were highly informative even as a separate feature: When setting the threshold of $> 90\%$ of familiar words per text, the model reached the accuracy of 0.84. This finding suggests that building a two-levelled corpus may be done in a relatively accurate way by using a simple feature such as the proportion of familiar words as basis and extending it with readability scores and more linguistically motivated features.

A low predictability power of the feature average readability score can be related to several factors. Firstly, the average of seven different readability measures smooths the difference between classes which is observed when dealing with particular readability measures separately. For instance, the average Powers-Sumner-Kearl Grade Level for Class II is 9.9, whereas the average Flesch-Kincaid Score for Class II amounts to 18.4. Secondly, different readability measures serve different purposes; for instance, Powers-Sumner-Kearl Grade Level is generally used for children under 10 years. Lastly, for lack of resources we only had five texts repre-

[4]NLTK: `http://www.nltk.org/`.

		Models						
Feature	Threshold	M1	M2	M3	M4	M5	M6	M7
Flesch-Kincaid score	> 19						x	
	< 9							
Average readability	> 15			x	x			
	> 12	x				x	x	x
#Familiar words	$< 80\%$ / t			x	x	x	x	x
	$> 90\%$ / t		x	x	x	x	x	x
#Abstract words	$> 8\%$ / s				x	x	x	x
	$< 2\%$ / s						x	
#Nouns	$> 60\%$ / s					x	x	x
	$< 20\%$ / s					x	x	x
#Adjectives	$> 16\%$ / s					x	x	x
	$< 5\%$ / s					x	x	x
#Adpositions	$> 20\%$ / s					x		
Mean accuracy		.64	.84	.89	.89	.89	.90	.91
sd		$\pm$ .10	$\pm$.08	$\pm$.05	$\pm$.07	$\pm$.05	$\pm$.06	$\pm$.06

Table 2: Classification results with different feature selections. According to a two sample t-test, the accuracies of M2-M7 are significantly different from the ones of M1; M7 differs from M2 with an error probability of $p = 0.05864$.

senting the level TRKI 4. Other texts of this level would presumably have had high average readability scores, which would in consequence ameliorate the prediction strength of this variable.

The proportion of *familiar* words, though, proved to be a well-suited predictor for discriminating between simple and difficult texts for L2 learners. This is likely due to the fact that *familiar* words included not only frequent words, but also numbers, pronouns, internationalisms and named entities, which, although they might still be incomprehensible or difficult to read for L2 learners, they do not compromise their comprehension of the text as a whole. Moreover, a list of the 5,000 most frequent Russian lemmas proved to be a suitable amount of words to use as a threshold for discriminating between texts below and above CEFR's B2 level, corresponding to TRKI 2.

In further work, we plan to work with the core vocabulary for all TRKI levels separately, instead of using the top word frequency list of 5,000 lemmas as a threshold between simple and difficult vocabulary. Once we provide some more text material, we are also planning to include more linguistically motivated features, such as discourse markers and syntactic markers as well as semantic features, such as the proportion of academic vocabulary words (Vajjala and Meurers, 2012; Vajjala and Meurers, 2014). Moreover, we are considering using a language-modelling approach (Collins-Thompson and Callan, 2004), which may be well suited for an extensible corpus.

7 Conclusion

We performed a text classification study to classify original, non-adapted Russian texts into two levels of difficulty for L2 learners. The trained classifier beat the baseline and achieved an average accuracy of 0.91 with surface-oriented features complemented by vocabulary-based features including part of speech information. The list of most frequent Russian words extended with named entities, numbers, pronouns and internationalisms proved to be the best suited predictor for text difficulty classification aimed to L2 learners. More linguistically-motivated features like syntactic and discourse features did not improve the classification results but we expect more conclusive results on a larger training base.

Acknowledgments

We would like to thank the anonymous reviewers for their very helpful remarks and Piklu Gupta for improving our English. All remaining errors are ours.

References

Rachel Allan. 2009. Can a graded reader corpus provide 'authentic' input? *ELT journal*, 63(1):23–32.

Guy Aston. 2000. Learning English with the British National Corpus. In M.P. Battaner and C. L'opez, editors, *VI jornada de corpus lingüístics*, pages 15–40, Barcelona.

Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural language processing with Python*. O'Reilly Media, Inc.

Tom Cobb. 2007. Computing the vocabulary demands of L2 reading. *Language Learning & Technology*, 11(3):38–63.

Meri Coleman and Ta Lin Liau. 1975. A computer readability formula designed for machine scoring. *Journal of Applied Psychology*, 60(2):283–284.

Kevyn Collins-Thompson and James P Callan. 2004. A language modeling approach to predicting reading difficulty. In *HLT-NAACL 2004: Main Proceedings*, pages 193–200, Boston, MA.

William H DuBay. 2004. The Principles of Readability. *Online Submission*: `http://files.eric.ed.gov/fulltext/ED490073.pdf`.

Lijun Feng, Martin Jansche, Matt Huenerfauth, and Noémie Elhadad. 2010. A comparison of features for automatic readability assessment. In *Proceedings of Coling 2010: Posters*, pages 276–284, Beijing, China.

Thomas François. 2014. An analysis of a French as a foreign language corpus for readability assessment. In *Proceedings of the 3rd workshop on NLP for computer-assisted language learning at SLTC 2014*, Uppsala University.

Michael Heilman, Kevyn Collins-Thompson, Jamie Callan, and Maxine Eskenazi. 2007. Combining lexical and grammatical features to improve readability measures for first and second language texts. In *Proceedings of HLT-NAACL 2007*, pages 460–467, Rochester, New York.

Elfrieda H. Hiebert and Michael L. Kamil. 2005. *Teaching and Learning Vocabulary: Bringing Research to Practice*. Lawrence Erlbaum Associates, Mahwah, New Jersey.

Nikolai Karpov and Vera Sibirtseva. 2014. Towards automatic text adaptation in Russian. *Higher School of Economics Research Paper No. WP BRP*, 16.

Danielle S McNamara, Arthur C Graesser, Philip M McCarthy, and Zhiqiang Cai. 2014. *Automated evaluation of text and discourse with Coh-Metrix*. Cambridge University Press.

Diane Napolitano, Kathleen Sheehan, and Robert Mundkowsky. 2015. Online readability and text complexity analysis with TextEvaluator. In *Proceedings of NAACL 2015: Demonstrations*, pages 96–100, Denver, Colorado.

Vladimir A. Plungian, Ekaterina V. Rakhilina, and Tatjana I. Reznikova, editors. 2009. *Nacional'nyj korpus russkogo jazyka: 2006-2008. Novye rezul'taty i perspektivy*. Nestor-Istorja, St. Petersburg.

Vladimir A. Plungian, editor. 2005. *Nacional'nyj korpus russkogo jazyka: 2003-2005. Rezul'taty i perspektivy*. Indrik, Moscow.

Ute Römer. 2008. Corpora and language teaching. In Anke Lüdeling and Merja Kytö, editors, *Corpus linguistics: An International Handbook*, Handbücher zur Sprache und Kommunikationswissenschaft. Volume 1, pages 112–130. Mouton de Gruyter, Berlin.

Helmut Schmid. 1994. Probabilistic Part-of-Speech Tagging Using Decision Trees. In *Proceedings of International Conference on New Methods in Language Processing*, Manchester, UK.

Sarah Schwarm and Mari Ostendorf. 2005. Reading level assessment using support vector machines and statistical language models. In *Proceedings of ACL 2005*, pages 523–530, Ann Arbor, Michigan.

Serge Sharoff, Mikhail Kopotev, Tomaz Erjavec, Anna Feldman, and Dagmar Divjak. 2008. Designing and evaluating Russian tagsets. In *Proceedings of LREC 2008*, pages 279–285, Marrakech, Marrocco.

Serge Sharoff. 2002. Meaning as use: Exploitation of aligned corpora for the contrastive study of lexical semantics. In *Proceedings of LREC 2002*, Las Palmas, Spain.

Andrea Steinbach and Sandra Birzer. 2011. Authentisches Sprachmaterial schnell gefunden. Das Potenzial russischer Textkorpora im Russischunterricht. *Praxis Fremdsprachenunterricht*, (2):7–10.

Sowmya Vajjala and Detmar Meurers. 2012. On improving the accuracy of readability classification using insights from second language acquisition. In *Proceedings of the Seventh Workshop on Building Educational Applications Using NLP*, pages 163–173, Montréal, Canada.

Sowmya Vajjala and Detmar Meurers. 2013. On the applicability of readability models to web texts. In *Proceedings of the 2nd Workshop on Predicting and Improving Text Readability for Target Reader Populations*, pages 59–68.

Sowmya Vajjala and Detmar Meurers. 2014. Assessing the relative reading level of sentence pairs for text simplification. In *Procedings of EACL 2014*, pages 288–297, Gothenburg, Sweden.

Menglin Xia, Ekaterina Kochmar, and Ted Briscoe. 2016. Text readability assessment for second language learners. In *Proceedings of the 11th Workshop on Innovative Use of NLP for Building Educational Applications*, pages 12–22, San Diego, CA.

Morphological analysis and lemmatization for Swiss German using weighted transducers

Reto Baumgartner
University of Zurich
retoflavio.baumgartner@uzh.ch

Abstract

With written Swiss German becoming more popular in everyday use, it has become a target for text processing. The absence of a standard orthography and the variety of dialects, however, lead to a vast variation in different spellings which makes this task difficult. We built a system based on weighted transducers that recognizes over 90% of the tokens in certain texts. Weights ensure preferring the best analysis for most words while at the same time allowing for very broad range of spelling variations. Our morphological tagset that we defined for this purpose and lemmas in Standard German open the possibility for further processing. Besides our morphological analyzer and lemmatizer, a morphologically annotated corpus offers new resources for Swiss German and helps spreading our tagset.

1 Introduction

With an increased use of written text in Swiss German (SwG), there is a growing interest in tools to process these texts. SwG dialects are spoken by more than 4 million people in Switzerland in everyday life around the centers Zurich, Basel and Bern whose dialects we covered in our system at this stage. In writing usually Standard German (StG) is preferred but for private communication many people use their SwG dialect.

SwG differs from StG in phonology, vocabulary and grammar. Its vowel system still resembles that of Middle High German (MHG) with *Ziit* "time" and *Huus* "house" (MHG *zît* and *hûs*; StG *Zeit* and *Haus*) while the differences in the consonant system and the loss of endings are more modern traits (Christen et al., 2012, p. 27). Over time SwG has lost its genitive case and the past preterite (Siebenhaar and Wyler, 1997, p. 37). Conversely it possesses infinitive particles that are not known to StG.

SwG consists of different local dialects that mainly differ in phonology and to a lesser extent in vocabulary. There is no standard orthography, but there are proposals for sound-character assignment like *Dieth-Schreibung* (Dieth, 1986) or *Bärndütschi Schrybwys* (Marti, 1985a) that are, however, not known to everyone. This results in a high variability of spellings influenced by both dialects and personal writing preferences. As an example for the StG word *Jahr* "year", we found in our corpus *Jahr* and *Jaar*, *Johr* and *Joor*, even *Joh* for different pronunciations and spelling preferences.

The lack of a standard orthography and the vastness of variants motivate the choices that have to be made to process these dialects. For lemmatization we use StG words. The variants can probably best be dealt with using finite-state technology that do not rely on huge corpora but on linguistic engineering. Weighted transducers can be used for a better trade-off between good coverage and overgeneration.

2 Related Work

The increase of SwG in writing led to a number of resources:

Corpora: By now two corpora consisting of everyday written language have been collected. The *Swiss SMS Corpus* (Stark et al., 2009 2015) counts 275 000 tokens in SwG from short messages. The corpus includes manually made glosses in StG. *NOAH's Corpus of Swiss German Dialects* (Hollenstein and Aepli, 2014) counts 115 000 tokens in SwG from different sources like blogs, wikipedia entries, literature, newspapers or a business report. The corpus has manually been annotated with parts of speech. With *Archimob – A corpus of Spoken Swiss German* (Samardžić et al., 2016), there is a corpus of transcribed spoken SwG, opposed to the others whose material was written first.

Taggers: Hollenstein and Aepli (2014) trained a Part-of-Speech tagger model on their collected data that reaches an accuracy of 90.62%.

Morphological generation: A closely related task to ours is morphology generation. An approach from Scherrer (2011) uses replacement rules and information about the dialects' location to build SwG word forms. As this system follows specific spelling guidelines for consistency, it is not suited for analysis where it is important to recognize a broad range of different spellings.

3 Annotation scheme

3.1 Parts of speech

As both the *Swiss SMS Corpus* and *NOAH's Corpus* make use of the *Stuttgart–Tübingen–TagSet* (STTS) (Schiller et al., 1999), we chose the same tagset for our parts of speech. As it was developed for StG we had to make some changes for use with SwG:

Changed use: Some tags had to be opened to different words with the same function. The use of *wo* "where" as relative pronoun (*PRELS*) or as subordinating conjunction (*KOUS*) like StG *als* "when" demands the expanded use of these tags. Similarly *für* "for" and *zum* "to the" can now be conjunctions that govern an infinitive (*KOUI*).

In contrast to StG, prepositions can be combined with any article. In consequence *APPRART* is also applicable for plural forms as in *id* "into the" or indefinite articles as in *ime* "in a".

Lacking a corresponding form, the tag *PRELAT* for attributive relative pronouns will not be used.

Additions: For infinitive particles like *go* or *cho* we decided to use the tag *PTKINF* like in the *Swiss SMS Corpus* and in *NOAH's Corpus*.

For merged words like *hets* "there is" (literally "has it") we copied the treatment from Hollenstein and Aepli (2014) with the plus sign. *hets* is therefore tagged with *VAFIN+PPER*. Unlike in their Part-of-Speech tagging task, for our morphological analysis task all tags must be kept.

A completely new tag is *PTKAM* for the particle *am* (literally "at the") in the progressive verb form. In StG examples like *Ich bin am Schreiben* literally "I am at-the writing", *Schreiben* is commonly analyzed as a substantified verb forming a prepositional phrase together with *am*. In SwG this construction is expanded with verbal objects more often than in most areas outside Switzerland (Van Pottelberge, 2005). Such an example would be *Ich bi en Brief am schriibe* literally "I am a letter at-the writing". The fact that *am* here stands between the object and the infinitive makes an analysis as prepositional phrase impossible and speaks against the tag *APPRART* for *am*. The comparison with *en Brief z schribe* "to write a letter" with the particle *z* is a good argument for *am* to be analyzed as a particle too. Our tag would also make sense for other varieties of the German language where such constructions occur or where their interpretation as verbal forms are preferred over one as preposition–noun sequences.

3.2 Morphological features

Due to the absence of an established morphological tagset for SwG, we defined a character based tagset that extends the STTS to *STTS.gsw*. The characters that make up the tags are listed in table 1.

Category	Tags
Degree	**p** (positive), **c** (comparative), **s** (superlative)
Person	**1** (first), **2** (second), **3** (third)
Case	**n** (nom.), **a** (acc.), **d** (dat.), **r** (nom./acc.)
Number	**s** (singular), **p** (plural)
Gender	**m** (masc.), **f** (fem.), **n** (neutral)
Mode	**i** (indicative), **j** (subjunctive I), **k** (subjunctive II)
Inflection	**s** (strong), **w** (weak)
Definiteness	**i** (indefinite), **d** (definite)

Table 1: Morphological tags.

We decided against a tag for the mixed adjective inflection that is used by many descriptions of the StG language. The reasons behind this are that this distinction is solely syntactic and that different SwG dialects use the strong and weak inflection differently.

As there is no past preterite, the category *time* could be spared. In consequence the two subjunctive tenses are interpreted as different modes (as *subjunctive I* and *II* instead of *subjunctive present* and *preterite*).

We introduced a shared tag for nominative or accusative cases even though this would constitute a large intervention from a linguistic perspective. As only personal and some related pronouns make a distinction between these cases, different tags for these forms would lead to competing analyzes that could only be distinguished through semantics. Therefore we exclude the task of disam-

biguating these cases but mark this with the tag *r* (from *rectus*).

In our example *hets*, the tag *VAFIN* is extended with *3si* (3rd person, singular, indicative) and *PPER* is extended with *3snn* (3rd person, singular, neutral, nominative).

3.3 Lemmas

For the choice of lemmas we decided to follow the rules from the *Swiss SMS Corpus* to ensure compatibility between different resources for SwG. Their main principles are that closely related words must be used, no new StG words must be invented and that the meaning should not be changed (Ueberwasser, 2013).

For example *hets* is annotated as *haben/VAFIN.3si+es/PPER.3snn* after including the morphological tags and lemmas.

4 Material

4.1 Corpus

For the calculation of the weights and for testing we annotated two sets of around 14 500 tokens taken from *NOAH's Corpus* using the morphology analysis tool in its development stage and selecting or adding the correct analysis.

4.2 Standard German resources

To avoid having to collect word stems and classifying them by inflection class, we took the allomorph list from *Morphisto* (Zielinski et al., 2009). Our material taken from this source counts 7833 nouns, 4300 verbs, 3178 adjectives, 1052 proper nouns and 781 adverbs. We used the lemma stem for our lemmas and the allomorph stems for later converting to SwG sounds. The inflection classes enable us to select the right endings in SwG and the word frequency classes are used as base for the weights of our tool.

With this connection to StG, the selection of stems can easily be changed without the need to collect more SwG stems and the lemmas are consistent with the resources used in this task.

For words from other parts of speech we had to take the frequency class from the *DeReWo* list from IDS (2012).

4.3 Swiss German resources

The forms of the closed word classes like pronouns, particles and similar were added with consulting dialect grammars from Weber (1948), Marti (1985b) and Suter (1992). In addition we added 11 for adjectives plus ordinal numbers (as *ADJA*), 127 adverb stems, about 50 noun stems and around 90 full verb stems (21 roots plus different prefixes) that cannot easily be derifed from StG forms.

5 Implementation

Our system is intended to be run with the *Helsinki Finite-State Transducer Technology* (HFST) (Lindén et al., 2009). HFST allows building and applying weighted finite-state transducers with tropical semi-rings. That means paths can be punished with weights that are added on the way and the paths with the lowest weights are preferred.

5.1 Forms

The implementation of the SwG word forms happens in two stages. The first stage is producing a hidden layer that represents the phonemes of different dialects. For open word classes like nouns or verbs we use replacement rules that we apply on the stems from *Morphisto*. For example *Zeit* "time" has to be converted to `zīt` while *heiß* "hot" will become `haĭss`. Those different replacements (see figure 1) for ⟨ei⟩ will be weighted by their probability, including phonological context as far as possible.

```
# ei before er
define EI1 [ {eier} (->) {īr}::0.1 ];
define EI2 [ {eier} (->) {īĕr}::4.7 ];
define EI3 [ {eier} -> {aĭĕr}::5.4 ];
# ei else
define EI4 [ {ei} (->) {ī}::0.9 ];
define EI5 [ {ei} (->) {ĩ}::5.4 ];
define EI6 [ {ei} -> {aĭ}::1.1 ];
# combined rule for ei
define EI [ EI1 .o.  EI2 .o.  EI3 .o.
EI4 .o.  EI5 .o.  EI6 ] ;
```

Figure 1: Replacement rules for ⟨ei⟩. First EI1–EI3 deal with ⟨ei⟩ before ⟨er⟩, then EI4–EI6 replace ⟨ei⟩ in all other cases. Higher weights indicate less frequent options.

For the closed word classes and words that do not exist in StG like *gheie* "to drop" we wrote the forms directly in phonemes.

In the second stage these phonemes are replaced by dialect specific spellings using a different set of rules for every dialect. Here we limited using weights to specific sound changes that are not represented by the chosen phonemes, as in most cases

list	phonemes	dialects	translation
Zeit	zīt	Ziit, zyt	“time”
Zeit-e	zītĕ	ziite, Zytä	“times”
ghĩĕ		gheie, ghie	“[I] drop”
ghĩt		gheit, ghiit	“[it] drops”
mīn		min, miin	“my”
mīnĕ		myne, minä	“mine”

Table 2: Form generation for open classes over a step in-between and for exclusively SwG words and closed classes directly in phonemes.

the different results are just different spellings for the same sounds. So far we made dialect modules for Basel, Bern and Zurich. Table 2 gives some examples how the different forms are generated.

Clitics are added between these steps and flag diacritics – a feature offered by HFST – are used to disable ungrammatical combinations.

5.2 Weights

The way the weights are calculated is motivated by the tropical semi-ring and the word frequency classes. For every competing alternative at a decision (e. g. in replacement rules), the the absolute value of the binary logarithm of the probability is added to the weight. Using this formulae all the weights are in the same currency and the different reasons for weights can be treated the same.

6 Results

6.1 Coverage

For 79% of the tokens in the test corpus, our system could produce the correct analysis according to their positions. With exclusion of foreign language material (*FM*), named entities (*NE*) and non-words (*XY*) this quota reached 86%.

In the blog data we could even observe that 90.8% of the tokens (without *FM*, *NE* and *XY*) could be reached. On the other side, the business report and wikipedia entries proved to be more difficult with 81.7% resp. 81.9%.

Table 3 shows the coverage for all tokens and some selected parts of speech. The closed word classes like negation particles (*PTKNEG*), indefinite pronouns (*PIDAT*) or interrogative pronouns (*PWS*) are fully covered. The open word classes like named entities (*NE*), nouns (*NN*) and adjectives (*ADJA*) are more difficult. While it was not the goal to include a lot of named entities, the nouns are

Parts of speech	correct
all	0.790
w/o FM, NE, XY	0.860
NN	0.583
ART	**0.970**
NE	*0.129*
APPR	**0.959**
VAFIN	**0.980**
ADJA	0.662
APPRART	**0.970**
KON	**0.992**
VVPP	0.851
VVFIN	0.881
PTKNEG	**1.000**
PIDAT	**1.000**
PWS	**1.000**

Table 3: Coverage of all tokens in the test corpus, the most frequent PoS and some selected PoS.

an open problem. Like most Germanic languages, SwG allows building a theoretically unlimited number of compounds which were hard to grasp and which shows in the low coverage in our system. Similarly also adjectives can be derived from other word classes. The most frequent case of this type proofed to be participles that had been turned into adjectives and declined accordingly.

6.2 Weights

Evaluating weights in a group of non-stanardized dialects is difficult because different speakers might not agree on what analyses are acceptable or not. Hence we chose a purely data driven approach which compares the ranking be our system with a random order of analyses using the *mean reciprocal rank* (MRR) (Büttcher et al., 2010, p. 409) and (Neumann, 2010, p. 587). The MRR averages the multiplicative inverse of the rank of the first correct solution for all evaluated tokens. To reduce the impact from uncovered words, we only looked at them where the correct solution is provided by the system.

The overall MRR of 0.843 by the system compared to the 0.531 for random orders shows that the weights successfully order the analyses.

Besides the overall MRR, table 4 shows that both open word classes like verbal participles (*VVPP*) and closed word classes like cardinal numbers *CARD* profit from the weights. Parts of speech like infinitives with *zu* “to” (*VVIZU*) could also do

Parts of speech	system	random
all	**0.843**	0.531
w/o FM, NE, XY	**0.842**	0.530
NN	**0.911**	0.595
ART	**0.721**	0.386
NE	**0.957**	0.783
APPR	**0.759**	0.375
VAFIN	**0.931**	0.450
ADJA	**0.439**	0.324
APPRART	**0.792**	0.423
KON	**0.908**	0.474
VVPP	**0.983**	0.645
VVFIN	**0.708**	0.349
CARD	**0.993**	0.886
VVIZU	1.000	1.000
PTKZU	*0.348*	0.587
PTKA	*0.417*	0.459

Table 4: Mean reciprocal rank on the correctly analyzed tokens, for all PoS, the most frequent and for some selected PoS. The random numbers set the baseline.

without weights.

On the other hand particles before infinitives (*PTKZU*) or before adjectives (*PTKA*) even suffer from weights. The cause there is that they are beaten by more frequent prepositions of the same form and are thus always on a deeper rank.

A small profit can be seen with adjectives (*ADJA*). There a large number of analyses for certain types pull down the MRR. For example *schööni* "beautiful" there can be found up to 5 valid analyses (out of 24).

For these problems with particles and adjectives, a word based procedure cannot solve the problem. However, with a language model the problem of competing analyses should be solved easily.

7 Conclusion

With a token coverage of the treated parts of speech of 86% up to 90% on selected texts our system clearly can help with the production of annotated resources for the SwG dialects.

An open problem is still the low coverage on nouns due to large potential to build new words. Enabling composition and derivation is a possible answer to this problem. For words unknown due to the lack of corresponding StG words, adding more stems seems the best way.

For the future we see much potential in language models to be able to distinguish between competing analyses. For this task our corpus can be used as training data. Experiments will have to decide if the weights can be used as emission models.

Another task for the future is the expansion of the program to process more dialects. Especially the alpine dialects differ from those covered here and could profit from this.

Acknowledgments

I would like to thank Simon Clematide from University of Zurich for his help and valuable input during this project. Also, I am thankful to Noëmi Aepli for explanations about the tagging in NOAH's corpus.

References

Stefan Büttcher, Charles L. A. Clarke, and Gordon V. Cormack. 2010. *Information Retrieval - Implementing and Evaluating Search Engines*. MIT Press, Cambridge MA, USA.

Helen Christen, Elvira Glaser, and Matthias Friedli, editors. 2012. *Kleiner Sprachatlas der deutschen Schweiz*. Huber, Frauenfeld, Switzerland, 4th edition.

Eugen Dieth. 1986. *Schwyzertütschi Dialäktschrift: Dieth-Schreibung*. Lebendige Mundart. Sauerländer, Aarau etc., Switzerland, 2nd edition.

Nora Hollenstein and Noëmi Aepli. 2014. Compilation of a Swiss German dialect corpus and its application to PoS tagging. In Marcos Zampieri, Liling Tan, Nikola Ljubešić, and Jörg Tiedemann, editors, *COLING 2014, Proceedings of the First Workshop on Applying NLP Tools to Similar Languages, Varieties and Dialects*, pages 85–94.

IDS Institut für Deutsche Sprache, Programmbereich Korpuslinguistik. 2012. Korpusbasierte Wortgrundformenliste DeReWo, v-ww-bll-320000g-2012-12-31-1.0, mit Benutzerdokumentation. `http://www.ids-mannheim.de/derewo`.

Krister Lindén, Miikka Silfverberg, and Tommi A. Pirinen. 2009. HFST tools for morphology - an efficient open-source package for construction of morphological analyzers. In *State of the Art in Computational Morphology. Workshop on Systems and Frameworks for Computational Morphology, SFCM 2009, Zurich, Switzerland, September 2009. Proceedings*, pages 28–47.

Werner Marti. 1985a. *Bärndütschi Schrybwys: ein Wegweiser zum Aufschreiben in berndeutscher Sprache: mit einer Einführung über allgemeine Probleme des Aufschreibens und einem Wörterverzeichnis nebst Beispielen*. A. Francke, Bern, Switzerland, 2nd edition.

Werner Marti. 1985b. *Berndeutsch-Grammatik für die heutige Mundart zwischen Thun und Jura*. A. Francke, Bern, Switzerland.

Günter Neumann. 2010. Text-basiertes Informationsmanagement. In Kai-Uwe Carstersen, Christian Ebert, Cornelia Ebert, Susanne J. Jekat, Ralf Klabunde, and Hagen Langer, editors, *Computerlinguistik und Sprachtechnologie. Eine Einführung*, pages 576–615. Spektrum Akademischer Verlag, Heidelberg, Germany, 3rd edition.

Tanja Samardžić, Yves Scherrer, and Elvira Glaser. 2016. Archimob - a corpus of spoken Swiss German. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Thierry Declerck, Marko Grobelnik, Bente Maegaard, Joseph Mariani, Asunción Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France, may. European Language Resources Association (ELRA).

Yves Scherrer. 2011. Morphology generation for Swiss German dialects. In *Systems and Frameworks for Computational Morphology - Second International Workshop, SFCM 2011, Zurich, Switzerland, August 26, 2011. Proceedings*, pages 130–140.

Anne Schiller, Simone Teufel, Christine Stöckert, and Christine Thielen. 1999. Guidelines für das Tagging deutscher Textcorpora mit STTS (Kleines und großes Tagset). `http://www.sfs.uni-tuebingen.de/resources/stts-1999.pdf`.

Beat Siebenhaar and Alfred Wyler. 1997. *Dialekt und Hochsprache in der deutschsprachigen Schweiz*. Edition „Pro Helvetia“, Zurich, Switzerland, 5th edition.

Elisabeth Stark, Simone Ueberwasser, and Beni Ruef. 2009–2015. Swiss SMS Corpus. `https://sms.linguistik.uzh.ch`.

Rudolf Suter. 1992. *Baseldeutsch-Grammatik*. Grammatiken und Wörterbücher des Schweizerdeutschen in allgemeinverständlicher Darstellung. Christoph-Merian-Verlag, Basel, Switzerland, 3rd edition.

Simone Ueberwasser. 2013. Non-standard data in Swiss text messages with a special focus on dialectal forms. In Marcos Zampieri and Sascha Diwersy, editors, *Non-standard Data Sources in Corpus-based Research. (=TSM-Studien, Schriften des Zentrums Sprachenvielfalt und Mehrsprachigkeit der Universität zu Köln 5. Hrsg: Christiane M. Bongartz und Claudia M. Riehl)*, pages 7–24, Aachen. Shaker Verlag.

Jeroen Van Pottelberge. 2005. Ist jedes grammatische Verfahren Ergebnis eines Grammatikalisierungsprozesses? Fragen zur Entwicklung des *am*-Progressivs. In Thorsten von Leuschner, Tanja Mortelmans, and Sarah Groodt, editors, *Grammatikalisierung im Deutschen*, pages 169–192. De Gruyter, Berlin, Germany.

Albert Weber and Bund Schwyzertütsch. 1948. *Zürichdeutsche Grammatik: ein Wegweiser zur guten Mundart*. Grammatiken und Wörterbücher des Schweizerdeutschen in allgemeinverständlicher Darstellung. Schweizer Spiegel-Verlag, Zurich, Switzerland.

Andrea Zielinski, Christian Simon, and Tilman Wittl. 2009. Morphisto: Service-oriented open source morphology for German. In *State of the Art in Computational Morphology - Workshop on Systems and Frameworks for Computational Morphology, SFCM 2009, Zurich, Switzerland, September 2009. Proceedings*, pages 64–75.

Item Presentation in Primers - An Analysis Based on Acquisition Research

Kay Berkling
Cooperative State University
Karlsruhe, Germany
berkling@dhbw-karlsruhe.de

Abstract

It is known that children have difficulties with correct spelling of orthographic regularities in German ('liebe', 'kennen'). By looking at instruction material in first grade, this work is a first step of an ongoing study to understand how children's spelling in German is affected by their method of instruction. A major influence on spelling and reading acquisition is the input children receive during the initial phase. It is therefore important to analyse the reading material and understand how these relate to research-based knowledge of acquisition. We show that there is a substantial difference between popular primers (first grade material to teach reading) on how they present material to first graders. It can also be seen that none of the modern primers seem to emphasize item presentation with regularities that help students learn to generalize to new words. These findings are important because the differences have a potential major effect on reading and orthography acquisition that remain mostly unknown and unstudied.

1 Introduction

There are a number of widely accepted theories in the community regarding the acquisition of orthographic and reading skills, though most agree that acquisition, especially in the crucial phase of reaching rapid word recognition is still not perfectly understood. Compounding the issue, is the difficulty in understanding how any findings generalize across languages (Share, 2008).

While there are differences, researchers agree that cognitive predictors are common across all languages, albeit to differing degrees (Ziegler and Goswami, 2005; Caravolas et al., 2012). Among these are Phonological Awareness (PA) [1] and Rapid Automatized Naming (RAN) [2]. Over the first years of acquisition one can observe a gradual shift from PA to RAN as a predictor, with PA being important for a longer period of time in deeper orthographies [3]. PA can be shown to contribute to individual variance in literacy development across languages (Moll et al., 2014). In several studied languages, PA was the best predictor of reading accuracy and spelling whereas RAN was the best predictor of reading speed.

Beginning reading and spelling acquisition therefore depends on phonological awareness and the ability to manipulate phonemes and graphemes in the process of phonological recoding of new words in orthographies of all depths. This holds true also for German.

Self-teaching theory (Share, 1995) is currently the most plausible model to explain the process of reading and spelling acquisition and the training of these relevant cognitive skills. It is based on the idea that children rely and build on phonological decoding skills to learn novel words. The combination of contextual inference, usage of inner lexicon and phonological recoding is then accompanied by the self-teaching strategy as a mechanism

[1] Phonological awareness involves the detection and manipulation of sounds within words - not necessarily involving the written word.

[2] A task that measures how quickly individuals can name aloud an object shown in a picture, including letters.

[3] Orthographic depth relates to the amount of context necessary in order to identify the correct phoneme-grapheme correspondence. For example 'Sp' vs. 'Sn' needs the following letter to determine correct choice of phoneme /ʃ/ vs. /s/. Depth can depend on syllable, word-level or even sentence level and is language dependent. A flat orthography such as Finish does not need context. In contrast English and French ('aimait' vs. 'aimaient', 'aimé' vs. 'aimer') are deep orthographies.

to grow the reader's orthographic lexicon (Ehri, 2005; Share et al., 1984; Jorm et al., 1984; Cunningham et al., 2002; Bowey and Muller, 2005).

Given this theory of self-teaching, the presentation of items to a learning reader is an important consideration in first grade texts (called primers from here on). To our knowledge, these have not been examined in detail. In our previous work we showed that children's orthography skills lack regarding highly frequent, regular German spelling patterns (such as 'l**ie**be', 'ke**nn**en') (Berkling and Lavalley, 2015). Unfortunately, the data for that study did not emphasize the relation to the teaching methods and motivated the need for further study and corpora. Indeed, we found that first grade primers indicated a lack of progression regarding these same regularities that students were having problems with (Berkling et al., 2015).

The work presented here catalogues criteria based on literature in the field of spelling and reading acquisition across disciplines and applies these in an analysis of various well-known primers. Correlating the effect of item presentation with student orthographic abilities or reading proficiency is beyond the scope of this paper but represents a clear next step resulting from a deeper understanding of how teaching materials should be constructed.

The rest of this paper proceeds as follows. Section 2 will compile the latest research results to create the theoretical foundation of the evaluation criteria; the tools for taking a critical look at primary materials used to teach reading and writing. Section 3 lists the primers that have been analyzed. Section 4 describes the system to automatically analyze the texts. Section 5 then discusses how the primers perform with respect to some of the criteria and Section 6 draws some conclusions and outlines some of the next steps in this research.

2 Theory

Self-teaching theory (Share, 1995) states that children are able to establish specific orthographic knowledge through reading experience. Translating letter strings into phonological code, called phonological recoding, is then also used in spelling productions. This has become a widely accepted model for both reading and spelling acquisition across languages (Caravolas et al., 2001; Caravolas and Volin, 2001; Martinet et al., 2004; Ziegler et al., 2014; Cunningham, 2006).

As a consequence of this, model item selection for reading material, especially in primers, for both reading and spelling acquisition may be of great importance. "Self-teaching opportunities afforded by phonological recoding represent the "cutting edge" of reading development not merely for the beginner, but throughout the entire ability range."(Share, 1995, p.155). Therefore, the items must be sequenced, and must either present a self-teaching or practice event for the student.

The goal of this section is to create the argument for item selection criteria that presents items in such a way as to build on the child's previous skills and produce the highest quality lexical entries in order to prepare for second grade reading skill acquisition to build on.

The following principles for item selection in primers are supported by literature findings as we will argue below. They are interdependent:

1. Train PA through phonological recoding while supporting natural sensitivity for regularities
2. Provide pressure for lexical restructuring through progression (presenting successfully more difficult words)
3. Take care with words that don't generalize

2.1 Training PA within Patterns

Phonological Awareness Through Manipulation:
Phonological awareness is trained by providing a network of graphemes and phonemes in various combinations to allow the student to train grapheme-phoneme correspondence and blending which does not happen through repeating words but through presenting many words with the same structure ('hat', 'cat', 'rat' provides more learning than 'hat', 'hat', 'hat'). According to (Melby-Lervag and Hulme, 2010), training children to manipulate phonemes in unfamiliar words improves phonemic manipulation and serial recall of those words.

Regularities:
Regularities are taught through phonics (Pinnell

et al., 1998), the method of scope and sequencing learning material in stages of complexity accompanied by explicit teaching of the material and phenomena that are introduced with each step. Recently, this was confirmed by a meta study on this topic (Galuschka et al., 2014). Duncan shows that phonics instruction increased explicit syllable and rime awareness as well as phonological awareness (Duncan et al., 2013). The effectiveness of teaching through patterns has been widely studied, including their effect on both reading and spelling (Ehri, 1987; Cunningham, 2006; Castles and Nation, 2006; Pacton et al., 2001; Anderson and others, 1977).

Children are very sensitive to the orthographic regularities of their writing system from an early age (Ouellette and Senechal, 2008; Pacton et al., 2001) and produce spellings that conform to the orthographic conventions of their writing system. In French, general orthographic knowledge of regularities influences the recall of newly learned orthographic representations (Pacton et al., 2014; Pacton et al., 2013; Sobaco et al., 2015). Readers learn about spelling patterns that recur in different words, these larger units are then used to form connections to remember words (Bhattacharya and Ehri, 2004). Thus students create patterns based on reading input items that they apply both in reading fluency and spelling skills. Conversely, this also means that showing wrong patterns will result in wrong generalizations, such as is the case with the use of the letter <i> in German children's spelling for /i:/ (Berkling and Lavalley, 2015) as in 'Tiger'. Having seen a larger number of first items in a primer that end in the letter <a>, such as 'Oma', 'Mama', 'Lula', 'lila' and 'Tiger' or 'Igel', a child might be inclined to apply the trained pattern and generalize to a new word /li:bɐ/ and spell it as 'liba' instead of 'lieber'.

There are two key insights towards practice:

- Children generalize rapidly from presented items towards patterns.
- This process has been proven in several languages, regardless of orthographic depth.

2.2 Lexical Restructuring and Progression

Input hypothesis (Krashen, 1981) states that learning moves in stages, where only one new item is presented at a time, building the learning material from simple to more complex items. This should be taken into account when selecting items for the reader. A word like 'Weihnachtsmann' would therefore not be in the 'i+1' scope [4] of a beginning reader and can not serve as an input item for self-teaching. Ability to read begins with simplest conventions at the beginning (Treiman and Cassar, 1997) and expands to more complex ones later on (Pacton et al., 2002), as has been shown in English and French, supporting the idea of allowing for a progression from simple to more complex in reading materials.

Lexical restructuring is a result of self-teaching opportunities at well designed steps of progression. Lexical quality hypothesis (Perfetti and Hart, 2002) states that words vary in the quality with which different aspects of their form and meaning are represented in memory. As the form relates to phonology, morphosyntax and orthography, items should provide these kinds of pressure for lexical restructuring and generalization (by providing new combinations of letters with slowly increasing difficulty, not promoting memorization). Improving the inner lexicon but also supporting emerging phoneme awareness is based on reading input in a similar manner across languages (Ziegler and Goswami, 2005; Ziegler and Goswami, 2005; Mann and Wimmer, 2002; Duncan et al., 2006).

Ignoring Progression:
Acquisition of more difficult words when the mapping between phonemes and graphemes is still unstable may lead to shallower learning and weaker orthographic representation and are prone to disappear in the long term (an example might be 'Fahrrad' spelled as 'Farhad'). Children more sensitive to the frequencies of phoneme-grapheme mappings are better able to detect inconsistencies and memorize these for novel irregular words (Biname and Poncelet, 2016). The study emphasizes the importance for children to master phonological recoding during the first

[4] 'i+1' relates to the next unlearned step that is within the grasp of a student given the current knowledge.

years of school in order to establish their orthographic learning abilities for the large number of words to come in further years of study. Only after learning patterns ('gehen', 'wehen'), students become sensitized to nuances in orthographic detail (for example 'drehen', 'dehnen') (Cunningham, 2006; Share, 2004). Words perceived as irregular effect both reading and writing. Wang (Wang et al., 2012) shows that irregular words are not only decoded less accurately but also encoded less well.

There are two key insights towards progression:

- Security in a lower level of acquisition makes the next level of difficulty possible.
- Ignoring sequences can lead to problems with recall in spelling and reading.

2.3 High Frequency Words

One exception to perceived irregularities are high frequency words (HFW). HFW have a regularity at the text level in that they appear very frequently (40-50% in a normal text) and are often function words. According to Gough (Gough, 1983), these words are predicted 40% of the time. They tend to be 1-Syllable words and often do not follow spelling patterns (for example "ihn"). According to (Ehri, 2005), any word that is read sufficiently often is a sight word. Many studies show that orthographic information is acquired fairly rapidly (Manis et al., 1993; Reitsma, 1983a; Reitsma, 1983b) and the child will recognize these frequent words visually. Therefore, HFW do not represent a self-teaching event to acquire en/decoding patterns for generalization to new words.

There are two key insights to emphasize regarding the use of HFW in item selection:

- Learning around 100 of these HFW will help a beginning reader to quickly be able to read almost half of the occurring words in any text, providing positive feedback.
- These words do not provide practice for self-teaching orthographic principles to support learning future unknown words.

2.4 Summary

The points discussed above are highly interrelated. Important considerations are the continuous training of phonological awareness through recoding within patterns and then moving on to new patterns as the preceding ones have been mastered. These steps provide practice and lexical restructuring in a controlled manner. As learning takes place rapidly, item selection is of crucial importance in the beginning.

The rest of the paper will examine a number of German primer texts with respect to these criteria. As the texts are examined, it is important to keep in mind that the first-grade materials are not limited to the text in the primers. Therefore, this analysis is only an approximation of the input that children receive in first grade teaching environments.

3 Corpus

A selection of well-known primers was taken that represent different ideologies regarding item presentation.

Syllabic Method: This methods assumes that reading is best taught starting with the syllable. Therefore, the book starts with teaching syllables instead of letters in isolation. Usually, this method also distinguishes stressed from unstressed syllables ('Mutter'). An example of such a primer is 'ABC der Tiere' (Handt et al., 2010). The first words in one of the versions consists of one page repeating the word 'mu'. Both letters are learned in the context of the syllable and not in isolation.

Analytic-Synthetic Method: This method assumes that there is more or less a 1-1 correspondence between phoneme and grapheme with some more nuances that are postponed to a later stage, without following a phonics approach. Examples tending in this direction to varying degrees are 'Kunterbunt' (Bartnitzky, 2009), 'Jo-Jo' (Namour et al., 2011), and 'Tinto' (Anders and Urbanek, 2004). It is assumed that after introducing letters 'Ii,Aa,Oo,Nn,Mm,Ll,Tt', the student can read 'Tina', 'Oma', 'Ali', 'Ana', 'Lila' by sounding out the letters. (The ensuing difficulty for children to decode (read) 'lieber' or encode (write) /liːbə/ with these training samples may be under-

estimated.)

The Whole Word Method: This method extends the previous with extensive practice at the word level. 'Tobi' (Metze, 2002) shows a tendency in this direction. The text has since been adapted to include syllables.

Phonics Method: The phonics approach emphasizes regularity ('lieber', 'Diener', 'bieter') over simplification ('lila', 'oma'). There are old primers from the turn of the century that exemplify this method, such as what will be called 'Alte Fibel' (Stöwesand, 1903) in this paper. Patterns are important and treated differently from 'HFW'. There is a clear progression from what the author deems 'easy' to 'difficult' in relation to patterns in typically German words.

4 System

In order to analyze the texts with respect to the criteria established in Section 2, a means of categorizing items (words) used in the primers is needed. The goal is to study the use of regularities in presented items because these support learning and generalizing from seen items to new words. Table 1 lists the most important categories of words, distinguishing regular spelling patterns in German from other categories of words. The regular word occurrences in learner texts are important to identify because they support self-teaching events at the beginner level. In its simplest form, there is the 2-syllable trochee (stressed/unstressed 'Betten') and the 1-syllable form that derives its spelling from the latter ('Bett'). The other listed categories include low-frequency exceptions (these do not help to generalize), high frequency words (these are quickly memorized as whole-word image and support reading but not generalization) and longer words ('Handschuh', 'vergehen'), regular but very difficult for the early stages. A hypothesis yet to be studied is that complex onset may prove difficult for children to learn, a reason to distinguish this additional category. Finally, the category 'other' encompasses word structures that inherently do not generalize to other German words as they do not conform to German orthography. Together, these categories cover items seen in the text. The automated classification algorithm needed to process large quantities of words is described in this section.

Software modules form the basis for analyzing the primers' texts building on the one described in (Berkling et al., 2015). The system proceeds in three steps. First, the pronunciation of each item is obtained with speech synthesis supporting tool Balloon (Reichel, 2012). Given a text, Balloon returns pronunciation, morpheme and syllable boundaries. In a second step, this output enables the construction of the correct grapheme sequence [5].

Word Type	Description or Example	Self-Teaching Event
HFW	100 most frequent words = %45 of all words	memorized
1-syllable short vowel	Bett (1Sv)	yes
1-syllable long vowel	saust, lieb (1SV)	yes
2 syllables short vowel one consonant	Sonne, Wonne (Betten)	yes
2 syllables short vowel two consonants	fester, Äste (besten)	yes
2 syllables long vowel	Nadel, geben (beten)	yes
complex onset	Klasse, klapper	yes
other	Lulu, Lala, Auto	not generalizable
aa,oo,eh,	Boot	exception
ck, tz, ng	Bäcker	exception
too long	Weihnachten Sommerferien	too difficult

Table 1: List of identified 'Word Categories' for the purpose of this study.

Finally, the graphemes can be assigned to allowed positions in the regular German syllable as shown in Table 1. $K1$, $K2$ etc. are column names

[5] Examples of grapheme assignment difficulties are 'a-n-||-n-eh-m-en' /nn/ vs. 'a-nn-e' /n/ depends on the morpheme boundary provided by Balloon; 'W-e-s-p-e' contains /sp/ vs 'g-e-sp-ie-l-t' /ʃp/; 'n.äh.r.en' vs. 'n.ä.h.en'. Graphemes in this context are letter sequences used for teaching purposes (sp, nn, h, sch,...).

for each of the allowed positions of graphemes. Depending on how these columns are filled, the words are then classified into the proposed categories (Berkling et al., 2015).

Regular words can be categorized according to the table above. When position $K(shortV)$ is filled, the vowel must be short. When the Rime is not filled, it is a 1-syllable word. The following are some examples:

1. 2-syllable, long vowel: K1-V-K1-Rime ('b-e-t-en')

2. 2-syllable, short vowel with one consonant phonemes: K1-v-K(shortvowel)-K1-Rime ('B-e-tt-en'), because K(shortvowel) and K1 after the syllable boundary are a single grapheme (indicated by '!')

3. Short vowel with two consonant phonemes: K1-v-K(shortvowel)-K1-Rime ('b-e-s-t-e-n')

Words in category 'other' are identified, when graphemes occur in non-allowed positions. They are not typically German words and do not generalize. Examples are 'Auto', 'Mama','Lula', where 'a' and 'o' are not allowed vowels in the second, unstressed syllable.

Further, high frequency words ('HFW') are distinguished in a separate category (Quasthoff and Richter, 1998), the list of words used to select this set are statistically those 100 words that make up the top 45% occurrences in normal German texts.

Words that do not fit into the 1- or 2-syllable category and are 'too long' (for primers), including prefix or composites ('vergehen', 'Weihnachtsmann' or 'Handpuppe') are listed in a separate category. They are easily identified by their number of syllables.

Spellings that follow German structure but exhibit irregular graphemes in consonants (such as 'ck') or vowels (such as 'oo') are categorized separately according to vowel and consonant exceptions.

Complex onsets, when columns $K1$ and $K2$ are both filled in the first syllable, are categorized separately in 'complex', regardless of whether they have one or two syllables.

The output of this algorithm produced a sequence of categories, one for each word of the entire text. This sequence is then analyzed regarding regularities it item presentation of any text. By inspection, few words are misclassified and the small number of mis-classifications do not affect the overall analysis results. An example of word classification output of this system is given below:

```
sein      (HFW)
sind      (HFW)
ist       (HFW)
fest      (1-Syllable short vowel)
reist     (1-Syllable long vowel)
saust     (1-Syllable long vowel)
alt       (1-Syllable short vowel)
klapper   (Comlex onset)
nudel     (2-Syllable long vowel)
sonne     (2-Syllable short vowel)
ente      (2-Syllable short vowel)
```

Word	Core	Morpheme Boudary	K1	K2	Vowel	K (shortV)	Syll	K1	K2	Morpheme B.	Rime
schnell	schnell	||	sch	n	e	l	.		l	||	
streben	streb	||	st	r	e		.	b		||	en
zielen	ziel	||	z		ie		.		l	||	en
ziehen	zieh	||	z		ie		.	h		||	en
essen	ess	||			e	s!	.	s!		||	en
lecker	leck	||		l	e	c!	.	k!		||	er
fasten	fast	||	f		a	s	.	t		||	en

Figure 1: *System of splitting graphemes into allowed syllable slots.*

5 Results

5.1 Practice and Patterns

Practice can be achieved at the level of phonemes, syllables (patterns) or words.

At the phoneme level, Table 3 (see Appendix) lists the letters and their frequency of usage within the patterns of trochee mentioned in Section 4 over the first 1300 words for the example of 'Alte Fibel'. It is more or less representative of all other primers. At this level of practice there is no visible difference. Regarding practice at the syllable and word level, Table 2 lists the lexical diversity and syllable diversity after the first 1300 word in

each of the primers. Most of the primers are more or less similar. The outlier is 'Alte Fibel' with a much larger lexical diversity. At the same time, the number of total used syllables is lower, indicating the use of shorter words. Looking at the syllable diversity, it can be seen that the reader is presented with a significant larger number of syllables than in any of the other readers (769 syllables, the runner up being 'Tinto' with 610). At the syllable level, the learner has been exposed to a larger number of self-teaching opportunities both at syllable and word level.

Title	LD	# Syllables	SD
ABC der Tiere	.39	1951	.24
JoJo	.32	1999	.21
Tinto	.45	1938	.31
Kunterbunt	.37	1979	.26
Tobi	.41	1937	.28
Alte Fibel	.62	1784	.43

Table 2: Statistics for each of the analyzed primers after the first 1300 words. LD=Lexical Diversity (Types/Tokens). SD=Syllable Diversity, using counts for types and tokens at syllable level. High diversity means more practice on new words/syllables.

5.2 Progression

Being exposed to a larger number of self-teaching opportunities may be difficult if these are not taking place within known patterns. This presupposes a progression at the pattern level.

Figures 2, 3 and 4 depict the percentage of 'word category' usage in the text seen up to a given point in time (marked by the number of words seen so far on the y-axis). 'Tobi' exhibits a natural combination of word categories in use from the start. 'Tinto' has a slightly manipulated higher frequency of HFW and 1-syllable words. 'Kunterbunt' starts with a large proportion of HFW with all other patterns appearing in equal measure. 'Jo-Jo' has a very low number of both HFW and regular words in the beginning. Both 'ABC der Tiere' and 'Alte Fibel' are different from the other primers. 'ABC der Tiere' exhibits a very high usage of 1-syllable words and a late but strong start of HFW (also usually 1-syllable

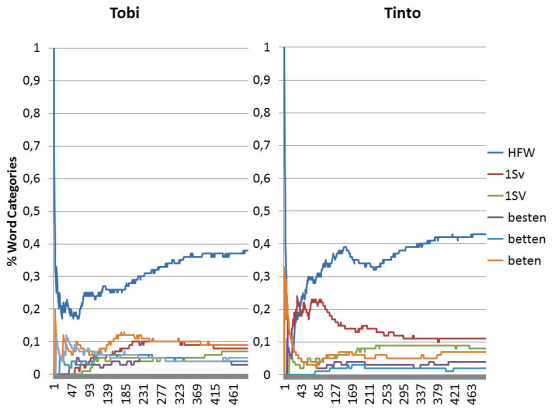

Figure 2: *% distribution of word category usage for regular patterns and HFW.*

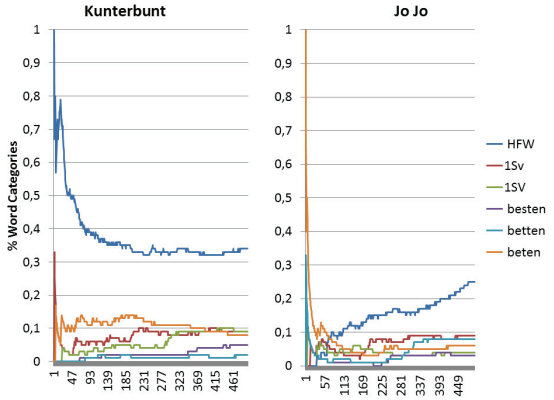

Figure 3: *% distribution of word category usage for regular patterns and HFW.*

words). 'Alte Fibel' is the only primer that shows a very clean separation in time for all different patterns. Starting the reader off with HFW and 1-syllable words of both long and short vowel types, 2-syllable words are introduced much later, one pattern at a time, only then moving on to words that include complex onset.

5.3 Function Words (HFW) and Non-Pattern Words

High Frequency Words: The natural occurrence frequency of HFW in a normal text is around 40-45%. By the end of the text, all primers reach about the level of 30-45%. It is interesting to note that 'Alte Fibel' de-emphasizes HFW in favor of self-teaching events that help students to generalize to new words. Figure 5 will show that

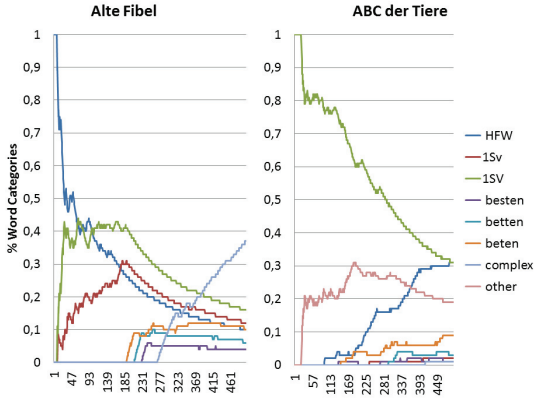

Figure 4: *% distribution of word category usage for regular patterns and HFW. This graph includes 'other' and 'complex onset'.*

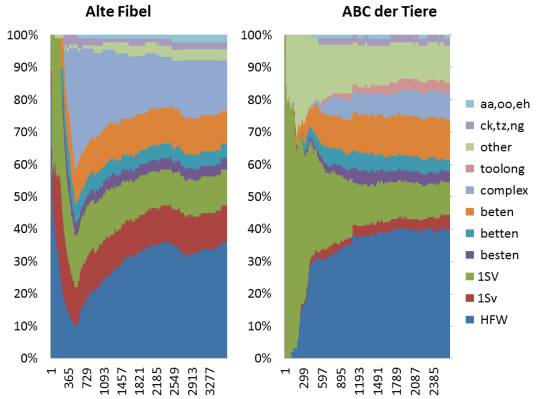

Figure 5: *Progression of word type % that make up the text as seen until word n. 'Alte Fibel' and 'ABC der Tiere' exhibit a clear contrast in approach during the first thousand item presentations while looking similar at the end. 'Alte Fibel' has almost no words of category 'other'.*

this is a temporary condition in the beginning of the book before moving to above 30% towards the end of the book.

Irregular Words: Irregular words are those that do not conform to the German spelling system, that is their patterns do not generalize to new words that are frequent in German. Their use not only does not generalize but may induce self-teaching events that will create patterns that are false. Since learning takes place with very few examples, these self-teaching events can affect further reading ability because the student fails to generalize to be able to read new words that have a different pattern from the ones learned in the beginning. It also has an effect on spelling as discussed in Section 2. It is therefore interesting to see if and how 'other' words (not conforming to the German spelling system, like 'Auto' or 'Oma') are used in the beginning. The percentage of 'other' words in the text is plotted over time in terms of words seen (x-axis) in Figures 6 and 7. It is interesting to note that 'Alte Fibel' has virtually none of these types of words. 'ABC der Tiere' has a consistent 20% of such words in the beginning (plotted in Figure 4). All primers exhibits a fairly high % of words that do not generalize to the regular patterns that appear in the German language. The percentage is especially high in the first presented items. Many of these words may be names and introduce the people in the story. In order to see whether these words become HFW through high usage, Figure 8 plots the number of words in this category and shows their distribution over the top most frequent words. What we would like to see is a large number of repetitions and few words of this type to prevent pattern formation across many self-teaching events and encourage visual recognition. 'Jo-Jo' and 'Tobi' show this profile while 'Kunterbunt' has a large number of 'other' words with low frequency count of each as do 'ABC der Tiere' and 'Tinto'.

Irregular spellings that are part of the German orthographic system like 'ck' instead of 'kk' or 'oo' in 'Boot' do not appear in the first items for both 'ABC der Tiere' and 'Alte Fibel' while they appear in all other primers. Irregular spellings are rare but have a low frequency and so do not necessarily move into visual recognition automatically for readers. They may or may not confuse the learner in the early pattern construction phase.

5.4 Summary

The theoretical background on what is known about reading and spelling acquisition in the context of the self-teaching theory provides guidelines for a structured approach towards analyzing the selection of items presented to beginning readers. Primers are an indicator of the order in which these are presented to children in the classroom.

In theory, practice, patterns, progression after

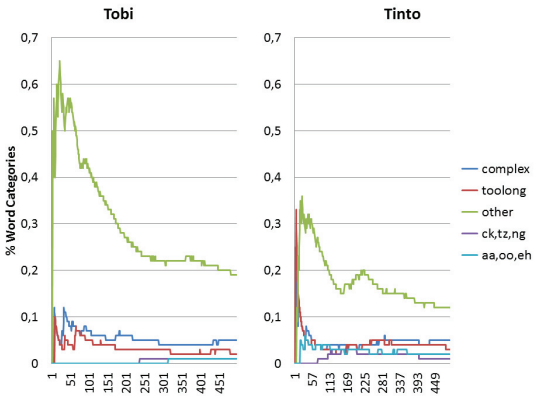

Figure 6: *% distribution of word type usage for irregular and more difficult patterns.*

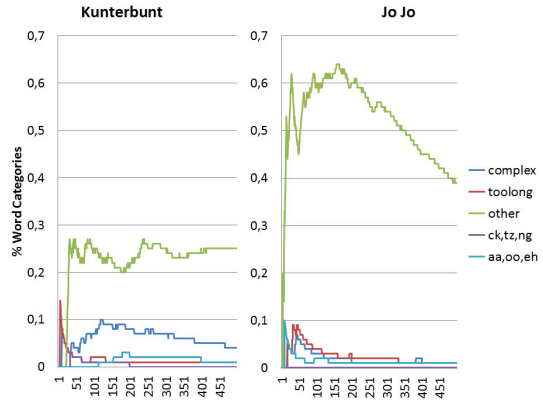

Figure 7: *% distribution of word type usage for irregular and more difficult patterns.*

mastery and careful use of items that do either not provide a self-teaching event ('HFW') or provide a wrong self-teaching events ('other') can be used to analyse the texts.

In all accounts, practice, patterns, clear progression and careful use of 'HFW' and 'other', as well as special attention to irregularities in German orthography, 'Alte Fibel' outperformed todays' primers. Other primers exhibit partial aspects in which they adhere to theoretical propositions on presenting items.

Research tells us that learning takes place quickly given the correct presentation of items. There is a clear need to study what the effect is on reading and spelling depending on item selection that places more emphasis on 'other' especially in the first items that are presented.

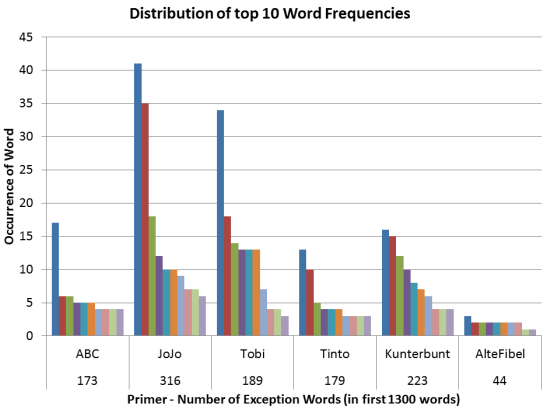

Figure 8: *Distribution of 10 most frequent non-regular words. Word list differs for each book.*

6 Conclusions

The main contributions of this paper are twofold:

- An overview of theoretical background to establish criteria for reading and writing teaching materials
- followed by an automated quantitative analysis of primers based on automated speech and text processing technology.

Given the theoretical background on what is known about reading and spelling acquisition, several points for evaluation of input items have been motivated through the literature and various primers were analyzed with respect to these criteria. It was shown that there are very different approaches for item presentation. It is also important to note that the input items are not restricted to the primers. However, the chosen method will most likely extend to the additional materials. These different methods for item presentations will have an effect on spelling and reading ability and this effect needs to be studied and understood in more detail. The presented analyses provide a small window on the learning materials with imperfect automated tools. It is well known that context effects are important in self-teaching events. These have not been addressed in the present analysis. Also, generalizability of items to new items can be quantified and will be part of future work in this area. The next step is to correlate orthographic skill acquisition with the quality of teaching material.

Appendix

K1v	K1v	K2v	K2v	V	V	vK1	vK1	K1e	K1e	K2e	K2e	Rime	
	89		147	e	52		102		64		155		207
h	21	w	21	a	43	l	20	g	26	l!	16	en	96
sch	20	r	15	o	29	n	18	d	21	l	14	e	60
s	15	m	11	ei	20	l!	16	t	18	n!	13	er	26
d	13	l	9	i	15	n!	13	s	15	n	5	el	21
v	11	n	8	ü	10	t!	12	t!	12	m!	4	te	2
st	11			ie	10	s!	11	s!	11	m	3	ern	2
sp	9			u	9	m!	4	f	9	t	1	et	2
f	8			ä	6	ch	4	b	8			ten	1
b	6			äu	5	f!	3	z	6			es	1
g	3			ö	5	s	3	ch	5				
k	2			au	5	m	1	k	4				
z	2			eu	2	p	1	h	4				
p	1					p!	1	f!	3				
						f	1	p	2				
						t	1	ß	2				
								p!	1				

Table 3: Distribution of letters and their occurrence frequency for "Alte Fibel". The other primers have similar distributions.(! denotes a letter in a double consonant, like "bit!t!er".)

References

Linda Anders and Rüdiger Urbanek. 2004. *Tinto blau: [Lesen- und Schreibenlernen im offenen Anfangsunterricht]*. Cornelsen, Berlin, 1. aufl., [nachdr.] edition.

Richard C. Anderson et al. 1977. Instantiation of Word Meanings in Children. *Technical Report*.

Horst Bartnitzky. 2009. *Kunterbunt / Fibel*. Klett, Stuttgart and Leipzig, 1 edition.

Kay Berkling and Rémi Lavalley. 2015. WISE: A Web-Interface for Spelling Error Recognition Description and Evaluation of the Algorithm for German. In *International Conference of the German Society for Computational Linguistics and Language Technology*, GSCL, pages 87–96. Gesellschaft für Sprachtechnologie and Computerlinguists.

Kay Berkling, Rémi Lavalley, and Uwe Reichel. 2015. Systematic Acquisition of Reading and Writing: An Exploration of Structure in Didactic Elementary Texts for German. In *International Conference of the German Society for Computational Linguistics and Language Technology*, GSCL, pages 87–96. Gesellschaft für Sprachtechnologie and Computerlinguists.

A. Bhattacharya and L. C. Ehri. 2004. Graphosyllabic Analysis Helps Adolescent Struggling Readers Read and Spell Words. *Journal of Learning Disabilities*, 37(4):331–348.

Florence Biname and Martine Poncelet. 2016. The development of the abilities to acquire novel detailed orthographic representations and maintain them in long-term memory. *Journal of experimental child psychology*, 143:14–33.

Judith A. Bowey and David Muller. 2005. Phonological recoding and rapid orthographic learning in third-graders' silent reading: a critical test of the self-teaching hypothesis. *Journal of experimental child psychology*, 92(3):203–219.

M. Caravolas and J. Volin. 2001. Phonological spelling errors among dyslexic children learning a transparent orthography: the case of Czech. *Dyslexia (Chichester, England)*, 7(4):229–245.

Markéta Caravolas, Charles Hulme, and Margaret J. Snowling. 2001. The Foundations of Spelling Ability: Evidence from a 3-Year Longitudinal Study. *Journal of Memory and Language*, 45(4):751–774.

Marketa Caravolas, Arne Lervag, Petroula Mousikou, Corina Efrim, Miroslav Litavsky, Eduardo Onochie-Quintanilla, Nayme Salas, Miroslava Schoffelova, Sylvia Defior, Marina Mikulajova, Gabriela Seidlova-Malkova, and Charles Hulme. 2012. Common patterns of prediction of literacy

development in different alphabetic orthographies. *Psychological science*, 23(6):678–686.

Anne Castles and Kate Nation. 2006. How does orthographic learning happen? *From inkmarks to ideas: Current issues in lexical processing*, pages 151–179.

Anne E. Cunningham, Kathryn E. Perry, Keith E. Stanovich, and David L. Share. 2002. Orthographic learning during reading: Examining the role of self-teaching. *Journal of experimental child psychology*, 82(3):185–199.

Anne E. Cunningham. 2006. Accounting for children's orthographic learning while reading text: do children self-teach? *Journal of experimental child psychology*, 95(1):56–77.

Lynne G. Duncan, PASCALE COLÉ, Philip H. K. Seymour, and ANNIE MAGNAN. 2006. Differing sequences of metaphonological development in French and English. *Journal of Child Language*, 33(02):369.

Lynne G. Duncan, Sao Luis Castro, Sylvia Defior, Philip H. K. Seymour, Sheila Baillie, Jacqueline Leybaert, Philippe Mousty, Nathalie Genard, Menelaos Sarris, Costas D. Porpodas, Rannveig Lund, Baldur Sigurethsson, Anna S. Thornrainsdottir, Ana Sucena, and Francisca Serrano. 2013. Phonological development in relation to native language and literacy: variations on a theme in six alphabetic orthographies. *Cognition*, 127(3):398–419.

Linnea C. Ehri. 1987. Learning to read and spell words. *Journal of Literacy Research*, 19(1):5–31.

Linnea C. Ehri. 2005. Learning to Read Words: Theory, Findings, and Issues. *Scientific Studies of Reading*, 9(2):167–188.

Katharina Galuschka, Elena Ise, Kathrin Krick, and Gerd Schulte-Korne. 2014. Effectiveness of treatment approaches for children and adolescents with reading disabilities: a meta-analysis of randomized controlled trials. *PloS one*, 9(2):e89900.

Philip B. Gough. 1983. Context, form, and interaction. *Eye movements in reading*, 331:358.

Rosmarie Handt, Klause Kuhn, Kerstin Mrowka-Nienstedt, and Ingrid Hecht. 2010. *Die Silbenfibel*, volume [Hauptbd.] of *ABC der Tiere - Die Silbenfibel : Lesen in Silben*. Mildenberger, Offenburg, 2 edition.

A. F. Jorm, D. L. Share, R. Maclean, and R. G. Matthews. 1984. Phonological recoding skills and learning to read: A longitudinal study. *Applied Psycholinguistics*, 5(03):201.

Stephen Krashen. 1981. Second language acquisition. *Second Language Learning*, pages 19–39.

Franklin R. Manis, Rebecca Custodio, and Patricia A. Szeszulski. 1993. Development of phonological and orthographic skill: A 2-year longitudinal study of dyslexic children. *Journal of experimental child psychology*, 56(1):64–86.

Virginia Mann and Heinz Wimmer. 2002. Phoneme awareness and pathways into literacy: A comparison of German and American children. *Reading and Writing*, 15(7-8):653–682.

Catherine Martinet, Sylviane Valdois, and Michel Fayol. 2004. Lexical orthographic knowledge develops from the beginning of literacy acquisition. *Cognition*, 91(2):B11–B22.

Monica Melby-Lervag and Charles Hulme. 2010. Serial and free recall in children can be improved by training: evidence for the importance of phonological and semantic representations in immediate memory tasks. *Psychological science*, 21(11):1694–1700.

Wilfried Metze. 2002. *Tobi-Fibel*. Cornelsen, Berlin, neubearb edition.

Kristina Moll, Franck Ramus, Jürgen Bartling, Jennifer Bruder, Sarah Kunze, Nina Neuhoff, Silke Streiftau, Heikki Lyytinen, Paavo H.T. Leppänen, Kaisa Lohvansuu, Dénes Tóth, Ferenc Honbolygó, Valéria Csépe, Caroline Bogliotti, Stéphanie Iannuzzi, Jean-François Démonet, Emilie Longeras, Sylviane Valdois, Florence George, Isabelle Soares-Boucaud, Marie-France Le Heuzey, Catherine Billard, Michael O'Donovan, Gary Hill, Julie Williams, Daniel Brandeis, Urs Maurer, Enrico Schulz, Sanne van der Mark, Bertram Müller-Myhsok, Gerd Schulte-Körne, and Karin Landerl. 2014. Cognitive mechanisms underlying reading and spelling development in five European orthographies. *Learning and Instruction*, 29:65–77.

Nicole Namour, Günter J. Renk, Wilfried Metze, Kerstin Rahm, Kai Stäpeler, Sabine Kierzek, Martina Schramm, Jana Arnold, Franz Zauleck, and Liane Lemke. 2011. *Jo-Jo Fibel: Ein Leselehrgang*. Cornelsen, Berlin, 2008 edition.

Gene Ouellette and Monique Senechal. 2008. Pathways to literacy: a study of invented spelling and its role in learning to read. *Child Development*, 79(4):899–913.

Sébastien Pacton, Pierre Perruchet, Michel Fayol, and Axel Cleeremans. 2001. Implicit learning out of the lab: The case of orthographic regularities. *Journal of Experimental Psychology: General*, 130(3):401–426.

Sébastien Pacton, Michel Fayol, and Pierre Perruchet. 2002. The acquisition of untaught orthographic regularities in French. *Precursors of functional literacy*, pages 121–137.

Sebastien Pacton, Jean Noel Foulin, Severine Casalis, and Rebecca Treiman. 2013. Children benefit from

morphological relatedness when they learn to spell new words. *Frontiers in psychology*, 4:696.

Sebastien Pacton, Gaelle Borchardt, Rebecca Treiman, Bernard Lete, and Michel Fayol. 2014. Learning to spell from reading: general knowledge about spelling patterns influences memory for specific words. *Quarterly journal of experimental psychology (2006)*, 67(5):1019–1036.

Charles A. Perfetti and Lesley Hart. 2002. The lexical quality hypothesis. *Precursors of functional literacy*, 11:67–86.

Gay Su Pinnell, Irene C. Fountas, and Mary Ellen Giacobbe. 1998. *Word matters: Teaching phonics and spelling in the reading/writing classroom*. Heinemann, Portsmouth, NH.

Uwe Quasthoff and Matthias Richter. 1998. Projekt Deutscher Wortschatz. *Linguistik und neue medien. DUV*.

Uwe Reichel. 2012. PermA and Balloon: Tools for string alignment and text processing. In *Proc. Interspeech*, pages 1874–1877, Portland and Oregon.

Pieter Reitsma. 1983a. Printed word learning in beginning readers. *Journal of experimental child psychology*, 36(2):321–339.

Pytter Reitsma. 1983b. Word-specific Knowledge in Beginning Reading. *Journal of Research in Reading*, 6(1):41–56.

David L. Share, Anthony F. Jorm, Rod Maclean, and Russell Matthews. 1984. Sources of individual differences in reading acquisition. *Journal of Educational Psychology*, 76(6):1309–1324.

David L. Share. 1995. Phonological recoding and self-teaching: Sine qua non of reading acquisition. *Cognition*, 55(2):151–218.

David L. Share. 2004. Orthographic learning at a glance: on the time course and developmental onset of self-teaching. *Journal of experimental child psychology*, 87(4):267–298.

David L. Share. 2008. On the Anglocentricities of current reading research and practice: the perils of overreliance on an outlier orthography. *Psychological bulletin*, 134(4):584–615.

Amelie Sobaco, Rebecca Treiman, Ronald Peereman, Gaelle Borchardt, and Sebastien Pacton. 2015. The influence of graphotactic knowledge on adults' learning of spelling. *Memory & cognition*, 43(4):593–604.

F. Stöwesand. 1903. *Lesebuch der Kleinen: Schreiblese- und Normalwortmethode, den Grundsatzen der Phonetik und mit Berücksichtigung der Schwachbegabten: Ausgabe A in zwei Teilen. Für Volksschulen: Erstes und zweites Schuljahr*, volume 1. Verlag von C.E.Klotz, Magdeburg, 3 edition.

Rebecca Treiman and Marie Cassar. 1997. Spelling acquisition in English. *Learning to spell: Research, theory, and practice across languages*, pages 61–80.

Hua-Chen Wang, Anne Castles, and Lyndsey Nickels. 2012. Word regularity affects orthographic learning. *Quarterly journal of experimental psychology (2006)*, 65(5):856–864.

Johannes C. Ziegler and Usha Goswami. 2005. Reading acquisition, developmental dyslexia, and skilled reading across languages: a psycholinguistic grain size theory. *Psychological bulletin*, 131(1):3–29.

Johannes C. Ziegler, Conrad Perry, and Marco Zorzi. 2014. Modelling reading development through phonological decoding and self-teaching: implications for dyslexia. *Philosophical transactions of the Royal Society of London. Series B, Biological sciences*, 369(1634):20120397.

Crowdsourcing Swiss Dialect Transcriptions for Assessing Factors in Writing Variations

Simon Clematide[1], **Karina Frick**[2], **Noëmi Aepli**[1], **Jean-Philippe Goldman**[2]
[1]Institute of Computational Linguistics,
[2]Zurich Center for Linguistics,
University of Zurich,
simon.clematide@uzh.ch

Abstract

In this paper, we systematically analyze writing variations of Swiss German in two existing corpora with standard German glosses, a corpus of 10,000 short text messages and a corpus of transcribed oral history recordings (90,000 tokens). We show that neither resource is sufficient for assessing factors in writing variations of users and describe a data collection project involving a citizen science community for solving this problem. Laymen will independently and redundantly transcribe 1,200 short samples (15-20 seconds) of audio material in Swiss German according to their own best practice.

1 Introduction

Over the last two decades, with the rise of new media in our everyday lives, writing in Swiss German has become very popular and its usage has increased considerably in private written communication such as text messages, e-mails or Facebook postings (Siebenhaar, 2008, p.2). There can no longer be talk of a "medial diglossia" (Kolde, 1981, p.68), which assumes that spoken dialect and written Standard German are functionally divided. Other factors, such as formality, communicative immediacy and distance have become far more important regarding the choice between (written) dialect and Standard German. Moreover, the popularity of writing in dialect has a lot to do with the fact that no official standard norm exists for the orthography of the Swiss German dialect (Christen, 2004, p.77). That is to say that users writing in Swiss German cannot violate any norms or make any mistakes which could possibly be sanctioned; this might be one of the main reasons why many language users in the German-speaking part of Switzerland prefer using dialect in their private correspondence (Aschwanden, 2001, 62). Furthermore, dialect is connoted very positively for Swiss German speakers and is also regarded as emotional whereas High German is perceived as rather impersonal and aloof (Sieber, 2010, p.380).

2 Related Work

The non-existence of an orthographic norm leads to many different writing variants in private written communication, as, for example, Siebenhaar (2003; 2006) has shown for Swiss chat rooms. He finds that there is a great variety of dialect writings for 8 investigated lexemes (Siebenhaar, 2006, 233). Although there have been various efforts to unify the spelling of Swiss German dialects, e.g. by (Dieth, 1986) (1938) or (Marti, 1985) (1972), they do not have any influence on chat users. This is certainly owed to the simple fact that users normally do not know these expert guidelines because they are not taught in school (Siebenhaar, 2006, 54). Instead, as Siebenhaar (2003, p.134) points out, the written dialect observable in chat rooms reflects a spontaneous vernacular spelling which is not bound to any standard rules but rather to phonetic distinctions in the different Swiss German dialects, e.g. Bernese or Zurich German. That is why in some cases the non-standardized vernacular writing "[...] still reflects the geo-linguistic distribution described in the linguistic atlas of German speaking Switzerland SDS (1962-1997) based on recordings of the 1940s and 1950s." (ibid: 125). Next to the phonetic influence, social variables and individual preferences concerning the scripting play an important role (ibid: 134).

3 Materials and Methods

There exist two larger corpora of Swiss German where spelling variation can be measured by comparing different realizations of written words with respect to normalized standard German glosses. The first one, SMS4science, is truly user-generated

German	*English*	*H*	*Swiss German Variants normalized to lowercase (Frequency)*
			SMS4Science
nächste	next	4.1	nächst(23), nächscht(16), nöchst(13), nögscht(11), nögst(6), next(6), nechscht(5), nägscht(4), negst(4), nögsch(4), negsht(4), näxt(4), nöchscht(3), negscht(3), nächsti(2), nächste(2), näxti(1), nächsht(1), nächschti(1), nägst(1), nöchschti(1), nechst(1), nöchschte(1), nöchsti(1), nächschte(1), nechsti(1), nàxt(1), nöxst(1), nögschd(1)
wochenende	weekend	3.2	wuchenend(36), wuchenänd(19), wucheänd(13), wucheend(11), wochenend(4), wucheendi(4), wochenende(3), wochenänd(3), we(2), wuchäänd(2), wocheänd(2), wuchaend(2), wuchenendi(2), wuchaändi(1), wocheendi(1), wuchenäd(1), wuchend(1), wochänend(1), wuchänänd(1), wuchanend(1)
vielleicht	maybe	3.3	vilicht(62), villicht(22), viellicht(16), vilich(11), velecht(9), filicht(8), vilech(4), velicht(3), velech(3), villich(3), vellecht(3), vielicht(3), filich(2), vielich(2), vellicht(2), viellech(1), filcht(1), vielech(1), vielleicht(1), vilivh(1), viellecht(1), vilecht(1), vellech(1), vilichd(1)
ich	I	1.3	ich(2896), i(1791), ech(115), ig(50), e(33), ih(17), iich(14), ni(9), ìch(5), ch(4), eg(3), ii(3), y(3), ici(2), hch(2), icg(1), ych(1), ìg(1), ìcg(1), iiich(1), isch(1), ibh(1), 'ch(1)
			Archimob
nachher	afterwards	4.2	nachher(13), ne(10), nòchethèèr(8), nòchhèèr(6), nòchher(5), nachhèèr(4), na(3), nòhèèr(3), nacher(3), naher(3), nòcher(2), no(2), nòher(2), näächer(2), nòchhèr(2), när(2), nachhär(1), nochhèèr(1), neecher(1), nä(1), nähär(1), nor(1), nochher(1), nahene(1), nòchether(1), nachhäär(1)
erdapfel	potato	2.3	hèrdöpfel(4), häärdöpfel(4), härdöpfel(3), härdepfu(1), hòrdöpfel(1), hò̀ò̀rdöpfel(1)
vielleicht	maybe	0.6	vilicht(66), vilich(4), viilicht(1), vilìcht(1), vìlìcht(1)
ich	I	1.0	ich(1157), iich(214), i(115), ch(3), ii(3), si(1)

Table 1: Writing variations in Swiss German short messages and expert transcriptions including their overall entropy (*H*)

content of short text messages originally written in Swiss German. Apart from the phonetic distinctions, we find all kinds of idiosyncratic spelling behaviour in this material, according to the "anything-goes" orthography (Dürscheid and Stark, 2013). The second corpus, ArchiMob, contains content that was transcribed from audio material by trained linguists. Therefore, the spelling variations should only reflect the phonemic distinctions that were in the focus for this corpus. In the next section, we contrast these two very different resources.

SMS4Science The Swiss SMS4Science Corpus[1] contains 10,706 short text messages that are mainly written in Swiss German. All messages were donated by volunteers who could also provide sociolinguistic and demographic metadata by filling out a questionnaire with topics such as gender, age, domicile, mother tongue, SMS use, or the use of T9.

As described in Ruef and Ueberwasser (2013), all messages were tokenized and an interlinear glossing in mostly standard German wordings (existing helvetisms were used as much as possible) was manually added. The glossing also split fused Swiss German words[2] and clitics (e.g. "chani" (*can I*) into their corresponding and orthographically correct equivalents ("kann ich"). The manually created glosses were then automatically processed by two different morpho-syntactic taggers, the TreeTagger (Schmid, 1995) assigning standard part-of-speech tags and the RFTagger (Schmid and Laws, 2008) assigning fine-grained morphological tags. The latter would allow to search for specific inflected words, for instance, a verb form in first person singular present tense. However, in order to keep the evaluation of both corpora comparable, we ignore the morphological features of SMS4Science.

For our evaluation on writing variations in short messages, we focus on words with single word glosses and ignore the phenomenon of dialectal or orthographical fusion of words. Using the ANNIS query interface to the Swiss German SMS4science subcorpus we searched for all words with a single gloss in standard German. For technical reasons[3],

[1] See sms4science.ch. Of total 25,947 messages, 41% are Swiss German, 28% Standard German, 18% French, 6% Italian, and 4% Romansh.

[2] Sometimes purely idiosyncratic orthography shows up, e.g. "ichdenkedudörfschsichermitfahre" (*I think you can surely ride with us*).

[3] Unfortunately, the SMS4science corpus cannot be downloaded in a suitable XML format. In order to exclude writing variations that originate from fused words, for instance, "chani" (*can I*) as a variation of "kann" (*can*), we had to restrict

the last token of each message could not be retrieved and from the total of 288,434 Swiss German tokens we could collect 249,029 (86%). Of these, 1,677 were manually marked as abbreviations and therefore excluded from our statistics.[4] To keep the results from SMS4science and Archimob comparable, we normalized the glosses and the word forms to lowercase. 49,591 glosses appear only once, leaving us with 197,761 tokens where we actually might observe writing variation.

We suggest to quantitatively measure the amount of variation in terms of the minimal amount of bits needed for encoding all variants, thus taking into account the number of different writings v, and also their relative frequency p_v:

$$H(V) = -\sum_{v \in V} p_v * log(p_v)$$

In a corpus with a strictly normalized orthography (and without any typo), each gloss would have an entropy of 0. If a writing variation is very rare compared to the others, the entropy will 'weight' the relative importance of this uncommon spelling accordingly. Table 1 illustrates spelling variations found in the SMS4science corpus. The word "nächste" (*next*) has the highest writing entropy (H=4.1) of all words.

Fig. 1 shows the overall distribution of entropy plotted against the frequency of glosses and illustrates the broad range of variations. This figure only reports about words that contain at least one alphabetic character. Out of 5,963 different types that fulfill this condition, 2,941 (49%) show no variation and 3,022 show at least 2.

ArchiMob The *ArchiMob Corpus*[5] (Samardzic et al., 2016) consists of 34 transcribed interviews (528,381 tokens) with Swiss citizens who witnessed the Second World War. The recordings are taken from the Archimob[6] oral history collections, which contain 555 videos, out of which 300 are in Swiss German.

The compilation of the ArchiMob Corpus started in 2004 and the three transcription phases extended over a period ten years. For the different phases, not only the tools but also the guidelines changed. The guidelines follow roughly the Dieth script (Dieth, 1986) but do not make use of all available phonemic distinctions. The grave accents in "*nòchethèèr*" ('*afterwards*') mark open vowels, more examples can be seen in Tab. 1. Because this distinction was dropped in later phases, we removed these grave markers for the data shown in Fig. 2.

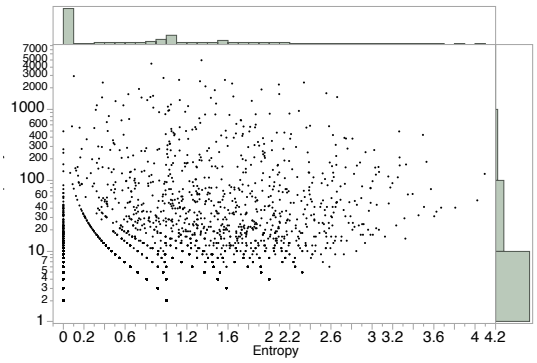

Figure 1: SMS4science corpus: plot of frequency of normalized words (y axis) against their writing variation entropy in Swiss German

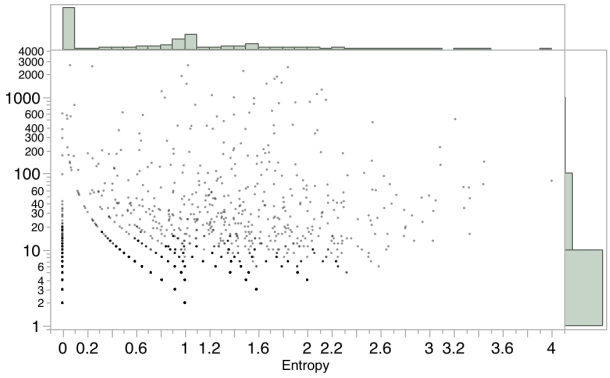

Figure 2: Archimob corpus: plot of frequency of normalized words (y axis) against their writing variation entropy in Swiss German

Furthermore, it has to be noted that not only the interviewees but also the transcribers have different dialectal backgrounds which, for instance, has an impact on the perception of vowels, leading to variations in transcriptions. Transcription variation has two sources: different dialects can use different words to refer to the same concept, and the same word can be pronounced and spelled differently. This results in a great number of potential variations which need to be reduced to a single canonical form in order to identify word variants. The general normalisation procedure is to transform every Swiss German word into the cor-

the query to tokens with a non-empty succeeding token.

[4] As can be seen in Tab. 1 in the row for "weekend", some abbreviations were not marked as such.

[5] www.spur.uzh.ch/en/departments/korpuslab/Research/-ArchiMob.html

[6] www.archimob.ch

Variations	*Alignment Output*
chaschmers sägä	`chasch-mers s-ägä`
chasch mirs sääge	`chasch mirs sääge`
can-you(-)me-it tell	Minimal Edit Distance = 4

Table 2: Pairwise Needleman/Wunsch sequence alignment of Swiss German transcriptions

responding standard German version following an etymological principle. Morphosyntactic features in Swiss German lexemes that are not implemented in standard German are transformed into morphologically transparent normalisations. For instance *dure* (through) does not exist in standard German, it would correspond to *durch* + *direction*, so it was normalised as *durchhin*.

At the current stage, only 6 recordings have normalizations attached to each word (93,455 tokens). For our evaluation, we dropped all fused words (2,915 tokens), which we identified by whitespace characters inside the normalization string. About 869 tokens did not have a valid normalization. For measuring the entropy, only words containing at least one alphabetic character were included. Out of 3,352 different types fulfilling this condition, 1,428 (43%) have no variation and 1,924 (57%) have at least 2 different spellings. Fig. 2 shows the observed spelling entropy, which in the case of ArchiMob should only express phonemic distinctions rather than personal writing and spelling habits.

Discussion Interestingly, Fig. 1 and 2 show a similar distribution although the underlying data was produced quite differently. These resources can be used for further explorations of typical pronunciation and writing variations in Swiss German. However, they both cannot be used to systematically correlate these variations with factors that might influence them. For the text messages, we are missing the phonetic form although we have real user-generated text. For the linguistic transcriptions, we are missing spelling variants, which native writers would produce. Therefore, we will collect new data in order to answer our research question.

Crowdsourcing Writing Variations The goal of our current project is to use a citizen science approach for collecting written Swiss German utterances as well as their standard German normalizations. Similar to the ArchiMob setup but different from the SMS4science setup, we will have spoken audio material that will be transcribed. However, the same material will be written in a spontaneous user-generated style (no guidelines, just the way they would write it in private communication) by several lay transcribers, which are to be recruited via a corresponding gaming platform on which users are able to locate Swiss German dialects with the help of the aforementioned audio stimuli.

These lay transcriptions give us the opportunity to assess the broad spectrum of spelling variations that is perceived as an adequate rendering of spoken Swiss German, and at the same time, correlate it with sociolinguistic factors that we assume to be relevant: (a) the dialect of the speaker and the transcriber (and their closeness), (b) the age and gender of the transcribers, (c) their expertise in writing in dialect. Accordingly, we are mainly interested in variation due to these social variables and not looking at variation caused by the medium or technical means, because we probably could not control the impact of the latter.

The consistency and variability of the independent parallel transcriptions can then be assessed automatically in a more fine-grained way. Character sequences can be aligned pairwise using sequence alignment algorithms (Needleman and Wunsch, 1970) as illustrated in Tab. 2.

We will also collect standard German "translations" of the Swiss German utterances, however, there will be no interlinear glossing in the style of SMS4science. Automatic normalization should be feasible given the available resources from SMS4science and ArchiMob, as shown in Samardzic et al. (2015; 2016).

User Interface Challenges for Transcription Transcribing audio recordings is a tedious and time-consuming task, especially for volunteering non-specialists. In the context of a web-based crowdsourced transcription project, volunteers should be extensively assisted in their transcribing task, or they would quickly give up. Usual facilitation for expert transcribers are all-in-one transcription software, or a USB pedal for convenient rewinding or slowing down of the speech rate, but none of them could apply here.

We will provide a simplified audio player with the usual facilities of playing and pausing as well as full and partial rewinding. Instead of displaying a continuous speech wave with a synchronized cursor

moving along the timeline, we represent the audio sample as consecutive blocks of speech segments. These speech units are pause-separated prosodic phrases, which corresponds to an average short-term memory span for audio transcription (Gentilucci and Cattaneo, 2005). As our audio material consists of about 1,200 15-to-20-second samples, the segmentation is automatically pre-computed with pause detection techniques[7] and should yield subsegments of 2-to-5 seconds for each sample. In the web interface, the user is able to play the full sample (with pausing at will) as well as to play segments individually. The current segment is highlighted. Eventually, simple keyboard shortcuts to avoid switching between keyboard and mouse are also available to enhance the user experience.

4 Conclusion

Systematically assessing factors of writing variation of Swiss German needs new resources that involve several transcriptions of the same audio stimulus. When dealing with highly user-specific writing habits, crowdsourcing transcriptions seems a natural approach for data collection. Independent transcriptions and their related sociolinguistic metadata enables us to investigate this phenomenon quantitatively. From an NLP perspective, acquiring more training material for automatic normalization of Swiss German is an important side effect.

Acknowledgments

This research was supported by the Swiss National Science Foundation under grant CRAGP1_164811/1 through the project "Citizen Linguistics: locate that dialect!" We would also like to thank the anonymous reviewer for his helpful comments on the first version of this paper.

References

[Aschwanden2001] Brigitte Aschwanden. 2001. »wär wot chätä?« zum sprachverhalten deutschschweizerischer chatter. online `http://www.mediensprache.net/networx/networx-24.pdf`.

[Christen2004] Helen Christen. 2004. Dialektschreiben oder sorry ech hassä text schribä. In *Alemannisch im Sprachvergleich. Beiträge zur 14. Arbeitstagung für alemannische Dialektologie in Männedorf (Zürich) vom 16.-18.9.2002*, ZDL-Beiheft 129, pages 71–85, Wiesbaden. Franz Steiner Verlag.

[Dieth1986] Eugen Dieth. 1986. *Schwyzertütschi Dialäktschrift: Dieth-Schreibung*. Lebendige Mundart. Sauerländer, Aarau etc. 2. Aufl. / bearb. und hrsg. von Christian Schmid-Cadalbert (1. Aufl. 1938).

[Dürscheid and Stark2013] Christa Dürscheid and Elisabeth Stark. 2013. Anything goes? sms, phonographisches schreiben und morphemkonstanz. In Martin Neef and Carmen Scherer, editors, *Die Schnittstelle von Morphologie und geschriebener Sprache*, Linguistische Arbeiten, pages 189–210. De Gruyter, Berlin.

[Gentilucci and Cattaneo2005] Maurizio Gentilucci and Luigi Cattaneo. 2005. Automatic audiovisual integration in speech perception. *Experimental Brain Research*, 167(1):66–75.

[Goldman2011] Jean-Philippe Goldman. 2011. Easyalign: an automatic phonetic alignment tool under praat. In *Proceedings of the 12th Annual Conference of the International Speech Communication Association*, pages 3233–3236, Florence, Italy.

[Kolde1981] Gottfried Kolde. 1981. *Sprachkontakte in gemischtsprachigen Städten. Vergleichende Untersuchungen über Voraussetzungen und Formen sprachlicher Interaktion verschiedensprachiger Jugendlicher in den Schweizer Städten Biel/Bienne und Fribourg/Freiburg i.Ue.* Franz Steiner Verlag, Wiesbaden.

[Marti1985] Werner Marti. 1985. *Berndeutsch-Grammatik für die heutige Mundart zwischen Thun und Jura*. A. Francke, Bern.

[Needleman and Wunsch1970] S B Needleman and C D Wunsch. 1970. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *J Mol Biol*, 48(3):443–53.

[Ruef and Ueberwasser2013] Beni Ruef and Simone Ueberwasser. 2013. The taming of a dialect: Interlinear glossing of swiss german text messages. In Marcos Zampieri and Sascha Diwersy, editors, *Non-standard Data Sources in Corpus-based Research*, volume 61-68 of *ZSM-Studien 5*. Shaker, Aachen.

[Samardzic et al.2015] Tanja Samardzic, Yves Scherrer, and Elvira Glaser. 2015. Normalising orthographic and dialectal variants for the automatic processing of swiss german. In *Proceedings of the 7th Language and Technology Conference: Human Language Technologies as a Challenge for Computer Science and Linguistics*.

[Samardzic et al.2016] Tanja Samardzic, Yves Scherrer, and Elvira Glaser. 2016. Archimob - a corpus of spoken swiss german. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, pages 4061–4066.

[7] Using tools like EasyAlign (Goldman, 2011) or WebMAUS (Strunk et al., 2014).

[Schmid and Laws2008] Helmut Schmid and Florian Laws. 2008. Estimation of conditional probabilities with decision trees and an application to fine-grained POS tagging. In *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)*, pages 777–784, Manchester, UK.

[Schmid1995] Helmut Schmid. 1995. Improvements in part-of-speech tagging with an application to german. In *Proceedings of the EACL SIGDAT-Workshop*. (überarbeitete Version).

[Siebenhaar2003] Beat Siebenhaar. 2003. Sprachgeographische aspekte der morphologie und verschriftung in schweizerdeutschen chats. *Linguistik Online*, 15(3):125–139.

[Siebenhaar2006] Beat Siebenhaar. 2006. Gibt es eine jugendspezifische varietätenwahl in schweizer chaträumen? In *Perspektiven der Jugendsprachforschung/Trends and Developements in Youth Language Research*, Sprache – Kommunikation – Kultur 3, pages 227–239. Lang, Frankfurt a.M.

[Siebenhaar2008] Beat Siebenhaar. 2008. Quantitative approaches to linguistic variation in irc: Implications for qualitative research. *Language@Internet*, 5(4).

[Sieber2010] Peter Sieber. 2010. Deutsch in der schweiz: Standard, regionale und dialektale variation. In *Deutsch als Fremd- und Zweitsprache. Ein internationales Handbuch*, HSK 35.1, pages 372–385. de Gruyter, Berlin, New York.

[Strunk et al.2014] Jan Strunk, Florian Schiel, and Frank Seifart. 2014. Untrained forced alignment of transcriptions and audio for language documentation corpora using webmaus. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*.

Brown clustering for unlexicalized parsing

Daniel Dakota
Indiana University
Ballantine Hall 844
Bloomington, IN 47405-7005
ddakota@indiana.edu

Abstract

Brown clustering has been used to help increase parsing performance for morphologically rich languages. However, much of the work has focused on using clustering techniques to replace terminal nodes or as a feature for parsing. Instead, we choose to examine how effectively Brown clustering is for unlexicalized parsing by creating data-driven POS tagsets which are then used with the Berkeley parser. We investigate cluster sizes as well as on what information (e.g. words vs. lemmas) clustering will yield the best parser performance. Our results approach the current state of the art results for the German TüBa-D/Z treebank when using parser internal tagging.

1 Introduction

Part of Speech (POS) tags are an essential aspect of any annotated corpus, in particular for treebanks. However, the development of optimal tagsets for a given language is still problematic. The granularity of the linguistic information has both practical and theoretical aspects, but the chosen tagset has direct consequences on performance of a given task, especially to parsing.

The argument can be made that regardless of the morphological complexity of a language, there still only exists a set of primary POS tags. This has resulted in the creation of simplified, coarse-grained tagsets, most notably the Universal Tagset (Petrov et al., 2012) consisting of only 12 primary POS tags. However, this oversimplifies the linguistic complexity of a language. Subsequently, too fine-grained of a tagset also results in a decrease in parser performance (Maier et al., 2014). Although statistical methods for parsing have improved over the past decade, the issue of complex morphology and its direct impact on parsing performance still remains. This is most evident in morphologically rich languages (MRLs) where a single form of a word may have dozens of surface forms. This has resulted in expanded tagsets for many languages that possess more morphology than English, as well as the addition of morphological information directly attached to the tags, which increases both the tagset size and the level of granularity.

With the creation of any tagset, how much linguistic information is relevant becomes a matter of debate. This has traditionally required a discussion about how best to incorporate the relevant linguistic information in order to categorize and sub-categorize various POS into a tagset. We choose to approach this problem by examining whether we can empirically and automatically create POS tags utilizing Brown clustering (Brown et al., 1992), and how effectively these tagsets can be used for parsing. By doing so, we group words together contextually and are able to add additional linguistic information into the process, which reduces the need to manually group morphologically complex words into various linguistic categories. We experiment with the granularity of these tags by clustering words, lemmas, and lemmas with morphological information and subsequently examine to what extent these tagsets still mimic linguistic categories. We utilize the unlexicalized Berkeley parser (Petrov and Klein, 2007) to examine the impacts of these tagsets on parsing performance of the German TüBa-D/Z treebank (Telljohann et al., 2015). Results fall in line with previous research on tagset granularity and show empirically created tagsets can come close to matching our established baseline using pre-defined tags as well as state of the art results when using parser predicted tags for parsing.

The remainder of the article is structured as follows. In section 2, we review previous work on clustering and POS tagset granularity. Section 3 presents the task while section

4 describes our experimental setup. Parsing results and discussion are presented in sections 5 and 6 before section 7 concludes the article.

2 Related Work

2.1 POS Tag Set Granularity

The granularity of a POS tagset is an important aspect of parsing since it directly impacts the parsing performance. English POS tags have continued to be based on the 36 tagset of the Penn Treebank (Marcus et al., 1993), but this has not confined other languages to such tag limits. For German, there is the 54 STTS tagset (Schiller et al., 1995) which can have morphological information attached to the tags increasing the maximum tagset size into the hundreds. This is a common strategy for many tagsets for MRLs which demonstrate much higher degrees of morphology. However, what morphology is optimal for improved parser performance for any given language has not been definitively determined, as the increase in the tagset subsequently increases sparsity of tags which influences the parser.

Although less granular POS tagsets can achieve a high rate of tagging accuracy, this does not necessarily mean they convey enough information for parsing. This was demonstrated by Maier et al. (2014) who utilized the Berkeley parser to tag and parse two German treebanks with three tagset variants, the UTS (Petrov et al., 2012) consisting of 12 tags, the STTS tagset (Schiller et al., 1995) consisting of 54 tags, and the STTS with morphological information resulting in hundreds of tags. Although the use of the UTS tagset resulted in the highest POS accuracy, it did not obtain the highest parsing performance which was obtained by the use of the STTS tagset.

Additionally, Marton et al. (2013) found that particular linguistic information (e.g. person, number, gender) for finer-grained tagsets can be useful when utilized as a gold POS tag for dependency parsing of Arabic, but detrimental when predicted by the parser internally, which benefits from coarser grained tagsets.

Seddah et al. (2009) investigated two tagsets with different granularity on French treebanks and concluded that the granularity of the tagsets can improve results, but with each improving either dependency or constituency parsing results respectively over the other.

2.2 Clustering

Clustering has been used in document classification, but there has also been an increase in its utilization to other areas of NLP such as to help improve POS tagging for Twitter (Owoputi et al., 2010). More recently it has been utilized in parsing to help reduce data sparsity, as statistical parsing suffers from data sparseness, particularly when parsing MRLs which have a higher ratio of word forms to lemmas (Tsarfaty et al., 2010).

Most, if not all, work has focused on replacing terminal nodes with clusters IDs or by using clusters as a feature for dependency parsing. Clustering has been shown to reduce sparsity issues, resulting in increased parser performance. Koo et al. (2008) showed that using Brown clustering to create cluster-based feature sets outperformed the baseline models in both English and Czech dependency parsing.

However, for MRLs how best to use Brown clustering to improve parsing performance is still unclear as clustering on words, lemmas, or lemmas with additional morphological information has yielded various results. Candito and Crabbé (2009) clustered what they termed *desinflected* French words. They removed unnecessary inflection markers using an external lexicon and then combined the *desinflected* form with additional features and replaced terminal nodes with the cluster ID. Although this increased French parsing performance with the Berkeley parser and improved results for both medium and higher frequency words (Candito and Seddah, 2010), the results were comparable to clustering the lemma with the predicted POS tag of the word.

Candito et al. (2010) found that replacing terminals with clustering-based features improved results for the Berkeley parser but not substantially for dependency parsers. Related work by Ghayoomi (2012) and Ghayoomi et al. (2014) used Brown clustering with POS information to resolve homograph issues in Persian and Bulgarian respectively to significantly improve class-based lexicalized parsing results over word-based parsing. Goenaga et al. (2014) created word clusters using both words (for Swedish) and lemmas with morphological information (for Basque) to create features of varying granularities for use in dependency parsing with noticeable improvements. Such findings are supported by Versley (2014) who noted that cluster-based features improved discontinuous constituent

parsing results for German considerably, but were also influenced by the granularities of the feature (i.e. a sequence of 0s and 1s to which every word is assigned indicating the cluster ID with shorten bit-strings representing more general, larger subsets of clusters).

3 Task

The question of how best to determine the granularity for POS tags for optimal parsing continues to persist for many languages, which is made more problematic by language-specific linguistic phenomena. We choose to investigate whether we can create empirically optimal tagsets using Brown clustering and obtain results similar to pre-defined tagsets. As has been shown, clustering has yielded positive results in parsing. However, much of the work has replaced terminal nodes with class-based representations. This has been demonstrated to be useful for lexicalized parsing, but for unlexicalized parsing, although improvements have been shown, the extent to which clusters can be utilized has been minimized. Terminal nodes (i.e. words) are only utilized for unlexicalized parsing when the parser needs more information than just using the tags, thus how often the terminals influence the parser is minimized when the POS tagging accuracy is high. For this reason, we choose to replace POS tags. During the clustering process, we examine the impact of word frequencies, clustering sizes, and granularity of information at the word level on parsing performance. German possesses a richer morphology than English, allowing for different linguistic phenomena that effect parser performance such as case. In particular, German morphology allows for a much freer word order than English, but not as free as other MRLs. For example, articles are inflected for case and gender allowing subjects, direct objects, and indirect objects to freely move in the sentence. One inherent complexity of German morphology is case syncretism. This is seen with articles where the case and gender for one object can mimic another (e.g. *die* is both the definite nominative feminine and definitive accusative plural). This means that grammatical functions improve the usefulness of a parse (Rafferty and Manning, 2008) but that they cannot be determined strictly by their position in the tree (Kübler, 2008).

4 Experimental Setup

4.1 Treebank

We use the German treebank TüBa-D/Z version 10.0 (Telljohann et al., 2015), taking the first 90% for training and performing a 3-fold cross-validation. Each fold consists of 57357 training sentences and 28678 for testing. The final 10% percent was left out for testing after further experiments have been run. The treebank was pre-processed by replacing all grammatical function (GF) dash separators with a "#" and collapsing all occurrences of label-internal dash separators (e.g. R-SIMPX $\rightarrow$ RSIMPX). This was done as the Berkeley parser treats anything after a "-" as a grammatical function and cuts it off.

4.2 Parser

For parsing, we use the Berkeley parser (Petrov and Klein, 2007). The parser is ideal to examine the impact of POS tags as it is unlexicalized. The Berkeley parser uses a system of split/merge cycles that should help to smooth over the variation in the tagset sizes. We evaluate using standard EVALB (Sekine and Collins, 1997) including grammatical functions, using a parameter file to delete VROOT. Non-parsed sentences are not calculated in the evaluation metrics, but we provide their number in the results.

4.3 Word Clustering

We use Brown Clustering (Brown et al., 1992) using the implementation from Liang (2005). Brown clustering is an unsupervised clustering method that obtains a pre-specified number of clusters (C). It assigns the C most frequent word tokens to their own cluster. Every subsequent word is assigned to one of the clusters by creating a new cluster and merging the C+1 cluster with an already defined cluster that minimizes the loss in likelihood of the corpus based on a bigram model determined from the clusters. Brown clustering is a hard clustering algorithm, thus the previous step is repeated for each subsequent word until every word is assigned a cluster, resulting in words having been clustered based on their contextual similarity to one another. The final product is a binary hierarchical structure with each cluster being represented by a bit-string of varying lengths. We cluster using a German wikipedia dump consisting of approximately 175 million words (Versley and Panchenko, 2012), which was also tagged with both POS infor-

mation and morphological information using Mate Tools (Björkelund et al., 2010).

By using Brown clustering, we are empirically creating tagsets that allow for words to be grouped together based on contextual similarity. This also allows for words normally assigned to the same linguistic category (e.g. nouns) to be possibly assigned to different clusters because their contextual similarity differs enough as defined by the clustering algorithm. This subsequently allows for a finer distinction of categories of words than would naturally be assumed. We replace POS tags in the treebank by looking up whether the word has a cluster ID and replacing it with the full bit-string. Any word in the treebank without a cluster ID was given a tag of '0' symbolic of an unknown tag. All punctuation was replaced with a single '-PUNCT-' in order to reduce the overall number of tags. This means for every cluster size C, the true number of tags in the set is C+2. We performed an initial experiment between words and lemmas in order to determine which of the two are a better basis for clustering tags. Since Brown clustering has different thresholds, we examined different minimum frequency of lemmas in the clustering corpus to examine a) what impact decreasing the minimum frequencies has on coverage and performance and b) whether there is a minimum frequency after which there are no longer improvements in results. Finally we performed two additional experiments by adding morphological information to the lemmas. The first experiment added both selected POS tag information and morphological information from Mate Tools (Björkelund et al., 2010) to the lemma. This was done to examine whether the use of some pre-defined STTS tag information with additional morphological information can be utilized in the clustering processing, as it adds additional German-specific linguistic information. The second experiment attached only morphological information to the lemmas. The list of selected tags are presented in Table 1. These tags were selected based on morphological information and not every possible STTS tag was selected. In particular, we focused on tags that tend to represent words that are inflected for case and gender (i.e. articles, adjectives, and personal pronouns). We also chose to simply assign all verbs a single VERB tag. This was done as verbs are particularly challenging to label for granularity in any given language. A summary of the selected tags with morphological information are presented in Table 2.

Name	Description
ART	article
ADJA	adjectives
PRELS	substituting relative pronoun
PIS	substituting indefinite pronoun
PPOSAT	attributive possessive pronoun
PPER	irreflexive personal pronoun
VERB	all verbs given simply VERB

Table 1: The selected POS tags for experiment 1

Name	Description
art+case	attach case to articles
art+gend	attach genders to articles
art+case+gend	attach both case and gender to articles
infl+case+gend	attach case and gender to all lemmas if applicable
verb+person	attach person to verbs
verb+num	attach number to verbs
verb+person+num	attach person and number to verbs
all	all features

Table 2: Description of Lemmas+Features used for clustering

Recall	Precision	F-score	POS Acc.	Unparsed Sent.
83.12	82.93	83.02	97.5	4

Table 3: Average results for 3-fold baseline with STTS tags

N1	N2	N3	F-score Average
83.53	83.25	82.29	83.02

Table 4: Individual F-scores for 3-fold baseline with STTS

4.4 Baseline

We establish a baseline by using the STTS tagset for the TüBa-D/Z treebank and report the average recall, precision, F-score for parsing, and POS accuracy which is calculated by comparing every tag in the gold and test files (Table 3). This was done as a basis of comparison for our experimental setup since there exists no previous findings which we can directly compare our results against. Table 4 provides the F-scores for each fold of the baseline. The varying results on each fold is consistent with other findings (see Levy and Manning (2003)) that have noted that any given section of a treebank may be more or less difficult to parse relative to another section. Here later portions of the treebank are inherently harder to parse.

Cluster Size	Recall	Precision	F-score	POS Acc.	Unparsed Sent.
25	78.16	78.89	78.52	95.10	8
50	79.24	79.85	79.54	95.25	6
75	79.05	79.56	79.30	95.39	2
100	79.41	79.73	79.57	95.34	4
125	79.50	79.75	**79.62**	95.52	3
150	79.35	79.67	79.51	95.59	4
175	79.38	79.64	79.51	95.67	17
200	79.29	79.47	79.38	**95.74**	16

Table 5: Words used as tag with a min. frequency of 100

Cluster Size	Recall	Precision	F-score	POS Acc.	Unparsed Sent.
25	79.17	79.67	79.42	93.32	3
50	79.75	80.15	**79.95**	93.26	4
75	79.57	79.95	79.76	**93.41**	1
100	79.60	79.90	79.74	93.17	3
125	79.77	80.03	79.90	93.03	10
150	79.49	79.56	79.53	93.10	5
175	79.84	79.87	79.85	93.18	6
200	79.34	79.32	79.33	93.16	5

Table 6: Lemmas used as tags with a min. frequency of 100

Min Occurrence	% of Words	% of Word Types
100	88.5	30.2
50	90.9	39.3
20	93.3	51.2
3	96.4	70.5
1	97.4	78.5

Table 7: The percentage of words and word types found in TüBa-D/Z from clustering corpus

Min Occurrence	% of Lemmas	% of Lemma Types
100	89.9	31.3
50	91.8	40.3
20	93.6	51.3
3	96.1	69.1
1	96.9	76.8

Table 8: The percentage of lemmas and lemma types found in TüBa-D/Z from clustering corpus

5 Results

5.1 Word vs. Lemma

When comparing POS tags created strictly on the words (Table 5) versus tags created on lemmas (Table 6) in all cases, except for a cluster size of 200, lemmas outperform words. However, for tags created on words, the highest F-score is obtained using a cluster size of 125, whereas for lemmas, the highest F-score is obtained with a cluster size of 50. Interestingly, the POS accuracy for word clusters increases with the cluster size which stands in contrast to the POS accuracy for lemmas, which tends to decrease in accuracy as the cluster size increases. A cluster size of 200 trained on just words obtained the highest POS accuracy of any of our experiments at 95.74%. However, this is consistent with the findings from Maier et al. (2014) that a higher POS accuracy does not necessarily result in the best parsing performance. This is further supported by the lower POS accuracies of the equivalent lemma POS cluster sizes which although lower, demonstrate a consistently higher F-score. None of the results reach our baseline; the closest, a cluster size of 50 using lemmas, is still more than 2.5% absolute below the baseline.

In order to investigate the coverage of clusters on words and lemmas in the treebank, we extracted the percentage of words and lemma tokens covered by the clusters, as well as extracting type coverage, the results of which are presented in Tables 7 and 8. We do not include punctuation, since stand-alone punctuation is not utilized during Brown clustering. Using a minimum frequency of 100 in the Wikipedia data, the resulting clusters cover 88.5% of total word tokens in the treebank, but represents merely 30.2% of all word types. Decreasing the minimum frequency to 1 increases coverage of the overall corpus to about 97% for both the raw words and lemmas but still about 25% of types are not covered. Using lemmas instead of word forms does not alter coverage percentages substantially. This is surprising given that reducing words to their lemmas should help decrease sparsity but the overall coverage between unlemmatized forms and their lemmas is comparable. However, by lemmatizing we increase the frequency of a given token type in the data which should help the parser, as given cluster tags will occur more frequently.

5.2 Lemmas

Noting the slightly better performance of lemmas over words, experiments were conducted clustering on lemmas but reducing the minimum frequency of a lemma for clustering to 50 times and 3 times, as presented in Tables 9 and 10 respectively. We choose not to utilize a minimum frequency of 1 to help reduce the number of possible typos or erroneous words for clustering given the nature of web data. We can see a general rise in F-scores as the minimum frequency of a lemma's occurrence for clustering decreases. However, it is not absolute, as there are several instances in which a higher minimum frequency outperforms a lower minimum frequency. This can be seen in Table 10 where the F-score for minimum frequency lemma of 3 with a cluster size of 50 is lower than the F-score in Table 6 for minimum lemma frequency of 100 for a cluster size of 100, which was the highest performing

Cluster Size	Recall	Precision	F-score	POS Acc.	Unparsed Sent.
25	79.48	79.95	79.72	**93.15**	0
50	80.03	80.39	80.21	92.79	1
75	80.13	80.46	**80.30**	92.96	1
100	79.69	79.98	79.83	92.67	3
125	79.71	79.94	79.82	92.47	0
150	79.79	79.79	79.79	92.53	5
175	79.36	79.35	79.36	92.54	11
200	79.81	79.75	79.78	92.50	6

Table 9: Lemmas used as tags with a min. frequency of 50

Cluster Size	Recall	Precision	F-score	POS Acc.	Unparsed Sent.
25	79.85	80.29	80.07	**93.63**	5
50	79.25	79.61	79.43	92.86	570
75	80.20	80.49	80.34	92.93	5
100	80.38	80.58	**80.48**	92.51	2
125	79.82	80.02	79.92	91.96	7
150	79.87	79.84	79.86	91.51	5
175	79.96	79.94	79.94	91.37	7
200	79.74	79.65	79.70	91.38	4

Table 10: Lemmas used as tags with a min. frequency of 3

cluster size for a minimum lemma frequency of 100. The overall trend of increased performance is supported with the percentages presented in Table 8 that showed slight increases in token coverage, but larger increases in type coverage as the minimum frequency decreases for lemmas to be clustered. On average only a few sentences are not parsed, but an anomaly occurs in Table 10 where a cluster size of 50 resulted in 570 sentences not being parsed. A reason for this has not been identified.

5.3 Lemma + Morphology

To examine the effect of adding morphological information to lemmas, we select the highest obtained F-score of 80.48% , which was with lemmas with a minimum frequency of 3 and a cluster size of 100. The results for adding selected POS tags plus morphology are presented in Table 11. Adding lemma and morphological information alters results, in some cases significantly. By simply adding the STTS article tag and case information, there is a decrease of almost 4% absolute. However, when adding person information to the verb, there is an increase in performance in both experiments. Interestingly, when using the VERB tag, the F-score is further increased when combining person and number, even though VERB tag and number information alone decreases performance from just the lemma. In contrast, when not using a VERB tag, adding number information decreases performance. Combining all the features reduces overall performance in both experiments.

Cluster Size	Recall	Precision	F-score	POS Acc.	Unparsed Sent.
art+case	80.07	73.53	76.67	92.14	3
art+gend	79.93	80.14	80.03	**92.42**	4
art+case+gend	80.1	80.21	80.15	92.16	3
infl+case+gend	79.37	79.55	79.46	92.13	6
verb+person	80.64	80.60	80.62	**92.42**	9
verb+num	80.04	80.08	80.06	92.37	4
verb+person+num	80.80	80.76	**80.78**	92.20	16
all	80.08	80.01	80.04	89.83	11
best word performance	79.50	79.75	79.62	95.52	3
best lemma performance	80.38	80.58	80.48	92.51	2

Table 11: Results for lemmas and selected POS tags with morphology

Cluster Size	Recall	Precision	F-score	POS Acc.	Unparsed Sent.
art+case	79.86	73.49	76.54	92.16	7
art+gend	79.93	80.14	80.03	92.42	4
art+case+gend	80.1	80.21	80.15	92.16	3
infl+case+gend	79.69	79.94	79.81	**92.87**	6
verb+person	81.11	81.04	**81.08**	92.42	5
verb+num	79.98	80.56	80.27	92.37	2
verb+person+num	80.80	80.76	80.78	92.20	16
all	79.06	79.06	79.06	90.81	8
best word performance	79.50	79.75	79.62	95.52	3
best lemma performance	80.38	80.58	80.48	92.51	2

Table 12: Results for only lemmas and morphology

6 Discussion

Currently state of the art results for German constituency parsing for the Berkeley parser on TüBa-D/Z is an F-score of 83.97 (Petrov and Klein, 2008), however this was done using Gold POS tags. We compare results to our own baseline using the STTS tagset as well as noting consistencies found by Maier et al. (2014).

Examining Tables 11 and 12 we see that in some cases we are able to increase the F-score by adding morphological information into the clustering process, but in other cases there is a decrease in performance. Simply adding case information to articles significantly decreases performance for both experiment. This can partially be attributed to the case syncreticsm seen in German. Our decision to treat all verbs with a single coarse-grained POS tag while selecting finer grained STTS tags for tags containing case and information most likely influenced the results between the two linguistic categories. This suggests that coarse tags may be slightly more beneficial when combined with morphological information. Overall, our results are consistent with issues regarding tag granularity and parsing performance.

Interestingly, there are three identical sets of results in the experiments. This could indicate that these particular morphological features are more important than the granularity of the tag itself (i.e. detailed information of the verb is not as important as the person and number information of the verb).

We are not able to match our baseline F-score of 83.02 using the original STTS tagset. However,

Tag	Recall	Precision	F-score	% in Gold	Majority Tags
0	75.58	86.39	80.62	11.09	unknown
00010010	99.43	99.89	99.66	6.11	nouns/adjectives
0010	92.55	88.36	90.41	5.00	proper nouns
0001001110	98.20	98.59	98.39	4.88	3rd person verbs
11000	90.82	78.76	84.36	4.08	nouns
000100110	91.45	85.66	88.46	3.29	3rd person verbs
01011	88.03	85.14	86.56	2.91	nouns
110111	89.33	81.04	84.98	2.89	nouns
11010	88.69	86.06	87.35	2.58	nouns
00011110	99.69	99.79	**99.74**	2.49	mixed

Table 13: POS Tag Analysis of fold 3 for lemmas and selected POS tags with morphology

Tag	Recall	Precision	F-score	% in Gold	Majority Tags
0	87.08	92.97	89.93	19.32	unknown
10010100	99.61	99.95	99.78	6.10	mixed
1010	92.59	88.39	90.44	5.05	proper nouns
1001010110	99.92	99.96	**99.94**	4.14	3rd person verbs
0111	90.62	79.09	84.47	3.96	nouns
0101	88.46	84.39	86.38	2.89	nouns
0010	90.23	79.39	84.47	2.89	nouns
000	88.11	85.40	86.73	2.56	nouns
10011110	99.60	99.71	99.65	2.49	mixed
10000010	89.36	88.10	88.73	1.85	adjectives

Table 14: POS Tag Analysis of fold 3 for only lemmas with morphology

our results do show that is possible to create empirically driven POS tags that are created using Brown clustering that can approach results using a predefined tagset as our best results perform only less than 2% absolute lower than our baseline. Furthermore we can individually demonstrate the effects of a single piece of morphological information has on parsing performance. This provides further evidence that there is a balance between granularity and optimal performance. Given the selective nature of what morphology we chose to add to the lemmas, it is possible that a different combination of morphological information may further improve results. Additionally, our results further reinforce that a high POS accuracy does not necessarily correlate to a higher parsing performance. In both experiments, the experiments achieving the highest POS accuracy did not obtain the highest parsing results.

In an attempt to ascertain what sort of clusters are more accurate in terms of tagging than others, an analysis was performed on individual tags. However, given the nature of clustering, it is difficult to provide too much detailed information on the clusters themselves, but rather one can extrapolate general patterns within the clusters by examining them manually.

In Tables 13 and 14 we present the top 10 most frequent POS tags from the 3rd fold from the "all" experiments of the results in Tables 11 and 12 by using the EVALB implementation in Disco-dop (van Cranenburgh et al., 2016) which provides more detailed POS tag information. We also provide what we manually identified as the majority tag (i.e. a manually assigned POS tag based on the majority of words in the cluster).

The 'unknown' tag of '0' indicating that the word did not have a cluster constitutes more than 11% of the overall tags in the fold in Table 13. As seen in Table 8, this is a higher than expected percentage given that only about 4% of the lemma tags in the entire treebank are not found in the clusters. However, this can be attributed to the addition of morphological information to the lemmas. Certain tags are tagged with a very high degree of accuracy at over 99%, while other tags are more difficult for the parser. We can assume however, that if 10% of the tags in the entire treebank are only accurately tagged 80% of the time (e.g. the '0' tag), this will introduce problems for the parser leading to a decrease in parser performance. Worth noting is that although the '0' tag constitutes almost 20% of the treebank when not using POS tag information, the F-score is 9% absolute higher. This may suggest that it is easier for the parser to correctly tag unknown words using morphological information over POS information. To help further reduce the number of unknown tags in future experiments, it may be beneficial to add the treebank corpus into the clustering corpus, as well as additional domain specific texts to help increase domain specific type coverage. By simply adding the lemmatized TüBa-D/Z corpus into the clustering data alone, and using a minimum frequency of 3, we can increase the lemma token coverage of the clusters on TüBa-D/Z corpus to 97.4% and the type coverage to 75.1%. This should also help increase parser performance, as out of domain parsing impacts parsing results (Gildea, 2001).

In order to further examine the size and frequency counts of individual clusters, Tables 15 and 16 contain the number of types in each cluster, and the percentage of types with less than or equal to 10 total counts in the clustering corpus.

At first glance, it appears that if the frequency of rare words are relatively high in the cluster, then the accuracy of the tags is higher. Although the two clusters with high rates of less frequent words obtain higher POS tagging rates, this does not mean there is direct association, although it most likely attributes to the higher accuracy. A counter example can be seen however with tag

Tag	Types	POS F-Score	% ≤ 10 Freq
00010010	16842	99.66	81%
0010	145065	90.41	58%
0001001110	4143	98.39	64%
11000	98360	84.36	62%
000100110	4143	88.46	64%
01011	84061	86.56	64%
110111	75296	84.98	63%
11010	90846	87.35	64%
00011110	7888	99.74	87%

Table 15: Cluster analysis of fold 3 for lemmas and selected POS tags with morphology

Tag	Types	POS F-Score	% ≤ 10 Freq
10010100	16428	99.78	75%
1010	147868	90.44	58%
1001010110	4352	99.94	66%
0111	96025	84.47	63%
0101	82531	86.38	64%
0010	76019	84.47	62%
000	89149	86.73	63%
10011110	7719	99.65	87%
10000010	23119	88.73	57%

Table 16: cluster analysis of fold 3 for lemmas with morphology

0001001110 in Table 15. Interestingly, this cluster consists predominantly of 3rd person plural verbs (e.g. gehen.VERB.3p "go") of high frequencies. This is also seen with cluster 10000010 in Table 16. This cluster has a relatively low percentage of rare words compared to the other clusters, but still has a relatively high F-score for tag accuracy. Manually inspecting the cluster reveals that it predominantly consists of adjectives without morphological information.

When further manually examining other tags that demonstrate lower F-scores, it appears that tags that represent clusters consisting of words with a high frequency of common words that have not been tagged with additional morphological information (particularly nouns) are tagged with lower accuracy. When examining the cluster for the least accurate tag 11000 in Table 15, it consists predominantly of common nouns (e.g. Raum "room"). This low accuracy may be due to the decision not to add additional morphological information to nouns (e.g. singular vs. plural) which, if provided, may have increased tagging performance for these clusters. It also confirms that the most frequent words in the clusters have the largest influence on the tagging accuracy regardless of size and proportion of rare words.

7 Conlusion and Future Work

We have shown that we can use Brown clustering to empirically create POS tags for parsing that yield results only slightly below than that of our baseline using the pre-defined STTS tagset, as well as similar results for Berkeley internal tagging and parsing on the German TüBa-D/Z treebank.

We can increase performance by simply clustering on lemmas instead of words to create tags, which can be further increased by adding additional morphological information. However, simply adding even a single piece of morphological information can either reduce or improve results, in some cases drastically. This aligns with previous research indicating that granularity of tags effects parsing performance (Maier et al., 2014; Marton et al., 2013), but further experimentation is still needed in order to better determine how best to incorporate additional morphological information into the POS tagset for clustering to improve parsing performance, and what introduces additional parser errors. However we have demonstrated a possible mechanism for creating empirically driven tagsets possessing different granularities using readily available tools. This allows both the incorporation of linguistic information into the tagsets, but bypasses the need to manually assign words to various finer grained tags and testing how different tagset sizes and granularities affect parsing. In order to improve performance using clustering, we must better understand how language specific clustering techniques need to be utilized. This is compounded by the fact that languages possess starkly different linguistic principles, so optimal settings for German may not work for other MRLs. Similar techniques need to be performed on a set of starkly different languages in order to see if a general pattern emerges, or whether for clustering to be effective, very specific language parameters must be fine-tuned.

Acknowledgments

We would like the thank Wolfgang Seeker and Bernd Bohnet for tagging the clustering corpus with morphological information as well as Djamé Seddah and Yannick Versley for providing the data for clustering and additional pertinent information.

References

Anders Björkelund, Bernd Bohnet, Love Hafdell, and Pierre Nugues. 2010. A high-performance syntactic and semantic dependency parser. In *Proceedings of the 23rd International Conference on Computational Linguistics: Demonstrations*, pages 33–36.

Peter Brown, Vincent Della, Peter Desouza, Jennifer Lai, and Robert Mercer. 1992. Class-based n-gram models of natural language. *Computational Linguistics*, 19(4):467–479.

Marie Candito and Benoît Crabbé. 2009. Improving generative statistical parsing with semi-supervised word clustering. In *Proceedings of the 11th International Conference on Parsing Technologies*, IWPT '09, pages 138–141, Paris, France.

Marie Candito and Djamé Seddah. 2010. Parsing word clusters. In *Proceedings of the NAACL HLT 2010 First Workshop on Statistical Parsing of Morphologically-Rich Languages*, SPMRL '10, pages 76–84, Los Angeles, California.

Marie Candito, Joakim Nivre, Pascal Denis, and Enrique Henestroza Anguiano. 2010. Benchmarking of statistical dependency parsers for French. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, COLING '10, pages 108–116, Beijing, China.

Masood Ghayoomi, Kiril Simov, and Petya Osenova. 2014. Constituency parsing of bulgarian: Word- vs class-based parsing. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 4056–4060, Reykjavik, Iceland.

Masood Ghayoomi. 2012. Word clustering for Persian statistical parsing. In Histohsi Isahara and Kyoko Kanzaki, editors, *Advances in Natural Language Processing*, volume 7614 of Lecture Notes in Computer Science: JapTal 12: Proccedings of the 8th International Conference on Advances in Natural Language Processing, pages 126–137, Kanazawa, Japan.

Daniel Gildea. 2001. Corpus variation and parser performance. In *Proceedings of the 2001 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 167–202, Pittsburgh, PA.

Iakes Goenaga, Koldo Gojenola, and Nerea Ezeiza. 2014. Combining clustering approaches for semi-supervised parsing: the BASQUE TEAM system in the SPRML2014 shared task. In *In First Joined Workshop of Statistical Parsing of Morphologically Rich Language and Syntactic Analysis of Non-Canonical Languages*, Dublin, Ireland.

Terry Koo, Xavier Carreras, and Michael Collins. 2008. Simple semi-supervised dependency parsing. In *Proceedings of ACL-08: HLT*, pages 595–603, Columbus, Ohio.

Sandra Kübler. 2008. The PaGe 2008 shared task on parsing German. In *Proceedings of the Workshop on Parsing German*, PaGe '08, pages 55–63, Columbus, OH USA.

Roger Levy and Christopher Manning. 2003. Is it harder to parse Chinese, or the Chinese Treebank? In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics*, pages 439–446, Sapporo, Japan.

Percy Liang. 2005. Supervised learning for natural language. Master's thesis, Massachusetts Institute of Technology.

Wolfgang Maier, Sandra Kübler, Daniel Dakota, and Daniel Whyatt. 2014. Parsing German: How much morphology do we need? In *Proceedings of the First Joined Workshop of Statistical Parsing of Morphologically Rich Language and Syntactic Analysis of Non-Canonical Languages (SPMRL-SANCL 2014)*, pages 1–14, Dublin, Ireland, August.

Mitchell P. Marcus, Mary Ann Marcinkiewicz, and Beatrice Santorini. 1993. Building a large annotated corpus of English: The Penn Treebank. *Computational Linguistics*, 19(2):313–330.

Yuval Marton, Nizar Habash, and Owen Rambow. 2013. Dependency parsing of Modern Standard Arabic with lexical and inflectional features. *Computational Linguistics*, 39(1):161–194, March.

Olutobi Owoputi, Brendan O'Connor, Chris Dyer, Kevin Gimpel, and Nathan Schnieder. 2010. Part-of-speech tagging for twitter: Word clusters and other advances. Technical report, Carnegie Mellon University.

Slav Petrov and Dan Klein. 2007. Improved inference for unlexicalized parsing. In *Proceedings of Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics*, pages 404–411, Rochester, NY.

Slav Petrov and Dan Klein. 2008. Parsing German with latent variable grammars. In *Proceedings of the Workshop on Parsing German at ACL '08*, pages 33–39, Columbus, Ohio.

Slav Petrov, Das Dipanjan, and Ryan McDonald. 2012. A universal part-of-speech tagset. In *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, pages 2089–2096, Istanbul, Turkey, May.

Anna N. Rafferty and Christopher D. Manning. 2008. Parsing three German treebanks: Lexicalized and unlexicalized baselines. In *Proceedings of the Workshop on Parsing German*, pages 40–46, Columbus, Ohio.

Anne Schiller, Simone Teufel, and Christine Thielen. 1995. Guidelines für das Tagging deutscher Textkorpora mit STTS. Technical report, Universität Stuttgart and Universität Tübingen.

Djamé Seddah, Marie Candito, and Benoît Crabbé. 2009. Cross parser evaluation: A French treebanks study. In *Proceedings of the 11th International Conference on Parsing Technologies (IWPT)*, pages 150–161, Paris, France.

Satoshi Sekine and Michael Collins. 1997. Evalb bracket scoring program. http://nlp.cs.nyu.edu/evalb/.

Heike Telljohann, Erhard W. Hinrichs, Sandra Kübler, Heike Zinsmeister, and Kathrin Beck. 2015. Stylebook for the Tübingen Treebank of Written German (TüBa-D/Z). Seminar für Sprachwissenschaft, Universität Tübingen, Germany.

Reut Tsarfaty, Djamé Seddah, Yoav Goldberg, Sandra Kübler, Marie Candito, Jennifer Foster, Yannick Versley, Ines Rehbein, and Lamia Tounsi. 2010. Statistical parsing of morphologically rich languages (SPMRL): What, how and whither. In *Proceedings of the NAACL HLT 2010 First Workshop on Statistical Parsing of Morphologically-Rich Languages*, SPMRL '10, pages 1–12, Los Angeles, California.

Andreas van Cranenburgh, Remko Scha, and Rens Bod. 2016. Data-oriented parsing with discontinuous constituents and function tags. *Journal of Language Modelling*, 4(1):57–111.

Yannick Versley and Yana Panchenko. 2012. Not just bigger: Towards better- quality web corpora. In *Seventh Web as Corpus Workshop (WAC7)*, pages 44–52, Lyon, France.

Yannick Versley. 2014. Incorporating semi-supervised features into discontinuous easy-first constituent parsing. In *In First Joined Workshop of Statistical Parsing of Morphologically Rich Language and Syntactic Analysis of Non-Canonical Languages*, Dublin, Ireland.

Creating and designing a corpus of rural Spanish

Carlota de Benito Moreno
Universität Zürich
Romanisches Seminar
CH-8032 Zürich, Switzerland
carlota.debenitomoreno@uzh.ch

Javier Pueyo
College of the Holy Cross
Worcester
01610 Masssachusets, USA
javier.pueyo@gmail.com

Inés Fernández-Ordóñez
Universidad Autónoma de Madrid / Real Academia Española
Departamente de Filología Española
28049 Madrid, Spain
ines.fernandez-ordonnez@uam.es

Abstract

In this paper we address some of the difficulties that arise when compiling a corpus of rural varieties (namely, the COSER corpus of Rural Spanish). These difficulties affect mainly two different aspects of the corpus-building process, i.e., the transcription process, especially regarding the conventions used, and the lemmatization process. We describe the main problems that affected the COSER corpus during these two processes and the solutions that were adopted.

1 Introduction

In this paper we aim to describe the processes of transcription and lemmatization of the COSER corpus, which documents rural varieties of spoken Peninsular Spanish; the difficulties associated to these two processes, and how they were addressed. In section 2 a brief description of the corpus, its compilation process and its main purpose is provided. Section 3 focuses on the transcription process, especially on the transcription rules that were designed specifically for the representation of rural Spanish within this corpus. Section 4 presents the lemmatization process, which had to be adapted to the specific transcription conventions used. Finally, some conclusions are summarized in section 5.

2 The COSER corpus

COSER (an acronym that stands for *Corpus Oral y Sonoro del Español Rural —Audible Corpus of Spoken Rural Spanish* in English) was designed by Inés Fernández-Ordóñez with the goal of documenting rural varieties of Peninsular Spanish in a format that enabled the morphosyntactic study of these varieties. COSER consists of spoken interviews to old rural non-mobile speakers of different villages in Spain that have been recorded *in situ* (that is, not in a lab setting). In its current composition, the mean of the duration of the interviews is 75 minutes – interviews must be lengthy in order to document sufficient instances of different morphosyntactic structures (Fernández-Ordóñez 2009, 2010a & b).

The interviews have been being recorded from 1990 on (and they are still ongoing). So far 1124 villages of 44 different provinces have been interviewed, which amount to 1434 hours of audio and 2248 recorded speakers. Currently, 147 interviews from 141 villages (ca. 184 hours) have been transcribed and are available online (see http://corpusrural.es/) – these amount to 2,727,967 tokens and 1,853,141 words, which comprise 106,505 conversation turns.

So far, the transcription process has been carried out manually by a number of collaborators in the project. Manual transcriptions are highly costly in both economic and time terms, but they also have advantages, especially when dealing with substandard speech, where a human transcriber is more likely to understand and transcribe correctly difficult fragments.[1] As will be seen in section 3 (cf. especially subsection 3.1), the fact that transcriptions are done manually has had a strong impact in the transcription rules.

[1] Now that a significant proportion of the interviews have been manually transcribed, collaboration with private partners to automatically transcribe the rest of the corpus is being sought. So far, we have established contact with Verbio, a firm that specializes in natural language processing.

The available transcriptions are currently being lemmatized automatically using FreeLing, a process that will be explained in detail in section 4.

3 Transcription rules

One of the main decisions that has to be made when compiling a corpus of substandard speech is which phenomena should be included in the transcriptions and which can be left out (for the impossibility of including every possibly relevant detail of the original data in a corpus edition, cf. López Serena 2006). The main guidelines for such a decision must be the purposes of the corpus, but their secondary uses can also be taken into account.

As said above, the main purpose of COSER is to provide a database for research on dialectal morphosyntax of Peninsular Spanish, which advises against providing a phonetic transcription of the interviews. However, some salient phonetic substandard phenomena can interact with morphosyntactic phenomena, which in turn suggests that phonetic phenomena should be included in the transcription.

Hence, COSER adopts an intermediate solution, using "regular" spelling (as opposed to phonetic alphabets) to reflect some phonological (but not phonetic) substandard phenomena. The two main phonological changes included in the transcription are the omission and the addition of phonological segments. For instance, the dialectal pronunciation of *mucho* [ʻmut͡ʃo] 'much' as [ʻmunt͡ʃo] is transcribed *muncho*, adding the extra <n> that reflects the substandard extra [n], and the colloquial pronunciation of *comprado* [kom'praðo] 'bought' as [kom'prao] is transcribed *comprao*, supressing the <d> also in the spelling. The suppression of phonological segments due to the concatenation of sounds within the sentence is marked by a single quote (ʻ): the fast pronunciation of *que has* 'that you have' /ke as/ as /kas/ is hence transcribed as *qu'has*.

While these examples are only phonetic, the application of these rules allows for including phenomena whose precise nature (whether phonetic or morphological) is unclear or debated. This is the case of the dialectal pronunciation of the modal adverb *así* 'so' as *asín*, which can be due both to phonetic and to morphological reasons (cf. Rodríguez Molina 2015); the addition of a final -n to the combination of some verbal forms with the reflexive clitic (*sentarse* 'sit down' > *sentarsen*), which has been interpreted both as a phonetic process and as the addition of a plural morpheme (cf. Heap / Pato 2012) or the common reduction of the universal quantifier *todo* 'all' to *to*, which has important morphosyntactic consequences, such as the loss of gender agreement (cf. Fernández-Ordóñez 2015).

Similarly, changes in the stress position are another phonological process represented in COSER transcriptions. That is, substandard pronunciations such as the pronunciation of proparoxytone words as paroxytone, typical of Aragonese varieties, are transcribed by using an extra accent — that may or may not be in accordance with the standard accentuation rules. For instance, the pronunciation of *pájaro* [ʻpaxaro] as [pa'xaro] is transcribed as *pajáro* (despite the fact that standard spelling would dictate the spelling *pajaro* for such a form). The reason for this "extra" marking is to indicate that there was an actual change in the stress position and that the lack of the accent is not a typo (as most lemmatizer softwares would most likely assume). Once again, while this transcription rule mostly affects phonological phenomena, changes in the stress position may also have morphological consequences, as with verbal forms – the change from [kan'taramos] (*cantáramos*) to [kanta'ramos] (*cantarámos*) alters the morphological relationships within the verbal paradigm.

This systematic representation of phonological phenomena sets COSER apart from similar projects in Spanish, such as PRESEEA, and other languages, such as CORDIAL-SIN for Portuguese, FRED for English or the Nordic Dialect Corpus for Scandinavian languages, which rely only on standard orthography except for those forms that are relevant to morphosyntactic analysis (CORDIAL-SIN transcription conventions: 8), but "do not offer any consistent renderings of phonological features" (FRED user's guide: 10). While it requires substantially more work, the advantage of the approach adopted by COSER is that it does not make any assumptions on what phenomena are relevant for morphosyntactic analysis, hence allowing for the potential discovery of morphosyntacic phenomena that have not yet been described. As secondary effects, these transcription rules make COSER useful also for those who are interested in phonological variation in Peninsular Spanish and provide a more accurate image of the speech of the informants.

A second aspect that had to be dealt with for designing COSER transcription rules is not related to its purpose, but to its material. Transcribing spoken interviews requires some circumstances of the conversation to be taken into account, es-

pecially those that refer to turn-taking, interruptions and self-corrections.

Conversational turns are normally not distributed orderly within the participants of a conversation, but they typically imply the overlapping of at least two speakers during a few seconds. Reflecting properly such overlaps in the transcription is crucial to its alignment with the audio files. In corpora designed for the study the characteristics of spoken language, overlaps during turn-taking are typically transcribed with indented lines, as in the following example taken from conversation 146a of Val.Es.Co:

6	B: pues lo tenemos que celebrar→	[¿eeh?]
7	A:	[claaro]

Figure 1. Indented overlaps in Val.Es.Co.

This convention, however, makes the transcription hard to read – an undesirable effect for a corpus whose main purpose is the documentation of morphosyntactic variation. COSER transcription rules, then, try to avoid this problem by using written tags to indicate that a specific fragment was produced simultaneously to some other fragment and who produced it. The simultaneous fragment does not appear in a new line or paragraph, but is instead inserted within the speech of the first speaker, in the exact moment where overlapping occurs.[2] Example (1), for instance, depicts the overlap of the interviewer (E) with the informant (I1). Different colours are used to increase legibility:

(1) **I1:** Es que es la cueva los Moros… Había una farmacia donde estaba todos los… que dejaba el botiquín. Mira todo esto eran todo casas, en cada de esta hay con dos ventanitas era **[HS:E** Sí…, sí.**]** una… Ahora vive ahí un señor soltero, después aquí las tienen aquí… (COSER 2501, Ausejo, La Rioja)

This representation was chosen both due to readability reasons and because it easily allows for differentiating the speech of various participants in the lemmatization process. An acknowledged shortcoming, however, is the fact that it does not specify the end of the overlap, which is made up for by the fact that the audio files are provided together with the transcription.[3]

Two other fundamental spoken phenomena that must be marked when transcribing interviews for linguistic purposes are interruptions and self-corrections. Not only is the proper representation of these phenomena key for the accuracy of the transcription, but they also have linguistic significance – interruptions can be used as discourse-planning tools (López Serena 2007) and self-corrections can be indicators of sociolinguistic awareness. In COSER, the hyphen (-) indicates an interrupted word (see example (2)) and the vertical bar (|) indicates an interruption followed by a sequence that does not repeat the interrupted sequence, i.e., the first sequence has been altered or corrected (see example (3)). The use of these transcription conventions, then, makes the COSER corpus a useful tool for discourse researchers too.

(2) y lo echas a una especie de banco, entonces **le cla-, le clavan** el cuchillo y sacan la sangre. (COSER 4128-2, Perales de Alfambra, Teruel)

(3) Hace dos años… me parece, no sé si son dos o tres, teníamos **una ce- | una cosecha** que era la, la, la mayor. (COSER 4128-2, Perales de Alfambra, Teruel)

3.1 The disambiguation convention

A special transcription rule is the so-called disambiguation convention, which was especially designed for easing the difficulties that COSER's substandard orthography could cause in the lemmatization process. The phonological processes included in the transcritpion (i.e. omissions of phonological segments and changes in the stress position) contribute to the proliferation of ambiguous forms in the final text and hence pose a potential problem to the lemmatization process. This proliferation of ambiguous forms is especially troubling insofar it affects substandard forms, i.e., the potentially most interesting forms of the corpus.

For instance, the loss of intervocalic /d/ and final /r/, extremely common in Southern varieties, produce the identical pronunciation of the infinitive and the participial adjective feminine of verbs in the 1st conjugation: *cantar* /kan'tar/

[2] Overlaps of more than two speakers are represented in the same way, with two consecutive "overlap tags" inserted within the speech of the primary speaker.

[3] Audio-text alignment is not yet provided in COSER, although it is planned for future stages.

'to sing' becomes *cantá* /kan'ta/, as does *cantada* /kan'tada/ 'sung.FEM'. Similarly, substandard pronunciation of the locative adverb *adonde* /a'donde/ 'where' renders the spelling *ande* /'ande/, a form identical to the 1st person singular of the present subjunctive of *andar* 'to walk'.

Since stress has distinctive value in Spanish, changes in the stress position can also result in ambiguities for the lemmatization tool. For example, a paroxytone pronunciation of *cántara* /'kantara/ 'jug', expected in Aragonese varieties, would become *cantára* /'kantara/, phonetically identical to the 1st and 3rd person singular of the subjunctive imperfect of *cantar*. Although these two forms are not spelled identically (the spelling of the verbal form would be *cantara*, with no accent, see above), *cantára* does not represent an unequivocal dialectal pronunciation of *cántara*, since a third possibility exists: *cantará* /kanta'ra/ is the 3rd person singular of the indicative future of *cantar* and a hypothetical change of stress could be also represented as *cantára*.

Lemmatization tools normally have disambiguation resources, but since these ambiguous forms are only ambiguous because of the special COSER transcription rules, a disambiguation convention was designed in order to help the lemmatization tool with these examples. That is to say, transcribers manually indicate whether a dialectal form is ambiguous and which is the standard reading of the form. This disambiguation convention is quite intuitive and uses the equality sign to identify the standard form, placing both between parentheses. The substandard form is placed at the left of the equality sign, while the standard form is placed at the right. That is to say, /kan'ta/ can be transcribed *(cantá=cantar)* or *(cantá=cantada)*, the adverb /'ande/ is transcribed *(ande=adonde)*, and the substandard pronunciation of the noun *cántara* is transcribed *(cantára=cántara)*, as opposed to a hypothetical *(cantára=cantará)*. The second word in the parenthesis is used by the lemmatization software to assign a tag to the first word, which in turn is the one maintained in the transcription (as available to the public).

4 The linguistic annotation process

The transcription system outlined above normalizes in some degree the language recorded in the interviews. However, as said above, it still preserves many of the phonetic, morphological, lexical, and even syntactic features of oral and rural Spanish, which prevents COSER from being fully lemmatized and PoS annotated with Natural Language Processing (NLP) tools developed for standard written Spanish. Typically, tools for the analysis and annotation of modern languages are trained on and applied to orthographically standardized varieties of such languages. Therefore, lemmatization and PoS annotation of the rural and conversational Spanish encoded in our particular transcription system is still a challenging process for any standard NLP tool.

In order to linguistically annotate our corpus, we decided to extend an existing tool, FreeLing, which is an open-source NLP system, developed at the Universitat Politècnica de Catalunya (Padró, 2011). FreeLing is both a state-of-the art NLP library and a set of linguistic resources with multilingual capabilities that is used for the linguistic processing of standardized modern languages as English, Spanish, Catalan or Russian, among others. Being open-source, it is possible to freely modify its computational code and create new lexical resources, linguistic rules, and statistical information for the analysis of languages originally excluded. More interestingly, it is also relatively easy to extend and adapt the code and the linguistic resources provided by FreeLing in order to analyze non-standard varieties of a language already included, such as standard Spanish in our case.

In order to adapt and extend FreeLing to analyze oral and rural Spanish and to deal with our particular transcription system, we had to modify some key modules that were primarily designed to process standard Spanish written sources:

4.1 Tokenizer

As explained above, our transcriptions include a sheer number of conversational, and linguistic marks, which FreeLing is not able to understand out-of-the-box. In order to preserve the conversational structure and information included within the transcriptions, we pre-processed the transcriptions and converted those marks to XML tags and attributes. For example, indications of simultaneous speech such as [HS:E Sí…, sí.] (see example (1) above) were converted to <HS speaker="E"> Sí…, sí. </HS>. We then modified FreeLing's tokenizer module to include rules that preserve XML tags without splitting them. Additionally, we extended FreeLing's tagging mod-

ule, in order for the tagger to assign customized labels to each of the XML tags in the corpus.[4]

Our transcription conventions also required to modify FreeLing's tokenizer rules to deal with the punctuation marks used to transcribe interruptions and self-corrections (-, …, and |, see section 3), and also to allow the program to recognize lexical blends and contractions containing single quotation marks (*qu'has* for *que has*, *pa'l* for *para el*, etc., see section 3), an orthographical practice unknown to modern Spanish. The tokenized transcriptions contained 8,684 interrupted words marked by a hyphen, and 14,768 self-corrections marked by a vertical bar.

4.2 Lexical dictionary

The first task we needed to address in order to use FreeLing's standard Spanish analyzer was to extend its some 600,000 words/lemma/PoS dictionary with new entries reflecting the vocabulary of the semantic fields related to the rural life. We developed tools to identify all the terms in our corpus that were not included in FreeLing's Spanish lexical resources, and manually confirmed or modified the lemma and PoS tag initially proposed by the program. More than 3,000 words/lemmas/PoS were added to the massive Spanish dictionary shipped with FreeLing.

As explained above, we had marked potentially ambiguous non-standard realizations of common Spanish words by means of equal signs, mapping the non-standard occurrences of a given word to its corresponding normalized form. For example, the adverb *muy* 'very' is frequently shortened to *mu* in oral speech, which is reflected in our transcription system as *mu(0=y)*. All these cases – which amount to 12,750 items – were extracted from the transcriptions and were automatically duplicated as new entries in FreeLing's Spanish dictionary, so that the non-standard form was associated with the lemma and PoS tag of its standard counterpart: *mu(0=y): mu muy RG (< muy muy RG)*. We were also able to automatically duplicate entries and analyses of words with alternating stress patterns since they receive special marking in our transcriptions (see section 3) and, thus, were easily recognized and mapped to standard entries in FreeLing's dictionary.

[4] A total of 296,218 XML marks were obtained in the preprocessing of the 147 available interviews – 24, 309 of which correspond to overlapping fragments.

4.3 Affixation rules

Some of the dialectal varieties of Spanish recorded in the COSER corpus use derivative suffixes and verbal morphological endings that somehow differ of those of standard Spanish. We have extended FreeLing's affixation rules, so that those suffixes and verbal endings are properly recognized and the adequate lemmas and PoS tags are correctly assigned by the tagger. For example, the diminutive suffixes *-ico/a* (in Aragonese Spanish and western dialects) or *-ín/-ina* (in Asturian Spanish and eastern dialects) are much more frequent than the standard ending *-ito/a*. We introduced rules to detect these non-standard derivative suffixes, extract the root from the form, and re-analyze it (for example, *jugos-inos, grande-cico*, etc. for standard *jugos-itos, grande-citos*, etc. are now correctly analyzed as diminutive forms of the lemmas *jugoso* 'juicy' or *grande* 'big').

Furthermore, we also had to extend FreeLing's rules of clitic pronoun annotation since in some varieties of Spanish, both the form of the pronouns (*mos, sos, tos, vos* for standard *nos* 'us' and *os* 'you.OBJ'), and their position differ from standard Spanish. For example, postponed-clitic constructions like *trájo-me-lo* (lit. 'he.brought it to.me') instead of the standard Spanish syntax *me lo trajo* (lit. 'to.me it he.brought') are frequent in the Asturian variety of Spanish.

Adapting an existing tool as FreeLing and its standard Spanish linguistic resources, both to our transcription system and to the oral and rural sources of the COSER has allowed us to fully lemmatize and annotate more than 180 hours of transcribed interviews. Furthermore, having been able to integrate this modified version of the tool into our own programs and workflow will allow our research team to keep updating FreeLing's linguistic resources for the COSER as the process of transcribing more interviews continues.

5 Conclusion

The substandard varieties documented in COSER pose a number of challenges to the adequate transcription and processing of the materials of the corpus. In this paper we have described how we have dealt with such challenges, both at the transcription (where we have resorted to a number of ad hoc conventions) and the lemmatization (where we have adapted previously available tools to such conventions) levels.

References

Araceli López Serena. 2006. La edición como construcción del objeto de estudio. El ejemplo de los corpus orales. In L. Pons Rodríguez (ed.), *Edición y crítica textual*, Madrid / Frankfurt, Iberoamericana / Vervuert, 301-334.

Araceli López Serena. 2007. *Oralidad y Escrituralidad en la Recreación Literaria del Español Coloquial*. Gredos, Madrid.

CORDIAL-SIN = Ana Maria Martins (coord.). 2000-2010. *CORDIAL-SIN: Corpus Dialectal para o Estudo da Sintaxe / Syntax-oriented Corpus of Portuguese Dialects*. Lisboa, Centro de Linguística da Universidade de Lisboa. http://www.clul.ul.pt/en/resources/411-cordial-corpus Transcription conventions available at: http://www.clul.ul.pt/english/sectores/variacao/cordialsin/manual_normas.pdf

COSER = Inés Fernández-Ordóñez. 2005-. *Corpus Oral y Sonoro del Español Rural*. http://corpusrural.es/

David Heap and Enrique Pato. 2012. Plurales anómalos en los dialectos y en la historia del español. In E. Montero Cartelle and C. Manzano Rovira (eds.), *Actas del VIII Congreso Internacional de Historia de la Lengua Española*. AHLE/Meubook, Santiago de Compostela, vol. 1, 829-840.

Inés Fernández-Ordóñez. 2009. Dialect grammar of Spanish from the perspective of the Audible Corpus of Spoken Rural Spanish (or Corpus Oral y Sonoro del Español Rural, COSER). *Dialectologia*, 3, 23-51.

Inés Fernández-Ordóñez, Inés. 2010a. La Grammaire dialectale de l'espagnol à travers le Corpus oral et sonore de l'espagnol rural (COSER, *Corpus Oral y Sonoro del Español Rural*). *Corpus: "La syntaxe de corpus / Corpus syntax*, 9, 81-114.

Inés Fernández-Ordóñez, Inés. 2010b. New methods for the study of grammatical variation and the Audible Corpus of Spoken Rural Spanish. In Gotzon Aurrekoetxea & José Luis Ormaetxea (eds.), *Tools for Linguistic Variation*, Bilbao, Universidad del País Vasco, 119-30.

Inés Fernández-Ordóñez. 2015. *Mucha trabajo*: sincretismo femenino en los cuantificadores evaluativos de Cantabria. In S. García *et al*., *Studium Grammaticae. Homenaje al profesor José Antonio Martínez*, EdiUNo, Oviedo, 337-349.

FRED = Bernd Kortmann et al. 2000-2005. *Freiburg English Dialect Corpus*. http://www2.anglistik.uni-freiburg.de/institut/lskortmann/FRED/ User's guide available at: https://www.freidok.uni-freiburg.de/fedora/objects/freidok:2489/datastreams/FILE1/content

The Nordic Dialect Corpus = Janne Bondi Johannessen, Joel Priestley, Kristin Hagen, Tor Anders Åfarli, and Øystein Alexander Vangsnes. 2009. *The Nordic Dialect Corpus*. http://www.tekstlab.uio.no/scandiasyn/

Javier Rodríguez Molina. 2015. El adverbio *así* en español medieval: variantes morfofonéticas. In J. M. García (dir.), *Actas del IX Congreso Internacional de Historia de la Lengua Española*. Arco/Libros, Madrid, 1049-1064.

Lluís Padró. 2011. Analizadores Multilingües en FreeLing. *Linguamatica*, 3(2):13-20.

PRESEEA = Francisco Moreno Fernández (coord.). 2014–. *Proyecto para el Estudio Sociolingüístico del Español del España y de América*. Alcalá de Henares: Universidad de Alcalá. http://preseea.linguas.net/Inicio.aspx

Val.Es.Co = Pons Bordería, Salvador et al., *Corpus anotado de español coloquial*, available at http://www.uv.es/corpusvalesco/index.html.

Paragraph Vector for Data Selection in Statistical Machine Translation

Mirela-Stefania Duma and Wolfgang Menzel
University of Hamburg
Natural Language Systems Division
{mduma, menzel}@informatik.uni-hamburg.de

Abstract

In this paper, we investigate data selection methods used in domain adaptation for Statistical Machine Translation targeting an in-domain made up of non-standard data, such as transcriptions of spoken data. In data selection, the sentences from the general domain are scored according to their similarity to the in-domain. This research explores Paragraph Vectors as means of scoring sentences from the general domain. The experimental evaluation results show that our method improves the translation quality over the baselines, as well as over a state-of-the-art data selection method.

1 Introduction

Data selection is a widely used method for performing domain adaptation for Machine Translation (MT). Given a large pool of general domain data and a smaller-sized in-domain data, the task is to filter the sentences from the general domain with respect to their similarity to the in-domain. After scoring the general domain sentences using a similarity metric, a ratio of the general domain is kept and used for SMT. The underlying assumption is that the general domain is big enough to contain sentences similar to the in-domain. The challenges in data selection consist of choosing a metric or a method that evaluates how similar is a sentence from the general domain to the in-domain and after scoring all sentences, determining what is the ratio of general domain sentences to be kept.

As general domain data we chose the Commoncrawl corpus[1] as it is a relatively large corpus and contains crawled data from a variety of domains as well as texts having different discourse types (including spoken discourse). The in-domain consisted of the TED Talks corpora used in the IWSLT 2016 MT Evaluation Campaign[2]. The difficulties in translating TED stems from the small size of the corpus and from the unconventionality of the corpus which is a concatenation of transcribed talks having different topics. The domain adaptation problem is not only a problem of adapting to a domain, but also to spoken discourse style.

In Le and Mikolov (2014) sentences are represented as continuous vectors with empirical results that show that Paragraph Vectors outperform the traditional bag-of-words approach of representing text. It was succesffuly applied in opinion mining and information retrieval tasks (Le and Mikolov, 2014).

In this paper, we aim to determine whether using Paragraph Vectors in the scoring phase is helpful in capturing the degree of similarity of general domain sentences to TED talks. The idea was first introduced in Duma and Menzel (2016) for the task of domain adaptation to the IT domain as part of the First Conference on Machine Translation (WMT 2016). The encouraging results using Paragraph Vectors constitute the basis of our work. We aim to introduce a new scoring formula that considers sentence length and to verify whether using Paragraph Vector is also useful in the setting of translating TED talks.

We trained SMT systems on the English-German language pair and used the BLEU (Papineni et al., 2002), NIST (Doddington, 2002) and METEOR (Denkowski and Lavie, 2014) metrics in assessing the performance of the systems.

We first shortly summarize related work in data selection for SMT in Section 2, then describe Paragraph Vector in Section 3. The next section presents the experimental settings for training the

[1] http://commoncrawl.org/

[2] https://sites.google.com/site/iwsltevaluation2016/mt-track

SMT systems along with the algorithm we used in performing data selection. Lastly, Section 5 contains an overview of the systems evaluation.

2 Related work

Three approaches are commonly used in data selection: information retrieval inspired (Hildebrand et al., 2005; Lü et al., 2007; Tamchyna et al., 2012), perplexity-based (Mandal et al., 2008; Axelrod et. al, 2011; Mansour et al., 2011) and edit distance similarity inspired (Wang et al., 2013).

The state-of-the-art data selection method we chose to use for comparison with our method is perplexity-based and presented in Axelrod et. al (2011). Four language models are trained for the in-domain and the general domain source and target sides of the corpora. Given a sentence pair from the general domain, the method scores it by summing up the cross-entropy difference scores from each side of the corpus. Axelrod et. al (2015) applied this method for general domains including the Commoncrawl corpus and for the in-domain TED Talks. We name this metric PPL in the rest of the paper.

In this paper, we propose a new scoring formula for determining the similarity of a general domain sentence to the in-domain using Paragraph Vectors (Le and Mikolov, 2014) for representing the sentences as continuous vectors. The direction of using Paragraph Vectors in data selection for SMT was introduced in Duma and Menzel (2016) where the semantic similarity of sentences was successfully employed in the task of domain adaptation of MT to the IT domain as part of WMT 2016. We extend that work by introducing a new scoring formula that combines the similarity scores produced by Paragraph Vectors and we further improve the final score of a general domain sentence by using a sentence length penalty.

3 Paragraph vector

The traditional representation of text consists in the bag-of-words model which has the disadvantage of not considering the semantics of words. In order to overcome this weakness, Le and Mikolov (2014) introduce Paragraph Vectors. Similar to word vectors (Mikolov et al., 2013), Paragraph Vectors give a continuous distributed vector representation of the input. Word vectors capture the semantics of words by looking at their representations in the vector space: similar words have vectors that are closer to each other compared to non-similar words. For example, the words ”strong” and ”powerful” have their word vectors close to each other indicating their semantic similarity. Moreover, algebraic operations can be applied on the word vectors where, for example, vector(”King”) - vector(”Man”) + vector(”Woman”) gives as result a vector that is close to the vector representation of the word ”Queen” (Mikolov et al., 2013).

Going one step further than the word vectors, aiming at representing a text of variable length (phrases, sentences, documents), Paragraph Vector uses the word vectors in computing the final vector representation of the text. Since we use bilingual corpora where the basic unit is a sentence, we chose to represent sentences as vectors. The paragraph vector is concatenated with several word vectors from the sentence and used in predicting the following word given the context. The contexts have a fixed length and are sampled from a sliding window over the paragraph. The paragraph vector acts like a memory that remembers the topic of the sentence or what is missing from the current context. The word vectors and the paragraph vectors are trained using the stochastic gradient descent and back-propagation (Rumelhart et al., 1986). While paragraph vectors are unique among sentences, the word vectors are shared (Le and Mikolov, 2014).

We use single sentences as paragraphs. The reason why we adopted Paragraph Vector is because similarly to word vectors, they reflect semantic relatedness. Moreover, we have chosen Paragraph Vectors for representing sentences as vectors because the approach does not require parsing or available labeled data. The implementation of Paragraph Vectors we used is Doc2vec from the *gensim* toolkit[3].

4 Experimental Framework

For tuning the MT systems we made use of the IWSLT16.TED.dev2010 dataset, also provided by IWSLT. For evaluating the systems the IWSLT16.TED.tst2014 dataset was used.

All systems have been developed with the widely used Moses phrase-based MT toolkit (Koehn et al., 2007) and the Experiment Management System (Koehn, 2010) that facilitates the

[3] https://radimrehurek.com/gensim/models/doc2vec.html

preparation of scripts for experiments.

4.1 Data preprocessing

The data was tokenized, cleaned and lowercased using the scripts from EMS. Furthermore, the general domain data was filtered by removing the sentence pairs that do not pertain to the English-German language pair according to the jlangdetect library[4].

Sentences that contain non-alpha characters were removed from both corpora and punctuation was normalized. Table 1 presents some data statistics for both domains after preprocessing:

Corpora	Sentences	Tokens	
		English	German
Commoncrawl	2.34M	50.33M	46.11M
TED	196K	3.49M	3.07M

Table 1: Corpora statistics after preprocessing

4.2 Experimental settings

Word alignment was performed using GIZA++ (Och and Ney, 2003) with the default *grow-diag-final-and* alignment symmetrization method. The target side of the Commoncrawl and TED corpora was utilised in estimating 5-gram language models (LM) using the SRILM toolkit (Stolcke, 2002) with Kneser-Ney discounting (Kneser and Ney, 1995). For most of the experiments we used LM interpolation where the in-domain LM and the general domain LM were interpolated using weights tuned to minimize the perplexity on the tuning set. The same data was used for tuning the systems with MERT (Och, 2003).

4.3 Baseline systems

Two baselines were trained using the concatenation of the in-domain data and the general domain data: BS_{simple} used an LM estimated from the concatenation of the data, while the stronger baseline BS_{strong} used LM interpolation.

4.4 Data selection using Paragraph Vector

In this section the algorithm for data selection is described (Figure 1). We name our doc2vec method $SEFp$ (Sentence Embedding Filtering with penalty). The filtering procedure is similar to the one presented in (Duma and Menzel, 2016). It receives as input the bilingual in-domain corpus $\mathcal{I}n$, the bilingual general domain $\mathcal{G}en$, $\mathcal{N}$ as the number of most similar sentences that should be retrieved given a threshold δ that is used in filtering the corpus and $\mathcal{P}$, the percentage of sentences to be selected from the general domain. To train the paragraph vectors we concatenated $\mathcal{I}n$ and $\mathcal{G}en$ resulting in data set $\mathcal{C}$. The steps needed for training the doc2vec model required tagging every sentence from the source side of the concatenated corpus $\mathcal{C}_{source}$ with its corresponding line number in the corpus and building a vocabulary from the tagged $\mathcal{C}$. Therefore, a sentence that came from $\mathcal{I}n$ was tagged with a number from $[1, size_{In}]$ and a sentence that came from $\mathcal{G}en$ was tagged with a number from $[size_{In} + 1, size_{In} + size_{Gen}]$. The doc2vec model $\mathcal{M}$ was trained on the tagged C_{source}.

Given a sentence pair $(s_i, t_i) \in \mathcal{G}en$, the top $\mathcal{N}$ most similar vectors to s_i are computed and retrieved in the form of a pair $(index, score)$ where $index$ gives the tag (i.e. the line number) of the selected similar sentence to s_i and $score$ specifies the similarity between s_i and s_{index}. The similarity is computed as the cosine between the two vectors.

The next step is computing the sentence score. The retrieved top $\mathcal{N}$ scores include similarities with sentences either from $\mathcal{I}n$ or $\mathcal{G}en$. Given a sentence $s_i \in \mathcal{G}en$, only the similarity scores between s_i and sentences from $\mathcal{I}n$ ($index_j < size_{In}$) contribute to building the final sentence score. These scores are filtered using the threshold δ set to 0.5. We plan to further experiment with other values of δ in future work. Since the degree of similarity matters, we favor similarity scores with $\mathcal{I}n$ sentences that are higher over similarity scores that are lower. This preference has been implemented by means of the position index j in the ranked list of selected sentences R_i.

The final score of the general domain sentence is built by accumulating all the intermediary scores. We observed that some sentences from Commoncrawl are very long leading to a very high score. We introduced a sentence length penalty with the purpose of giving a penalty to long sentences by dividing the final score to $size_{s_i}$, the number of words for s_i.

In comparison to the work in (Duma and Menzel, 2016), here we introduce a new scoring formula with the aim of investigating new possibilities of combining similarity scores produced by

[4] https://github.com/melix/jlangdetect

Algorithm 1 Doc2vec Filtering with penalty

procedure FILTER($\mathcal{I}n, \mathcal{G}en, \mathcal{N}, \delta, \mathcal{P}$)
  $\mathcal{C} \leftarrow \mathcal{I}n + \mathcal{G}en$
  for each sentence $s_i \in \mathcal{C}_{source}$ **do**
    tag s_i with the line number i
  build vocabulary from tagged $\mathcal{C}_{source}$
  train doc2vec model $\mathcal{M}$ using tagged $\mathcal{C}_{source}$
  for each sentence pair $(s_i, t_i) \in \mathcal{G}en$ **do**
    $\mathcal{R}_i = top(\mathcal{N}, mostSimilar(\mathcal{M}, s_i))$
    $Sim_{s_i} = \{(index, score) \in \mathcal{R}_i |\ index \in [1, size_C], score \in (0, 1)\}$
    for $(index_j, score_j) \in Sim_{s_i}$ **do**

$$score_{i,j} = \begin{cases} score_{i,j}^2 * \frac{\mathcal{N}}{j}, & \text{if } index_j < size_{In} \text{ and } score_j > \delta \\ 0, & \text{otherwise} \end{cases}$$

    $$score_i = \sum_{j=1}^{\mathcal{N}} \frac{score_{i,j}}{size_{s_i}}$$

  sort sentences $\in \mathcal{G}en$ by their score in descending order
  while $i \leq \mathcal{P}$ **do**
    add (s_i, t_i) to $FilteredCorpus_{\mathcal{P}}$

Figure 1: Doc2vec filtering algorithm

Doc2Vec. Moreover, we consider the sentence length penalty in computing the final score of a general domain sentence.

After scoring all the general domain sentences, we sorted them in descending order and filtered the general domain sentences using the percentage $\mathcal{P}$ as the ratio of Commoncrawl sentences to be kept. We increased the ratio with 10% at every SMT model training.

The final step consisted in training several SMT systems on a concatenation of the reduced general domain corpus $FilteredCorpus_{\mathcal{P}}$ and the in-domain data $\mathcal{I}n$ and using LM interpolation of an LM estimated using $\mathcal{I}n$ and an LM estimated using the full $\mathcal{G}en$. The same interpolated LM was used in the PPL experiments and in the BS_{strong} baseline.

5 Evaluation and Conclusions

We evaluated the two baselines, the PPL metric and our proposed $SEFp$ metric using the BLEU, NIST and METEOR metrics, widely used in evaluating MT output.

Both data selection methods outperform the baselines as their maximum BLEU, NIST and METEOR scores are greater than the baseline scores. According to the BLEU scores, the best result is obtained by the $SEFp$ metric, when selecting 40% of the general domain data (BLEU = 20.23). It is to be noted that the best BLEU result obtained by the state-of-the-art metric is achieved when selecting 60% of the data (BLEU = 20.11), thus it requires more data compared to $SEFp$. The NIST scores indicate also that the best result is obtained using our method, when selecting 40% of the general data (NIST = 20.34). Evaluating the results using METEOR, both methods give the same maximum score of 41.84. However, our method uses 50% of the general domain data, while the PPL metric requires 80% of the general domain data to achieve the maximum score.

For future work we plan to further exploit Paragraph Vector by employing other scoring methods, evaluating the method proposed in (Duma and Menzel, 2016) on TED talks and also combining the currently presented approach with the Axelrod et al. (2011) approach. Moreover, an interesting idea for combining the bitexts (the in-domain data and the general domain selected sentences) is presented in Wang et al. (2016) where balanced concatenation with repetitions is used in order to have comparable sizes of bitexts.

To conclude, in this paper we introduced a new scoring method for data selection in SMT using

	BLEU		**NIST**		**METEOR**	
Percentage $\mathcal{P}$ of Commoncrawl	PPL	$SEFp$	PPL	$SEFp$	PPL	$SEFp$
10%	19.8	19.87	19.88	19.98	41.7	41.67
20%	19.91	19.51	20.02	19.62	41.7	41.59
30%	19.95	19.6	20.08	19.75	41.52	41.51
40%	19.96	**20.23**	20.09	**20.34**	41.63	41.77
50%	20	19.78	20.12	19.91	41.63	**41.84**
60%	**20.11**	19.89	**20.21**	20	41.73	41.61
70%	19.63	20.01	19.75	20.11	41.26	41.8
80%	20.03	19.98	20.16	20.15	**41.84**	41.78
90%	19.52	19.64	19.65	19.79	41.43	41.36
BS_strong	19.66		19.82		41.28	
BS_simple	19.79		19.89		41.11	

Table 2: Evaluation results with the BLEU, NIST and METEOR metrics

Paragraph Vector for determining the similarity of the sentences from the general domain to the in-domain. Our method outperformed the baselines and a state-of-the-art method with respect to commonly used MT evaluation metrics by achieving the highest scores using the least amount of filtered general domain data.

References

Amittai Axelrod, Xiaodong He and Jianfeng Gao. 2011. Domain Adaptation via Pseudo In-Domain Data Selection. *Proceedings of EMNLP 2011*.

Amittai Axelrod, Ahmed Elgohary, Marianna Martindale, Khánh Nguyen, Xing Niu, Yogarshi Vyas, Marine Carpuat. 2015. The UMD Machine Translation Systems at IWSLT 2015. *Proceedings of IWSLT 2015*.

Boxing Chen, Roland Kuhn and George Foster. 2013. Vector Space Model for Adaptation in Statistical Machine Translation *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, pages 1285-1293, Sofia, Bulgaria, August 4-9 2013.

Michael Denkowski and Alon Lavie. 2014. Meteor Universal: Language Specific Translation Evaluation for Any Target Language. *Proceedings of the EACL 2014 Workshop on Statistical Machine Translation*.

George Doddington. 2002. Automatic Evaluation of Machine Translation Quality Using N-gram Co-occurrence Statistics *Proceedings of the Second International Conference on Human Language Technology Research*.

Mirela-Stefania Duma and Wolfgang Menzel. 2016. Data selection for IT Texts using Paragraph Vector. *Proceedings of the First Conference on Machine Translation*, Volume 2: Shared Task Papers, pages 428–434.

Almut Silja Hildebrand, Matthias Eck, Stephan Vogel, and Alex Waibel. 2005. Adaptation of the Translation Model for Statistical Machine Translation based on Information Retrieval. *Proceedings of EAMT 2005*.

Reinhard Kneser and Hermann Ney. 1995. Improved backing-off for N-gram language modeling. *Proceedings ICASSP*, pages 181-184.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin and Evan Herbst. 2007. Moses: open source toolkit for statistical machine translation. *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*. June 25-27, 2007, Prague, Czech Republic.

Philipp Koehn. 2010. An experimental management system. *The Prague Bulletin of Mathematical Linguistics*, 94.

Quoc Le and Tomas Mikolov. 2014. Distributed Representations of Sentences and Documents. *Proceedings of the 31st International Conference on Machine Learning*, volume 32, Beijing, China. JMLR: W&CP.

Yajuan Lü, Jin Huang and Qun Liu. 2007. Improving Statistical Machine Translation Performance by Training Data Selection and Optimization. *Proocedings of EMNLP-CoNLL 2007*.

A. Mandal, D. Vergyri, W. Wang, J. Zheng, A. Stolcke, G. Tur, D. Hakkani-Tür, and N. F. Ayan. 2008.

Efficient data selection for machine translation. *Proceedings IEEE Workshop on Spoken Language Technology*.

Saab Mansour, Joern Wuebker and Hermann Ney. 2011. Combining Translation and Language Model Scoring for Domain-Specific Data Filtering. *Proceedings of IWSLT*.

Tomas Mikolov, Kai Chen, Greg Corrado and Jeffrey Dean. 2013. Efficient Estimation of Word Representations in Vector Space. *arXiv preprint arXiv:1301.3781*.

Franz Josef Och. 2003. Minimum error rate training in statistical machine translation. *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics*, pages 160-167, July 07-12, 2003, Sapporo, Japan.

Franz Josef Och and Hermann Ney. 2003. A Systematic Comparison of Various Statistical Alignment Models. *Computational Linguistics*, volume 29, number 1, pages 19-51.

Kishore Papineni, Salim Roukos, Todd Ward and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, July 07-12, 2002, Philadelphia, Pennsylvania.

Rumelhart, David E, Hinton, Geoffrey E, and Williams, Ronald J. 1986. Learning representations by back-propagating errors. *Nature*, 323(6088):533–536, 1986.

Andreas Stolcke. 2002. SRILM - an Extensible Language Modeling Toolkit. *International Conference on Spoken Language Processing*.

Aleš Tamchyna, Galuščáková Petra, Kamran Amir, Stanojević Miloš and Bojar Ondřej. 2012. Selecting Data for English-to-Czech Machine Translation. *Proceedings of the Seventh Workshop on Statistical Machine Translation*.

Longyue Wang, Derek F. Wong, Lidia S. Chao, Junwen Xing and Yi Lu. 2013. Edit Distance: A New Data Selection Criterion for Domain Adaptation in SMT. *Proceedings of Recent Advances in Natural Language Processing*.

Pidong Wang, Preslav Nakov and Hwee Tou Ng. 2016. Source Language Adaptation Approaches for Resource-Poor Machine Translation. *Computational Linguistics* Vol. 42, No. 2: 277–306.

Creating Silver Standard Annotations for a Corpus of Non-Standard Data

Kerstin Eckart **Markus Gärtner**
Universität Stuttgart
Institut für Maschinelle Sprachverarbeitung
Pfaffenwaldring 5b, D-70569 Stuttgart
{kerstin.eckart,markus.gaertner}@ims.uni-stuttgart.de

Abstract

We present our approach for annotating a large collection of non-standard multimodal data. Its automatically created silver standard annotations lack the quality of their manual counterparts but will be enriched with confidence estimations which allow an assessment of an annotation's expected correctness. For this we first aim at providing many different annotation layers with multiple concurrent annotations. The approach is exemplified on a collection of German radio interviews and their transcripts. Finally we argue for inclusion and consideration of a tool's own confidence values in annotations and research.

1 Introduction

Non-standard data is data which is not typical for a specific application or line of research. And since in text and speech processing, many tools nowadays work well on their typical data, time has come to take the next logical step towards other domains, modalities, languages, registers and time periods. This includes of course the handling of additional phenomena. Switching the domain might change the vocabulary and switching the modality might blur the basic structure of processing, e.g. when a parser which bases its analysis on the unit of a sentence is applied to spontaneous speech. Amongst others, shared tasks have fostered the development of approaches to domain adaptation (Petrov and McDonald, 2012), and the development of approaches that can be applied to several languages (Seddah et al., 2014).

For many tools, a set of high-quality annotated data is needed to train them on, or be adapted to. For German, the NoSta-D corpus (Dipper et al., 2013) provides a collection of non-standard data, including historical data, chat data, learner data, literary prose and spoken data from a map task. Since all parts have been manually annotated, the corpus can be used in training and evaluation. However, to study specific or less frequent phenomena, huge corpora might be necessary (Zarrieß et al., 2013).

Our goal is to provide a large collection of non-standard data for various research fields which includes two non-standard areas, spoken and web data. The data will be enhanced with several annotation layers, including interaction of tools from text and speech processing. Due to the size no full manual annotation is possible, therefore we opt for a silver standard approach, as exemplified in (Rebholz-Schuhmann et al., 2010). The silver standard provides annotation quality between gold standard and uncontrolled automatic annotation. For this, we combine information from multiple tools and annotation layers, include manual and automatic annotations, and argue for a visible confidence estimation along with annotations. We present the silver standard idea in Section 3, and focus on a current set of speech data, for which we introduce an "unnormalized" layer that constitutes non-standard data for both speech and text processing pipelines.

2 Data

The data set we focus on here is a collection of German radio interviews. The primary data available from the radio station consists of recordings of the interviews (.mp3) and edited transcripts (.pdf)[1].

The data set is non-static, i.e. more interviews are being added. At the time of writing the set comprises ca. 100 interviews of about 10 minutes length each, collected from broadcasts between May 2014 and July 2016.

The setting of the interviews is such that a host from the radio station interviews a guest on topics from the (at that time) current political and social discussion. The guest appears in a professional role

[1] For a few transcripts a .doc file was made available instead of a .pdf file.

(political representative, commissioner, founder of an association, managing director, etc.).

The definition of non-standard data varies with the task or line of research in which the data is applied. The NoSta-D corpus (Dipper et al., 2013) contains several different subcorpora of data that is considered non-canonical; and while Hirschmann et al. (2007) state that non-canonical cases can only be defined with respect to a canon – in their case a linguistic framework or an annotation scheme, Petrov and McDonald (2012) go further in the direction of processability by a tool. Transferring the latter to speech corpora includes e.g. data that is non-canonical due to recording settings. Additionally, what is non-standard data for one setting might be completely canonical in another.

Since our goal is to enhance data with various layers of annotations, we consider this data non-standard in various respects.

Regarding spoken data, planned or read speech recorded under laboratory conditions is clearly more canonical for processing and annotating than spontaneous conversations recorded in a noisy environment. Our data set is somewhere in between: semi-planned speech[2], recorded in the studio of a professional radio station. Despite the latter, the available audio recordings contain both speakers in the same file and while there is only little overlap, we regard the data as non-standard with respect to processing. An additional dimension for non-standard speech is the eloquence of the speaker. While the hosts are professional speakers from the radio station the guests vary along this dimension.

Regarding written data, newspaper text is adequately processable by most tools. Thus, non-standard data for these tools includes e.g. web data, historical data, and also written representation of features of orality. The transcripts which the radio station provides are however an edited version of the interview. The transcriber introduces sentence borders, corrects the syntax and even adds words where necessary to form a sentence. Thereby the transcripts are rather canonical data for text processing and neither include fillers, false starts or repairs nor do they necessarily keep the original syntax. Since it is our goal to adapt our text processing tools (in small steps) to more non-standard data, we reintroduce some of the features of orality to the transcripts, cf. Section 4, i.e. we create a closer transcription of what was actually said.

[2]Topics of the interview are probably known in advance.

3 Silver Standard Approach

The data described in Section 2 is part of an ongoing initiative to create a so called *silver standard collection* in the SFB732 [3]. It is meant to contain a large number of annotated resources that vary with regards to modality, language, domain and (non-)canonicality. Since manual annotations are not feasible for such a large data set[4], annotations need to be created automatically. For this the term "silver standard" describes a level of annotation quality between a manually created gold standard and the unchecked output of automatic processing. Sections 3.1 to 3.3 outline the annotation project and describe methods usable to ensure an adequate level of annotation quality or to provide quality indicators.

3.1 Variety of Annotations

Besides previously introduced radio interviews the silver standard collection will contain French radio conversations and a selection of already available German and English web corpus data. It covers different modalities (speech, written transcripts, textual web data), languages (German, French, English) and domains (interviews, conversations, blog/forum posts), making it an ideal source of non-standard data for many research fields.

While the goal is to provide a large number of automatically annotated resources that contain various types of non-standard phenomena, we still need a small set of manually annotated gold data for training or evaluation. For the German radio interviews we selected a subset of 20 interviews and their transcripts, totaling ~ 3 hours of audio and ~ 36.000 written tokens. They are being annotated for part-of-speech (TIGER/STTS guidelines by Brants et al. (2004) and Schiller et al. (1999) with additions by Seeker (2016)), information status (RefLex scheme by Baumann and Riester (2012)) and discourse.

The different data sets in the silver standard collection will then receive stand-off annotations created automatically by several tools (cf. Section 3.2) for multiple layers. Figure 1 shows a simplified version of the annotation workflow. It indicates where

[3]SFB: Sonderforschungsbereich (Collaborative Research Center) `http://www.uni-stuttgart.de/linguistik/sfb732/`

[4]For example the time cost for prosodic annotations of speech data according to the Tones and Break Indices (ToBI) system alone is around 100-200 times the real time (Syrdal et al., 2001) for experienced annotators.

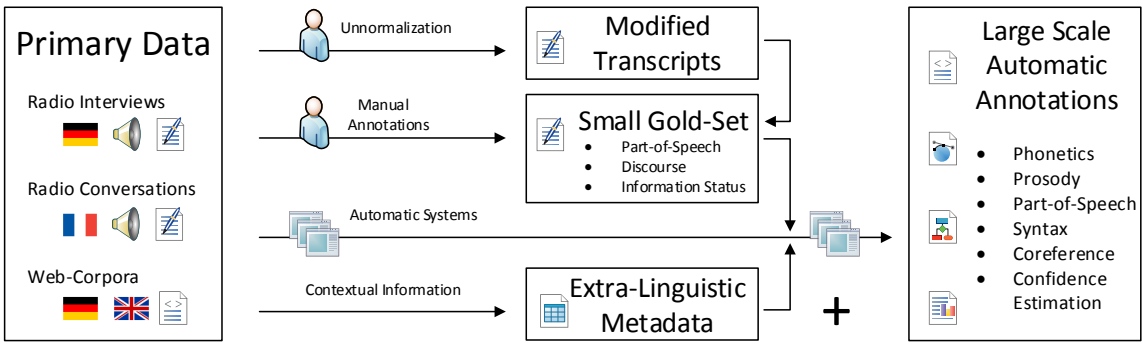

Figure 1: Composition of the silver standard collection and annotation workflow for the subset of German radio interviews (cf. Section 2).

direct human work is involved in the annotation process and which types of automatic annotations we plan to make available. Besides this vertical variety of annotation types there will also be horizontally concurrent annotations for (ideally) each type. That is, we intend to use multiple tools to create annotations for the same level. Those (potentially different) outputs can both help to get a better understanding of the data at hand or offer the basis for confidence estimations (cf. Section 3.3) or indicators for processability of data points.

We will also include extra-linguistic information[5] as additional annotations, if available. This allows to incrementally take more context into account when analyzing data. This is especially true for speech data, where Lewandowski (2013) showed the relevance of personality-related information for phonetic convergence.

Besides extra-linguistic information derived directly from the primary data, we also aim to attach the confidence or scoring values for automatically created annotations, retrieved from the respective tools. We argue that the difficulty for automatic processing presented by non-standard data makes it particularly valuable to not only look at annotations in isolation when analyzing, but also at the relative confidence with which the respective tools made those predictions. By making this information available as additional (meta-)annotation layers in our corpus it can directly be used in exploration tools such as ICARUS (Gärtner et al., 2013) for investigation together with regular linguistic features.

[5]No additional identity related data is included.

3.2 Automatic Processing

This section gives a (non-exhaustive) overview of the systems used for automatic annotations in the silver standard collection.

For processing of text resources we mainly employ pipeline systems covering multiple annotation layers, e.g.: BitPar (Schmid, 2006; Schmid, 2004), IMS-SZEGED-CIS (Björkelund et al., 2013), Mate (Bohnet and Nivre, 2012; Bohnet, 2010), IMSTrans (Björkelund and Nivre, 2015; Björkelund et al., 2016), FSPar (Schiehlen, 2003), TreeTagger (Schmid, 1994). Table 1 shows which annotation layers are covered by those systems.

In addition the IMS HotCoref DE system by Roesiger and Kuhn (2016) is used for German text to create automatic coreference annotations.

System	Syntax	Lemma	PoS	Morph.
BitPar	C		+	+
ISC	C+D		+	+
Mate	D	+	+	+
IMSTrans	D			
FSPar	D	+	+	+
TT		+	+	

Table 1: List of systems planned to be used for text processing and the annotation layers they cover (C: constituency, D: dependency, ISC: IMS-SZEGED-CIS, TT: TreeTagger).

Our pipeline for speech resources is very similar to the one applied by Schweitzer and Lewandowski (2013) for the GECO corpus. It uses IMS Festival Morphology [6] and IMS Aligner (Rapp, 1995) to

[6]http://hdl.handle.net/11022/1007-0000-0000-8E71-1

produce various annotations on the segment, syllable and word level. We further include an approximation of the F_0 contour using PaIntE (Möhler, 1998; Möhler, 2001) and on top of this categorical prosody labels (e.g. following GToBI(S) by Mayer (1995)) predicted automatically (Schweitzer, 2010; Schweitzer and Möbius, 2009).

3.3 Evaluation and Quality

To obtain meaningful confidence estimations for automatic annotations we employ different strategies. For local (i.e. within one and the same annotation layer) inconsistencies detection is facilitated using the approach developed by DECCA (Boyd et al., 2008). An implementation of their idea for part-of-speech and dependency syntax annotations with an interactive visual front-end exists in one of the plugins (Thiele et al., 2014) for ICARUS.

Taking information from multiple annotation layers into account, we can exploit various redundancies. In-level (or horizontal) redundancy constitutes for example the output of different tools for the same annotation type. It can be used to produce confidence statements based on the agreement of those tools as shown by Haselbach et al. (2012) for parser outputs. A pilot study for the web data part (George, 2016) used a token-based comparison of the output from three parsers with respect to the aspects *head*, *label*, and the combination of both (cf. also the "disagree" method from Smith and Dickinson (2014)). Confidence was derived from the number of parsers that agreed for a specific token and mapped to a respective color scheme.

Cross-level (or vertical) redundancy on the other hand exists when multiple annotation layers describe aspects that are related. If support or contradiction exists between information from different layers, we can use this to assign tentative confidence or simply mark those data points. Dickinson (2015) refers to this as making use of annotation layer inconsistencies, and gives examples for methods taking part-of-speech, syntactic and semantic information into account. With our spoken data, additional annotation layers can be taken into account, e.g. with respect to syntactic and prosodic phrase recognition.

Conventional evaluation of the tools used for automatic annotations will be performed using small gold subsets, e.g. the one mentioned in 3.1. This provides us with performance information that we can attach to entire annotation layers as metadata.

Note that all these confidence or performance values (including a tool's own confidence estimation) are not meant to be used for some a priori cleaning of the data. Instead they are treated as an annotation layer and act as possible indicators for data points which might be of interest or should be ignored for certain research questions. One can then produce excerpts of the entire data set based on the required level of confidence.

4 "Unnormalization": Including Features of Orality for Text Processing

As discussed in Section 2, an aspect of the available interview transcriptions is the omission of features of orality. While the edited transcript is suited for text processing, it is unfit for the speech processing pipeline, when trying to align text and audio data. For the interviews which are part of the gold standard, we reintroduced some of the omitted features in a way that the result is neither canonical data nor an unsolvable puzzle for one of the processing pipelines. Since a step that produces canonical forms from non-standard data is often referred to as *normalization*, we call this step *unnormalization*.

An important fact is that we consider both types of available primary data (audio and edited transcript) as equal in status. The text files are not seen as 'wrong' transcriptions or annotations, which can just be changed, but as an interesting source in its own right, e.g. for research on typing errors or aspects of edition. Thus, the original primary data is kept and the modified transcripts are created as an additional layer based on the primary data. Furthermore, the decisions made in the original transcription process are taken into account in the process of unnormalization. That is, in cases where several transcriptions are possible and the original transcription is among them, it is kept.

4.1 Process

The unnormalization is similar to processes of normalization and annotation. Guidelines have been defined and each interview is modified by two annotators independently. Adjudication is done by a third person. The guidelines comprise cases of spelling errors; missing, additional or different words; word order; repairs; repeats; and unrepaired slips of the tongue. Thereby the main principles are: (i) correct and completely heard words should be part of the modified transcript, while (ii) the transcript is changed as little as possible, such that

the decisions of the transcriber are still reflected. The results include all fully spoken words (including repetitions) in the original word order from the audio file. This is helpful for the aligner but introduces non-standard features for the text processing. On the other hand, the modified transcript does not include any fillers or words that have been uttered only partially, which would pose a vocabulary problem for the text processing, but this way the result provides still no optimal representation for the aligner. Example (1) shows a case where a word and a filler from an utterance in the audio file were not included in the transcript (2)[7]. In our process of unnormalization, the filler was also ignored, but the completely heard word *in* was added (3).

(1) obwohl die [...] in vielfach **äh** günstiger sind als
although they in several times ehm cheaper than
'although [...] they have become (in) several times cheaper than'

(2) obwohl die [...] vielfach günstiger sind als

(3) obwohl die [...] in vielfach günstiger sind als

4.2 Quantification

To give a rough quantification, that unnormalization is a first step to non-standard data for text processing, we apply a method from Faaß and Eckart (2013). They use the robust rule-based dependency parser FSPar (Schiehlen, 2003), which includes all input tokens into a dependency graph, but attaches parts which could not be properly embedded to the artificial root node. Based on the number of these attachments and the number of tokens in the sentence, Faaß and Eckart (2013) compute an *error rate* and exclude sentences from a web corpus which are considered less processable.

For our small study we parsed 10 transcripts and their manually modified counterparts with FSPar. Since FSPar comes with its own pre-processing pipeline we leave sentence border detection to this pipeline and only mark each speaker turn as its own text.[8] Table 2 shows the results of the comparison between the original and modified transcripts. The error rate increases slightly for the modified transcript. Thus, the parser encountered more tokens it could not attach properly to the dependency graph in the transcripts after the unnormalization step, i.e. the data became a bit more non-canonical for the parser. Still, we are far away from the sentences being hardly parsable at all, which is due to the official interview situation where at least one of the participants is a professional speaker from the radio station.[9]

	orig. transcript	mod. transcript
error rate	0.157	0.163

Table 2: Processability values based on FSPar.

5 Conclusions and Future Work

We presented our approach to create a large silver standard collection of non-standard data. In particular we discussed one ongoing annotation project for German radio interviews and their written transcripts. With the size of resources involved making an exhaustive manual annotation impossible we instead use existing tools to create (concurrent) annotations on various linguistic levels. To gain indicators for annotation quality we estimate confidence values for individual annotations or entire layers based on consistency checks or redundancy along horizontal and vertical annotation axes. This places the silver standard somewhere between true gold standards and raw automatic annotations in terms of quality. Data from the SFB732 silver standard collection will be made available for research purposes, along with CMDI[10] metadata and a persistent identifier for each release.

For the future we also plan to further raise awareness regarding the integration of a tool's own confidence estimation in its output. This is to motivate developers of both processing tools and data formats to consider those meta-annotations in their work, as well as to encourage their usage in research and development. We are aware of the current lack of standardization or comparability for this type of annotation, and therefore will investigate sensible ways of normalization to make confidence annotations a valuable part of NLP data.

Acknowledgments

This work was funded by the German Research Foundation (DFG) via the SFB 732, project INF.

[7] The subject is renewable energy and the larger context gives a strong indication for this reading.

[8] This decision is debatable, since a sentence might be continued by another speaker, and overlap might occur at speaker turns.

[9] Faaß and Eckart (2013) deleted sentences with an error rate above 0.7, however in a corpus with tables, etc. from raw web data.

[10] Component Metadata Infrastructure from CLARIN, https://www.clarin.eu/

References

Stefan Baumann and Arndt Riester. 2012. Referential and Lexical Givenness: semantic, prosodic and cognitive aspects. In Gorka Elordieta and Pilar Prieto, editors, *Prosody and Meaning*, number 25 in Interface Explorations. Mouton de Gruyter, Berlin.

Anders Björkelund and Joakim Nivre. 2015. Non-Deterministic Oracles for Unrestricted Non-Projective Transition-Based Dependency Parsing. In *Proceedings of the 14th International Conference on Parsing Technologies*, pages 76–86, Bilbao, Spain, July. Association for Computational Linguistics.

Anders Björkelund, Ozlem Cetinoglu, Richárd Farkas, Thomas Mueller, and Wolfgang Seeker. 2013. (re)ranking meets morphosyntax: State-of-the-art results from the SPMRL 2013 shared task. In *Proceedings of the Fourth Workshop on Statistical Parsing of Morphologically-Rich Languages*, pages 135–145, Seattle, Washington, USA, October. Association for Computational Linguistics.

Anders Björkelund, Agnieszka Faleńska, Wolfgang Seeker, and Jonas Kuhn. 2016. How to train dependency parsers with inexact search for joint sentence boundary detection and parsing of entire documents. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1924–1934, Berlin, Germany, August. Association for Computational Linguistics.

Bernd Bohnet and Joakim Nivre. 2012. A transition-based system for joint part-of-speech tagging and labeled non-projective dependency parsing. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 1455–1465, Jeju Island, Korea, July. Association for Computational Linguistics.

Bernd Bohnet. 2010. Top accuracy and fast dependency parsing is not a contradiction. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, pages 89–97, Beijing, China, August. Coling 2010 Organizing Committee.

Adriane Boyd, Markus Dickinson, and W.Detmar Meurers. 2008. On detecting errors in dependency treebanks. *Research on Language and Computation*, 6(2):113–137.

Sabine Brants, Stefanie Dipper, Peter Eisenberg, Silvia Hansen-Schirra, Esther Knig, Wolfgang Lezius, Christian Rohrer, George Smith, and Hans Uszkoreit. 2004. TIGER: Linguistic interpretation of a German corpus. *Research on Language and Computation*, 2(4):597–620.

Markus Dickinson. 2015. Detection of annotation errors in corpora. *Language and Linguistics Compass*, 9(3):119–138. LNCO-0526.R1.

Stefanie Dipper, Anke Lüdeling, and Marc Reznicek. 2013. NoSta-D: A corpus of German non-standard varieties. In Marcos Zampieri and Sascha Diwersy, editors, *Non-standard Data Sources in Corpus-based Research*, pages 69–76. Shaker.

Gertrud Faaß and Kerstin Eckart. 2013. Sdewac a corpus of parsable sentences from the web. In Iryna Gurevych, Chris Biemann, and Torsten Zesch, editors, *Language Processing and Knowledge in the Web*, volume 8105 of *Lecture Notes in Computer Science*, pages 61–68. Springer Berlin Heidelberg.

Markus Gärtner, Gregor Thiele, Wolfgang Seeker, Anders Björkelund, and Jonas Kuhn. 2013. ICARUS – an extensible graphical search tool for dependency treebanks. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 55–60, Sofia, Bulgaria, August. Association for Computational Linguistics.

Tanja George. 2016. Confidence estimation for automatic parsing of large web data sets. Masterarbeit, Institut für Maschinelle Sprachverarbeitung, Universität Stuttgart.

Boris Haselbach, Kerstin Eckart, Wolfgang Seeker, Kurt Eberle, and Ulrich Heid. 2012. Approximating theoretical linguistics classification in real data: the case of German "nach" particle verbs. In *Proceedings of COLING 2012*, pages 1113–1128, Mumbai. The COLING 2012 Organizing Committee.

Hagen Hirschmann, Seanna Doolittle, and Anke Lüdeling. 2007. Syntactic annotation of non-canonical linguistic structures. In *Proceedings of Corpus Linguistics 2007*, Birmingham.

Natalie Lewandowski. 2013. Phonetic convergence and individual differences in non-native dialogs. Abstract presented at the New Sounds Conference in Montréal.

Jörg Mayer. 1995. Transcription of German Intonation. The Stuttgart System. ms.

Gregor Möhler. 1998. Describing intonation with a parametric model. In *Proceedings of the International Conference on Spoken Language Processing*, volume 7, pages 2851–2854.

Gregor Möhler. 2001. Improvements of the PaIntE model for F_0 parametrization. Technical report, Institute of Natural Language Processing, University of Stuttgart. Draft version.

Slav Petrov and Ryan McDonald. 2012. Overview of the 2012 shared task on parsing the web. *Notes of the First Workshop on Syntactic Analysis of Non-Canonical Language (SANCL)*, 59.

Stefan Rapp. 1995. Automatic phonemic transcription and linguistic annotation from known text with Hidden Markov Models – An aligner for German. In *Proc. of ELSNET Goes East and IMACS Workshop*

"Integration of Language and Speech in Academia and Industry" (Russia).

Dietrich Rebholz-Schuhmann, Antonio Jos Jimeno-Yepes, Erik M. van Mulligen, Ning Kang, Jan Kors, David Milward, Peter Corbett, Ekaterina Buyko, Katrin Tomanek, Elena Beisswanger, and Udo Hahn. 2010. The calbc silver standard corpus for biomedical named entities a study in harmonizing the contributions from four independent named entity taggers. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Bente Maegaard, Joseph Mariani, Jan Odijk, Stelios Piperidis, Mike Rosner, and Daniel Tapias, editors, *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta. European Language Resources Association (ELRA).

Ina Roesiger and Jonas Kuhn. 2016. Ims hotcoref de: A data-driven co-reference resolver for german. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Thierry Declerck, Sara Goggi, Marko Grobelnik, Bente Maegaard, Joseph Mariani, Helene Mazo, Asuncion Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France, may. European Language Resources Association (ELRA).

Michael Schiehlen. 2003. A cascaded finite-state parser for German. In *Proceedings of EACL 2003*, pages 163–166, Budapest.

Anne Schiller, Simone Teufel, Christine Stckert, and Christine Thielen. 1999. Guidelines fr das Tagging deutscher Textcorpora mit STTS.

Helmut Schmid. 1994. Probabilistic part-of-speech tagging using decision trees. In *International Conference on New Methods in Language Processing*, pages 44–49, Manchester, UK.

Helmut Schmid. 2004. Efficient Parsing of Highly Ambiguous Context-Free Grammars with Bit Vectors. In *Proceedings of the 20th International Conference on Computational Linguistics (COLING 2004)*, Geneva, Switzerland.

Helmut Schmid. 2006. Trace prediction and recovery with unlexicalized pcfgs and slash features. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 177–184, Sydney, Australia, July. Association for Computational Linguistics.

Antje Schweitzer and Natalie Lewandowski. 2013. Convergence of articulation rate in spontaneous speech. In *Proceedings of the 14th Annual Conference of the International Speech Communication Association (Interspeech 2013, Lyon)*, pages 525–529.

Antje Schweitzer and Bernd Möbius. 2009. Experiments on automatic prosodic labeling. In *Proceedings of Interspeech 2009*, pages 2515–2518.

Antje Schweitzer. 2010. *Production and Perception of Prosodic Events – Evidence from Corpus-based Experiments*. Doctoral dissertation, Universität Stuttgart.

Djamé Seddah, Sandra Kübler, and Reut Tsarfaty. 2014. Introducing the SPMRL 2014 shared task on parsing morphologically-rich languages. In *Proceedings of the First Joint Workshop on Statistical Parsing of Morphologically Rich Languages and Syntactic Analysis of Non-Canonical Languages*, pages 103–109, Dublin, Ireland, August. Dublin City University.

Wolfgang Seeker. 2016. Guidelines for the Annotation of Syntactic Structure in the IMS Interview Corpus.

Amber Smith and Markus Dickinson. 2014. Evaluating parse error detection across varied conditions. In *Proceedings of the 13th International Workshop on Treebanks and Linguistic Theories (TLT13*, pages 230–241, Tübingen, Germany.

Ann K. Syrdal, Julia Hirschberg, Julie McGory, and Mary Beckman. 2001. Automatic tobi prediction and alignment to speed manual labeling of prosody. *Speech Commun.*, 33(1-2):135–151, January.

Gregor Thiele, Wolfgang Seeker, Markus Gärtner, Anders Björkelund, and Jonas Kuhn. 2014. A graphical interface for automatic error mining in corpora. In *Proceedings of the Demonstrations at the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 57–60, Gothenburg, Sweden, April. Association for Computational Linguistics.

Sina Zarrieß, Florian Schäfer, and Sabine Schulte im Walde. 2013. Passives of reflexives: a corpus study. Abstract at LinguisticEvidence – Berlin Special.

Diachronic Evaluation of NER Systems on Old Newspapers

Maud Ehrmann, Giovanni Colavizza, Yannick Rochat, Frédéric Kaplan
Digital Humanities Laboratory (DHLAB)
Swiss Federal Institute of Technology in Lausanne (EPFL)
CDH, INN 116, Station 14, Lausanne, Switzerland
name.surname@epfl.ch

Abstract

In recent years, many cultural institutions have engaged in large-scale newspaper digitization projects and large amounts of historical texts are being acquired (via transcription or OCRization). Beyond document preservation, the next step consists in providing an enhanced access to the content of these digital resources. In this regard, the processing of units which act as referential anchors, namely named entities (NE), is of particular importance. Yet, the application of standard NE tools to historical texts faces several challenges and performances are often not as good as on contemporary documents. This paper investigates the performances of different NE recognition tools applied on old newspapers by conducting a diachronic evaluation over 7 time-series taken from the archives of Swiss newspaper *Le Temps*.

1 Introduction

Recognition and identification of real-world entities is at the core of most text mining applications. As a matter of fact, referential units such as names of persons, locations and organizations underlie the semantics of texts and guide their interpretation. Since the seminal MUC shared-task (Grishman and Sundheim, 1996), named entity-related tasks have undergone major evolutions, from entity recognition and classification to entity disambiguation and linking (Nadeau and Sekine, 2007; Rao et al., 2013). Besides the general domain of well-written news-wire data, NE processing is also applied on specific domains, particularly biomedical (Kim et al., 2003), and on more noisy inputs such as speech transcriptions and tweets (Galibert et al., 2014; Ritter et al., 2011). More recently, NE processing has also been called upon to contribute to the domain of digital humanities, where massive digitization of historical documents is producing huge amounts of texts.

In the last few years, many cultural institutions have indeed engaged in large-scale digitization projects (Gerhard and van den Heuvel, 2015), some with a general scope, e.g. Europeana[1] or CultureSampo[2], others focusing on specific resources such as historical newspapers, e.g. Europeana Newspaper[3] (Neudecker and Antonacopoulos, 2016) or the National Digital Newspaper Program[4]. Millions of images are being acquired and, when it comes to text, their content is transcribed, either manually via dedicated interfaces, or automatically via Optical Character Recognition (OCR). If this represents a major step forward in terms of preservation and document accessibility, much remains to do in order to provide an extensive and sophisticated access to the *content* of digital resources. In this regard, information extraction techniques, particularly NE extraction and linking, can certainly be regarded as among the first steps.

Historical documents, however, pose many challenges for language technologies (Sporleder, 2010). Due to the acquisition process and/or the conservation state, input texts can be extremely noisy. Next, language(s) of earlier stage(s) may feature old vocabulary and turns of phrases and, in the case of NE extraction, can contain entities for which adequate linguistic resources and knowledge bases are missing (Ehrmann et al., 2016). Finally, as demonstrated by Vilain et al. (2007), the transfer of NE tools from one domain to another is not straightforward and performances of NE tools, initially developed for homogeneous texts of the

[1] http://www.europeana.eu/portal/about.html

[2] http://www.kulttuurisampo.fi/about.shtml?lang=en

[3] http://www.europeana-newspapers.eu/

[4] https://www.loc.gov/ndnp/

immediate past, are very likely to be affected by these phenomena.

Named entity processing tools are particularly requested in the context of historical newspapers, where historians wish to discover, among others, the "5 W's": *who did what when and where with whom*. In this paper, we experiment with the application of prototypical NE recognition and classification (NERC) approaches on a newspaper digital archive. More specifically, we are interested in investigating whether the performances of NE tools degrades when going back in time. To this end, we apply 4 NER systems on 7 document time-series (1804 to 1981) from the archives of French speaking Swiss newspaper *Le Temps*.

The remainder of the paper is organised as follows. Section 2 presents the main challenges of NE processing on historical text and discusses how they were tackled in related work. Next, section 3 describes our experimental settings, with the presentation of the source (section 3.1), the evaluation data set (section 3.2) and the systems (section 3.3). Section 4 details the results and provides an error analysis and, finally, section 5 concludes and considers future work.

2 Named Entity Processing for Cultural Heritage domains

Along with the increasing demand for language technologies support for cultural heritage domains, recent years have seen a surge in research on NE processing for historical texts. Work in this domain can be divided according to the nature of the texts which is dealt with (e.g. museum record metadata, administrative documents, genealogical data, newspapers), according to the written modality (handwritten or typeset), and according to the targeted task (NE recognition and classification, entity linking, or both). Most experiments follow one of the two following strategies: application and/or tuning of an already existing system (available in-house or publicly released, e.g. Stanford NER[5]), or use of NE processing web-services. Overall, existing work concerns a wide variety of texts covering different historical periods (from 16^{th} to 20^{th} c.), focus on different domains and use different typologies. This great variety demonstrates how many and varied the needs – and the challenges – are, but makes performance comparison difficult, not to say impossible.

Compared to the standard analysis of present-time English, very often news, the application of NE tools on historical texts faces news challenges, which can be defined as follows: (i) noisy input texts, (ii) lack of coverage in linguistic resources and knowledge bases, and (iii) dynamics of language. This section briefly elaborates on these challenges.

2.1 Noisy input texts

Texts acquired from digitized historical material can be extremely noisy. Errors can be caused either by the original source, e.g. degraded material or non standardized language, or from processing effects, e.g. poor OCR quality. They do not resemble tweet misspellings or speech transcription hesitations, problems for which adapted approaches have already been devised (Ritter et al., 2011; Parada et al., 2011).

Language variation was successfully tackled by Borin et al. (2007), who tuned an existing rule-based system with a name similarity calculation mechanism. Working on Swedish literary classics from the 19^{th}c., they were able to recognize entities belonging to 8 categories with a F-measure of 92.8%.

In some contexts, OCR errors have been handled positively, e.g. as part of the French *Quaero* project[6]. First, a comparative study of structured NE manual annotation in broadcast news vs. 19^{th}c. historical newspapers (*Le Temps*, *La Croix* and *Le Figaro* of December 1890) has been conducted, showing that OCR noise requires some guideline adaptations (Rosset et al., 2012). Three systems were subsequently evaluated on the annotated data with a F-measure ranging from 57.6% to 65.2% (Galibert et al., 2011a). Later on, Dinarelli and Rosset (2012) implemented several OCR correction strategies on this material, leading to a reduction of SER (Slot Error Rate, explained hereafter) of 8 points.

However, it appears sometimes that not even dedicated manual efforts seem to improve the quality of the recognition for historical data. Rodriquez et al. (2012) compared the performances of four NER system (Stanford, OpenNLP, AlchemyAPI and OpenCalais) on two data sets related to WWII: individual Holocaust survivor testimonies from the Wiener Library of London and letters

[5] http://nlp.stanford.edu/software/CRF-NER.shtml

[6] http://www.quaero.org

of soldiers from King's College archive. Performances are evaluated against a (small) gold standard comprising person, location and organisation names. Results on OCRed data are between 47% and 54% F-measure for the testimonies (Stanford being the most accurate), and between 32% and 36% for letters (OpenCalais performing best). When applied on manually corrected OCR, tools performed better, but not significantly. Other major identified sources of errors are different ways of naming and metonymy phenomena (esp. war ships named after people) and lack of knowledge of the systems (esp. for organisations).

2.2 Poor resource coverage

Many NE tools rely, at least in part, on existing linguistic resources and knowledge-bases, such as Wikipedia/DBpedia. However, the coverage of any knowledge base is at best uneven when going back in time. Three different phenomena are likely to impact on the lack of proper coverage in knowledge-bases: mentions of minor or not well-known entities, entities that changed name over time, and names that were used for different entities over time (ambiguity).

The general poor performance of knowledge-based systems was highlighted for example by Hooland et al. (2015). They aimed at indexing the descriptive fields of records from the Cooper Hewitt museum of New York. To this end, they developed an OpenRefine NER extension based on multiple NER web-services (AlchemyAPI, DBpedia Spotlight and Zemanta), giving the possibility for data curators to automatically annotate and link entities within records. Evaluation was done against a manually built gold standard with 4 categories, with a F-measure ranging from 10 to 60% and a low recall for all systems.

Nevertheless, others found it possible to rely on knowledge bases in order to enrich them. As an example, Huet et al. (2013) explored how to mine history from *Le Monde* French newspapers (issues between 1944-1986) by linking entities occurring in articles to YAGO referents. Entities are broadly defined (we assume all entity types of YAGO) and their recognition is done via a look-up procedure, with a Precision of 86.8% and a Recall of 77.1%.

2.3 Dynamics of language

The last source of errors, and to the best of our knowledge the least explored by research up to date, relates to the dynamics of language. Most projects dealing with historical textual data cannot assume that similar rules and conventions for the use of written language applied at all times. Some previous studies showed how older data might be more problematic. Grover et al. (2008) focused on British parliamentary proceeding from the end of the 17^{th} and the beginning of 19^{th} centuries. OCRed documents are given as input to an in-house rule-based system in order to extract person and place names. The overall performance is evaluated against a gold standard of ca. 6000 person and 3600 place names, with an F-measure of about 70% for both periods. Results are comparable for person names, but the earliest period has significantly worse performance for locations.

In order to compensate for lack of dedicated studies on the problem of language change over time and to better understand NE recognition performances on historical texts, we conducted a diachronic evaluation of different NER tools over 200 years of historical newspapers. This work is in line with both (Rodriquez et al., 2012) and (Hooland et al., 2015) who applied different NE tools on historical texts, and (Galibert et al., 2010) and (Rosset et al., 2012) who explored NE annotation on French old newspapers. However, our approach features web-based NE annotation tools – never evaluated on newspapers to the best of our knowledge – and considers time series data sets. Those times series are derived from the archives of *Le Temps* newspaper, established in the French speaking part of the Swiss Confederation.

3 Experimental setting

3.1 *Le Temps* digital archive

The Swiss newspaper *Le Temps* originates from the merger of *La Gazette de Lausanne* (GDL), *Le Journal de Genève* (JDG) and *Le Nouveau Quotidien* in 1998. Born in 1798, 1826 and 1991 respectively, these three publications compose the digital archive of *Le Temps*, which was acquired in 2008 via optical character recognition (OCR) and layout detection. Together, the *Gazette de Lausanne* and the *Journal de Genève* comprise about 1 million pages and 4 million articles spanning 200 hundred years of Swiss and international history. Taking a linguistic, historical or sociological view point, motivations to explore this collection of past events and society are manifold (Bingham, 2010), and named entity recognition can in this regard be of great assistance.

	# words		# pers		# loc		# entities	
	GDL	JDG	GDL	JDG	GDL	JDG	GDL	JDG
1804	33,773	-	417	-	990	-	1,407	-
1826	33,353	14,074	471	184	946	151	1,417	335
1841	40,784	5,558	553	70	1,137	55	1,690	125
1881	55,751	12,360	950	227	912	280	1,862	507
1921	20,117	3,587	377	47	572	136	949	183
1961	23,332	8,301	529	115	556	149	1,085	264
1981	17,759	3,672	258	79	363	56	621	135
TOTAL	299,212	65,139	3,555	722	5,476	827	9,031	1,549

Table 1: Data set statistics.

3.2 Data set

We randomly selected 40 article files from GDL and 10 from JDG for the years 1804, 1826, 1841, 1881, 1921, 1961 and 1981[7]. The choice of these years was not motivated by any specific historical events but to ensure even coverage of the period. Article files were built by parsing the XML output of the OCR system, that is to say by re-building the text from the xml-tagged token singletons, and by assembling different text blocks belonging to the same article.

The selected files were annotated according to the *Quareo* guidelines (Rosset and Grouin, 2011), which have already been used for the annotation of French historical newspapers. With this choice the present data will therefore contribute to the constitution of a larger and diversified set of NE-annotated historical newspaper corpora and, on the long run, ensure performance comparison. *Quareo* typology is both hierarchical and compositional with, on the one hand, 7 entity types and 32 sub-types which categorize entities and, on the other, 24 entity components which specify the various elements making up the entities. For the present annotation task we did not considered components and targeted exclusively Person and Location entity types, with their relative *Quaero* subtypes (*pers.ind*, *pers.coll*, *loc.adm.reg*, *loc.admin.nat*, etc.).

Manual annotation was carried out from scratch by the authors (two native French speakers, and one fluent in French) using the brat rapid annotation tool (Stenetorp et al., 2012). As noticed by Rosset et al. (2012), annotation of old texts is possible but not straightforward. Annotation was done without looking at the image of the articles, that is to say relying on the OCRed text only. In this regard, decision was made to annotate entity mentions containing OCR noise as far as the annotator could recognize and identify them (e.g. *Constap. iipopjle*). The reason why we included noisy entities is because "OCR name variants" can legitimately be recognized and can be useful in an information retrieval or text mining application context. A bot or an information seeking person would indeed certainly be interested in retrieving docs in which the original text was referring to a certain entity, whatever its OCR transcription. As for historically moving entities, annotation was done according to their more recent status (e.g. *Malte* annotated as *loc.adm.nat*). Also, it should be noted that nested entities are annotated, e.g. in *Bern University*, *Bern* is annotated as Location and *Bern University* should be – but we do not consider this type at the moment – annotated as Organization. Finally, according to *Quaero* guidelines, titles such as *M.*, *Mme*, *Mlle* are part of person names, whereas functions such as *prime minister* are not.

In order to estimate the quality of the annotation, agreement rate between the 3 annotators has been computed over 3 documents of GDL from 1826, 1921 and 1981. Fleiss coefficient (Fleiss, 1971) with boundary fuzzyness on fine and coarse-grained types corresponds to 0.88 and 0.95 resp., which can be considered as satisfactory.

Table 1 shows the overall statistics of the annotated texts. Among the two newspapers, GDL is the biggest corpus; it gathers 280 articles with 3555 person and 5476 location names, for a total of about 300k words. 1881 is the year with the most entities, 1921 with the less. JDG is more reduced, with only 60 articles, 722 persons, 827 locations and about 65k words. In both corpora, the overall number of entities first increases and then decreases. This could be connected with the evolution of articles' length, getting longer during the 19^{th} c., and shorter during the 20^{th} c.

3.3 Systems

Four systems were included in our study. With the primary condition of having parsing capacities for French language, the selected tools represent major approaches for NERC: symbolic system with ExPRESS, supervised machine learning with mXS[8] and proprietary web services offering NER functionalities with AlchemyAPI[9] and Dan-

[7] JDG starts in 1826 only.

[8] https://github.com/eldams/mXS

[9] http://www.alchemyapi.com/products/alchemylanguage/entity-extraction

delionAPI[10,11]. In all experiments systems have been applied out of the box without any adaptation.

Rule-based system This NERC system consists of a set of manually curated language-independent rules that make use of language specific lexicons encoding information about entity names and trigger words. Defined via the extraction pattern engine ExPRESS (Piskorski, 2007), rules are modelled as a cascade of finite-state grammars where units are processed in increasing order of complexity. Apart from a light pre-processing including tokenization and sentence splitting, no morphological analysis nor POS-tagging is required. In concrete terms, NE rules focus on typical patterns of person, location and organisation names, e.g. an adjective (*former*) followed by a function name (*President of the Confederation*), a first (*Ruth*) and a last (*Dreifuss*) name. Besides modifiers (*famous*) and function names (*minister*), trigger words cover professions (*guitarist, football player*), expression indicating age (*42 years-old*), demonyms and markers of religion or ethnical groups (*Italian, Genevan, Bambara, Muslim*), and more. This system is derived from the multilingual NER framework developed in the context of the *Europe Media Monitor* (EMM) (Steinberger et al., 2009), from where originates the entity resource JRC-Names (Steinberger et al., 2011; Ehrmann et al., 2016). This system is tuned to recognize at least one mention per documents and is therefore better at precision than recall. In this work only the French grammar is considered.

mXS is a supervised machine learning system which learns extraction patterns for named entities. The specificity of mXS (Nouvel et al., 2014) is that it tries to detect separately the left and right boundaries of entities, a strategy particularly useful with noisy texts such as speech transcriptions where boundary markers differ due to hesitations and disfluencies. Using data mining techniques, the model first learns extraction patterns, before applying filters and a Maximum Entropy classifier over the patterns. Its performance has been evaluated against the ETAPE French corpora of speech transcriptions (Gravier et al., 2012) with a Precision of 79.8% and a Recall of 64.9%.

AlchemyAPI The AlchemyAPI is a hybrid system which combines supervised classification and rules based on textual cues to perform NERC and disambiguation. The backbone knowledge graph is proprietary; it includes all main open KBs and entities are disambiguated towards, among others, DBpedia, Freebase, GeoNames, Census and OpenCyc.

DandelionAPI Dandelion is based on a knowledge graph built from several repositories and mostly composed of places, events, organisations and people (Parmesan et al., 2014). The backbone of this knowledge graph is DBpedia, whose textual content and internal entity relations are used to perform NERC and disambiguation. In this work both Alchemy and Dandelion are used for their entity recognition and classification capacities only.

4 Evaluation

4.1 Metrics

System performances are evaluated in terms of precision and recall for each time period and in terms of their aggregation over all entities across all documents, that is to say Micro-Average precision and recall (MAP/R), for the whole period. In both cases the harmonic mean F-measure (F1) is also reported.

As demonstrated by Makhoul et al. (1999), if these measures are good at evaluating what is correct (or not), they however do not fully nor truly account for errors, especially the F-measure. As a consequence, we additionally consider the Slot Error Rate (SER), a measure analogous to the Word Error Rate in speech recognition, computed as follows:

$$SER = \frac{D + I + STB + 0.5 \times (ST + SB)}{R} \quad (1)$$

where D corresponds to the number of *Deletions* (false negatives), I to the number of *Insertions* (false positives), ST to the number of *Type Substitutions*, SB to the number of *Boundary Substitutions*, STB to the number of *Type and Boundary Substitutions* (i.e. items with incorrect type and boundaries but having a common component with an item of the reference) and R to the total number of reference entities. The adopted weighting scheme is similar as in (Galibert et al., 2011a) and gives less importance to type or boundary substitutions. Contrarily to the previous measures, SER

[10] https://dandelion.eu/

[11] Proprietary web-services were used during May 2016. We thank both IBM (Alchemy) and Spazio Dati (Dandelion) for willingly providing free API access for the purpose of this research.

	Dandelion			Alchemy			Rule-based			mXS		
	P	R	F	P	R	F	P	R	F	P	R	F
1804	20.4	11.0	14.3	53.7	27.8	36.7	62.4	12.7	21.1	28.3	18.9	22.7
1826	20.1	9.6	12.9	46.3	31.0	37.2	61.9	14.9	24.0	26.0	22.1	23.9
1841	24.4	11.6	15.7	60.3	33.3	42.9	69.1	11.8	20.1	25.7	17.2	20.6
1881	26.0	8.7	13.1	73.7	40.4	52.2	67.2	14.0	23.2	38.8	26.5	31.5
1921	38.1	13.5	20.0	72.0	41.6	52.8	69.6	23.1	34.7	32.2	23.3	27.1
1961	39.0	22.1	28.2	73.3	51.4	60.4	67.5	25.5	37.0	41.6	27.8	33.3
1981	29.9	30.6	30.3	**75.9**	**56.2**	64.6	72.8	41.5	52.8	30.6	31.8	31.2
All years	28.1	13.6	18.4	65.7	**39.5**	**49.3**	**67.6**	18.3	28.8	32.7	23.8	27.6
Baseline	52.8	34.3	41.6	**86.7**	55.6	67.7	86.3	39.7	54.4	77.3	**72.8**	**75.0**

Table 2: Precision, Recall and F-measure for *Person* on GDL corpus, plus Baseline on Quaero corpus.

	Dandelion			Alchemy			Rule-based			mXS		
	P	R	F	P	R	F	P	R	F	P	R	F
1804	63.4	64.3	63.9	63.7	28.2	39.1	90.1	43.0	58.2	71.4	32.2	44.4
1826	60.8	64.1	62.4	59.0	25.7	35.8	85.0	45.6	59.3	69.0	33.8	45.4
1841	70.6	74.0	72.3	55.4	28.1	37.3	**91.1**	51.2	65.6	70.6	35.7	47.5
1881	51.4	68.3	58.7	55.0	33.6	41.7	77.2	53.7	63.3	62.0	38.0	47.1
1921	65.2	**75.3**	69.9	53.2	28.7	37.3	87.0	50.2	63.6	63.3	30.8	41.4
1961	54.2	69.2	60.8	60.9	27.7	38.1	82.3	34.2	48.3	68.0	30.6	42.2
1981	52.2	67.2	58.8	50.2	29.5	37.2	72.5	39.9	51.5	62.0	34.2	44.0
All years	60.4	**68.8**	**64.3**	57.0	28.7	38.2	**84.6**	46.6	60.1	67.1	34.0	45.1
Baseline	57.5	**77.7**	66.1	50.6	35.7	41.8	84.7	66.0	74.2	**85.2**	68.8	**76.1**

Table 3: Precision, Recall and F-measure for *Location* on GDL corpus, plus Baseline on Quaero corpus.

is not a figure of merit but of error, therefore the lower its value the better the performance of the system. Under high error conditions, SER can be greater than 1.

4.2 Results and Error Analysis

The discussion focuses on Tables 2 and 3 which show results for the four systems in terms of precision, recall and F-measure for the GDL data set. Tables 4 and 5 report on the same measures but with a "fuzzy" setting where boundary mistakes are accepted. Given that all systems do not follow the same annotation conventions than the one we adopted, this tolerant evaluation scheme allows for a better comparison of systems. Annotation differences include insertion or not of titles and functions in person names (e.g. `<pers>` *chancellor Adenauer* `</pers>` vs. *chancellor* `<pers>` *Adenauer* `</pers>`), and of specifiers in location names (`<pers>` *district of Nyon* `</pers>` vs. *district of* `<pers>` *Nyon*`</pers>`). Regarding titles and functions, recall that Quaero guidelines include the former but exclude the latter (cf. section 3.2); in this regard, mXS and Dandelion are penalized for they exclude titles, Alchemy for it includes functions. As for locations, *Quaero* ask for the annotation of specifiers; all systems exclude them and are penalized in the same way. Finally, Figure 1 render the same measures in a graphical manner and Tables 6 and 7 present the Slot Error Rates for the Person type. The comparison with JDG is omitted for brevity as it largely confirms those from GDL.

Baseline As we wish to assess the performance gaps of NERC tools between present and historical texts (in this case newspapers), we compute a baseline against one of the few recent gold standard for French: the test data of the *Quaero* Broadcast News evaluation campaign (Galibert et al., 2011b). It is composed of speech transcriptions of radio and TV broadcasts from the year 2010; 1386 entities of type Person and 747 of type Location[12] are annotated according to the *Quaero* annotation conventions. Baseline figures are shown in last rows of Tables 2, 3, 4 and 5. In both settings and for both types, mXS (trained on speech data) performs best, with F-measures of 75% (Per-

[12]We did not consider all annotations but only the ones corresponding to our data sets.

	Dandelion			Alchemy			Rule-based			mXS		
	P	R	F	P	R	F	P	R	F	P	R	F
1804	37.3	20.1	26.2	70.4	36.5	48.0	88.2	18.0	29.9	45.9	30.7	36.8
1826	43.3	20.6	27.9	70.2	46.9	56.2	92.0	22.1	35.6	49.8	42.3	45.7
1841	53.4	25.3	34.4	76.1	42.0	54.1	**97.9**	16.6	28.4	40.9	27.3	32.8
1881	48.6	16.3	24.4	87.7	48.1	62.1	96.0	20.0	33.1	59.9	40.9	48.7
1921	64.2	22.8	33.7	89.9	52.0	65.9	92.0	30.5	45.8	53.1	38.5	44.6
1961	55.0	31.2	39.8	89.2	62.6	73.6	94.0	35.5	51.6	58.1	38.8	46.5
1981	41.3	42.2	41.8	85.9	**63.6**	73.1	95.2	54.3	69.1	44.4	46.1	45.2
All years	48.4	23.5	31.6	82.0	**49.3**	**61.6**	**94.0**	25.4	40.0	51.6	37.6	43.5
Baseline	59.5	38.6	46.8	96.5	61.8	75.4	**97.3**	44.7	61.3	86.9	**81.8**	**84.3**

Table 4: *Fuzzy* Precision, Recall and F-measure for the type *Person* on GDL corpus.

	Dandelion			Alchemy			Rule-based			mXS		
	P	R	F	P	R	F	P	R	F	P	R	F
1804	67.5	68.5	68.0	**94.5**	41.8	58.0	92.6	44.2	59.9	76.7	34.6	47.7
1826	68.8	72.4	70.5	93.2	40.6	56.6	89.5	48.0	62.5	75.0	36.8	49.4
1841	75.1	78.8	76.9	92.4	46.8	62.1	92.0	51.8	66.3	74.4	37.7	50.0
1881	55.5	73.7	63.3	76.3	46.5	57.8	80.2	55.8	65.8	68.9	42.3	52.4
1921	68.4	**79.0**	73.3	89.9	48.4	63.0	87.9	50.7	64.3	70.1	34.1	45.9
1961	59.2	75.5	66.4	85.8	39.0	53.6	86.1	35.8	50.6	71.2	32.0	44.2
1981	58.7	75.5	66.0	76.1	44.6	56.3	83.5	46.0	59.3	69.5	38.3	49.4
All years	65.3	**74.4**	**69.6**	87.4	44.0	58.6	**87.7**	48.3	62.3	72.7	36.8	48.9
Baseline	63.9	**86.3**	73.4	75.9	53.5	62.7	86.2	67.2	75.5	84.3	71.4	**79.1**

Table 5: *Fuzzy* Precision, Recall and F-measure for the type *Location* on GDL corpus.

son) and 76.1% (Location) in normal setting and of 84.3% and 79.1% in fuzzy setting. Regarding Person, Alchemy and the rule-based (RB) systems score high in precision whereas recall is lower, particularly for RB. Dandelion is overall better than Alchemy for Location, but performs equally than RB on this type.

General observations In terms of precision, performances over all years ranges from 28.1% to 67.6% for the type Person and from 57% to 84.6% for the type Location (cf. Tables 2 and 3). In terms of recall, performances reach values from 13.6% to 39.5% (Person) and from 28.7% to 68.8% (Location). Best F-measures correspond to 49.3% for Person and 64.3% for Location. When considering the fuzzy scheme (cf. Tables 4 and 5), performances are better, particularly for Person's precision and recall which show success rates at 94% and 49.3%, respectively. Location's performances increase as well but not that greatly. In this setting, best F-measures reach 61.6% and 69.6% for Person and Location respectively. High slot error rates echo these figures, with 0.63 for Person and 0.58 for Location at minima (cf. Tables 6 and 7 for Person; Location tables are omitted).

Not surprisingly, these results do not compare with the mid-90s F-score achieved by the MUC systems and are below the usual performances on news genre; they are however in line with the figures obtained on historical newspapers in (Galibert et al., 2010; Galibert et al., 2011a). Overall, the situation is better better for Location than for Person in terms of both precision and recall, and performances show important disparities between systems.

Compared to the baseline, all systems show degraded performances. Overall, losses are more important for Person than for Location and are different among systems. mXS is the most affected, with F-measure downgraded by 40.8 points on Person and 30.2 on Location (fuzzy setting). Alchemy and RB have important losses regarding Person, Alchemy rather on precision (-14.5 points on fuzzy), RB rather on recall (-19.3). Dandelion is mainly affected on the Person type, generally, and on Location, for recall only.

Considering general performances on the historical corpus, the rule-based system stands on the podium during the first half of the period in terms of Person precision, before being overtaken by

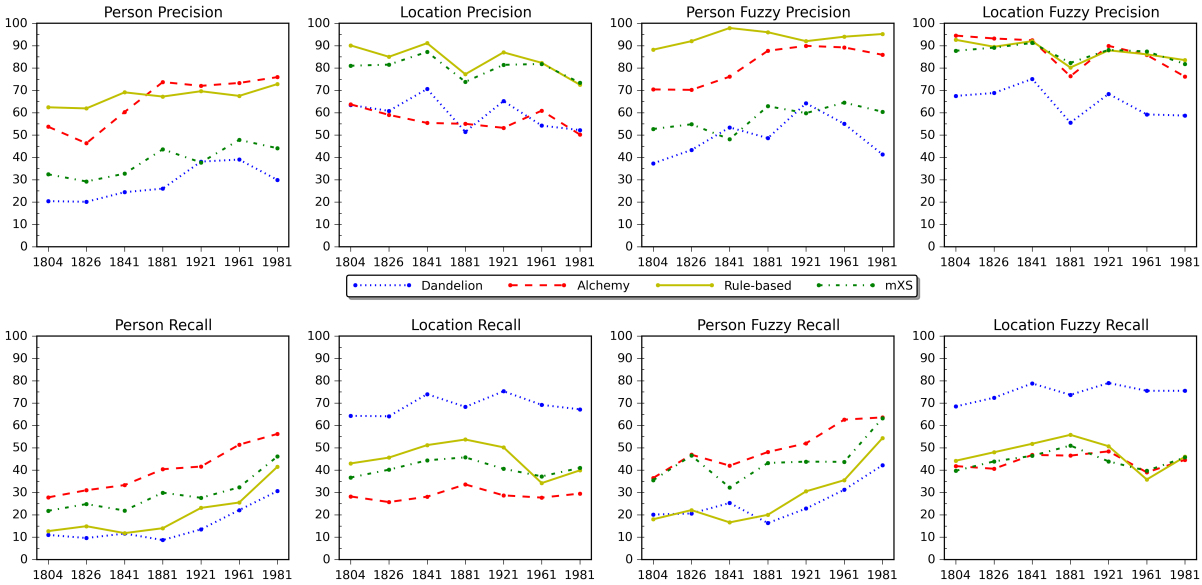

Figure 1: Precision and Recall plots for all systems, with normal and fuzzy settings.

Alchemy for the second half. It however stays the first system for Location precision, very closely followed by mXS. With respect to recall, Alchemy and Dandelion show opposite and reversed performances: Alchemy is best for Person but worst for Location, and the contrary for Dandelion. The same holds true with the fuzzy setting.

Time-based observations For both types evolution of system's precision over time is quite irregular, with several ups and downs for all systems, except RB and Alchemy which are slightly more stable for Person and Location respectively (cf. Figure 1). Similar trends can be observed under the fuzzy scheme. Contrary to what could have been expected, precision do not show clear increase over time, since the situation kind of improves for Person, and even degrades for Location. On the opposite, recall show less variability over time, with a slight but regular increase for Person towards the year 1981, and a more stable situation for Location. We may conclude that the way location names are introduced in texts is more stable than for person names, and that the contribution of knowledge bases (or gazetteers in the case of RB and mXS) is in this case more profitable.

System-based observations Considering the different systems, we observe important performances discrepancies in both absolute terms and time-related trends; the ease and the difficulties are not the same for all systems. The most stable systems over the years are RB for Person's precision and mXS for Location's recall. In terms of overall precision, Alchemy and RB are good for Persons, while mXS and RB systems are efficient for Locations. Person's top precision is reached by Alchemy in normal setting and by RB in fuzzy setting; Location's top one by RB (normal setting) and Alchemy (fuzzy setting). As for recall, Alchemy is the best for Person, Dandelion for Location, while mXS shows a better balance over both types.

Tables 6 and 7 detail the various types of errors in terms of SER variables (on Person type only; however, we also report figures on Location hereafter). For both types Dandelion has the highest number of *Insertions*; their evolution through time is irregular for Person, while they regularly decrease for Location. For all systems the number of *Deletions* evolves quite irregularly, but is lower at the end of the period than at the beginning. Systems who deleted most entities are Dandelion and RB for Persons, and Alchemy and mXS for Locations. Dandelion and RB do not confuse Person types, but can do mistakes for Location. Alchemy and mXS often mistaken Person for Location, but less Location for Person.

4.3 Discussion

This diachronic analysis allowed us to peek under the hood of different NE tools challenged with texts from historical newspapers. Despite the fact that this is a first evaluation, some trends emerge. First, performances degrade compared to

	Dandelion						Alchemy					
	I	*D*	*ST*	*SB*	*STB*	SER	*I*	*D*	*ST*	*SB*	*STB*	SER
1804	132	309	1	38	8	1.12	40	262	18	36	6	0.8
1826	123	342	3	52	1	1.05	43	241	47	75	4	0.74
1841	116	374	3	76	4	0.96	36	307	33	48	5	0.7
1881	150	753	2	72	14	1	33	464	29	73	3	0.58
1921	45	278	3	35	0	**0.91**	11	180	10	39	1	0.57
1961	122	351	10	48	3	0.95	29	191	10	59	20	0.52
1981	148	139	1	30	6	1.2	16	91	10	19	1	**0.47**
All years	836	2546	23	351	36	1.01	208	1736	157	349	40	**0.63**

Table 6: Alchemy and Dandelion SER results for type *Person* on GDL corpus.

	Rule-based						mXS					
	I	*D*	*ST*	*SB*	*STB*	SER	*I*	*D*	*ST*	*SB*	*STB*	SER
1804	2	325	0	22	8	0.83	110	270	26	49	17	1.04
1826	7	345	2	34	0	0.79	155	268	41	95	5	1.05
1841	1	438	1	27	2	0.82	152	400	61	56	11	1.12
1881	6	708	1	57	4	0.79	216	562	43	137	3	0.92
1921	4	252	1	28	6	0.73	96	229	19	57	14	1
1961	12	329	0	53	2	0.7	122	317	24	58	2	**0.91**
1981	7	112	0	33	0	**0.53**	118	138	31	37	0	1.12
All years	39	2509	5	254	22	0.76	969	2184	245	489	52	1

Table 7: Rule-based and mXS SER results for type *Person* on GDL corpus.

the adopted baseline and are lower than those observed during traditional NE evaluation campaigns such as MUC or CoNNL. However, they are in line with other work on historical newspapers.

Next, results show more irregularities over time than expected, as well as strong disparities between systems. Nevertheless, the historical trend for Location recall confirms the intuition that the more recent the texts, the more entities we can recognize. This suggest that the lexical coverage of gazetteers and/or knowledge bases (which constitutes the backbone of some systems) is lower when going back in time. Then, the significant performance drop on earlier years (especially for recall) might be due to a lower OCR quality and to text variability. We tend to discard a strong impact of language variability issues afterwards, since newspapers were commonly proofread. The same applies to OCR impact, for which an evaluation campaign is ongoing.

Finally, performances over historical newspapers vary depending on entity types. Contrarily to Persons, Location names can be expected to be mentioned in a more stable way over time; this is confirmed by higher performances on this type, especially in terms of recall and for systems relying on knowledge bases.

Regarding the best strategy to follow in order to adopt an NE tool to process historical newspapers, this analysis shows that no clear-cut solution exists: all tools have strengths and weaknesses either over time, or over specific types of NEs, or over recall and/or precision optimization. The best solution might therefore be to make a diachronic evaluation and then select or combine the best tools for a given period, a given type of entity and a given preferred application scenario.

5 Conclusion and Future work

We presented a diachronic evaluation of 4 NERC tools applied to 7 time-series from Swiss newspaper archive *Le Temps*. The evaluation spans almost 200 years and allows to understand better the behaviour of NE tools on historical data. Performances are overall lower than on contemporary texts and, interestingly, the intuition that they degrade when going back in time is only partially validated: it holds true for the Location type but not for Person.

Many directions remain open as future work. We intend to evaluate the impact of OCR errors, to expand our NE set to the full *Quaero* typology, and to consider others historical data sets. Such developments will lay the ground for advanced text mining over *Le Temps* corpus and, more generally, over historical newspapers.

References

A. Bingham. 2010. 'The Digitization of Newspaper Archives: Opportunities and Challenges for Historians'. *Twentieth Century British History*, 21(2):225–231.

L. Borin, D. Kokkinakis, and L-J. Olsson. 2007. Naming the past: Named entity and animacy recognition in 19th century swedish literature. In *Proceedings of the Workshop on Language Technology for Cultural Heritage Data (LaT-eCH 2007)*, pages 1–8.

M. Dinarelli and S. Rosset. 2012. Tree-Structured Named Entity Recognition on OCR Data: Analysis, Processing and Results. In N. Calzolari, K. Choukri, T. Declerck, M. U. Dogan, B. Maegaard, J. Mariani, A. Moreno, J. Odijk, and S. Piperidis, editors, *Proceedings of the 8th International Conference on Language Resources and Evaluation, Istanbul, Turkey, May 2012*. European Language Resources Association (ELRA).

M. Ehrmann, D. Nouvel, and S. Rosset. 2016. Named Entities Resources - Overview and Outlook. In N. Calzolari Conference Chair, K. Choukri, T. Declerck, M. Grobelnik, B. Maegaard, J. Mariani, A. Moreno, J. Odijk, and S. Piperidis, editors, *Proceedings of the 10th International Conference on Language Resources and Evaluation, Portoro, Slovenia, May 2016*.

Joseph L Fleiss. 1971. 8Measuring nominal scale agreement among many raters. *Psychological bulletin*, 76(5):378.

O. Galibert, L. Quintard, S. Rosset, P. Zweigenbaum, C. Nédellec, S. Aubin, L. Gillard, J-P. Raysz, D. Pois, X. Tannier, L. Deléger, and D. Laurent. 2010. Named and specific entity detection in varied data: The quæro named entity baseline evaluation. In N. Calzolari, K. Choukri, B. Maegaard, J. Mariani, J. Odijk, S. Piperidis, M. Rosner, and D. Tapias, editors, *Proceedings of the 7th International Conference on Language Resources and Evaluation, Valletta, Malta, May 2010*, pages 3453–3458.

O. Galibert, S. Rosset, C. Grouin, P. Zweigenbaum, and L. Quintard. 2011a. Extended Named Entity Annotation on OCRed Documents : From Corpus Constitution to Evaluation Campaign. pages 3126–3131.

O. Galibert, S. Rosset, C. Grouin, P. Zweigenbaum, and L. Quintard. 2011b. Structured and Extended Named Entity Evaluation in Automatic Speech Transcriptions. In *Proceedings of 5th International Joint Conference on Natural Language Processing*, pages 518–526, Chiang Mai, Thailand, November. Asian Federation of Natural Language Processing.

O. Galibert, J. Leixa, G. Adda, K. Choukri, and G. Gravier. 2014. The ETAPE speech processing evaluation. In *Proc. of the 9th International Conference on Language Resources and Evaluation (LREC'09)*, Reykjavik, Iceland.

J-N. Gerhard and W. van den Heuvel. 2015. Survey Report on Digitisation in European Cultural Heritage Institutions 2015. Technical report, Europeana/ENUMERATE, June.

G. Gravier, G. Adda, N. Paulson, M. Carré, A. Giraudel, and O. Galibert. 2012. The ETAPE corpus for the evaluation of speech-based TV content processing in the French language. In *Proceedings of the 6th International Conference on Language Resources and Evaluation, Marrakech, Morocco, 26 May – 1 June 2008*, Turkey.

R. Grishman and B. Sundheim. 1996. Message Understanding Conference - 6: A brief history. In *Proceedings of the 16th Conference on Computational Linguistics - Volume 1*, COLING '96, pages 466–471, Stroudsburg, PA, USA. Association for Computational Linguistics.

C. Grover, S. Givon, R. Tobin, and J. Ball. 2008. Named entity recognition for digitised historical texts. In N. Calzolari, K. Choukri, T. Declerck, H. Loftsson, B. Maegaard, J. Mariani, A. Moreno, J. Odijk, and S. Piperidis, editors, *Proceedings of the 6th International Conference on Language Resources and Evaluation, Marrakech, Morocco, 26 May – 1 June 2008*.

S. Van Hooland, M. De Wilde, R. Verborgh, T. Steiner, and R. Van de Walle. 2015. Exploring entity recognition and disambiguation for cultural heritage collections. *Digital Scholarship in the Humanities*, 30(2):262–279.

T. Huet, J. Biega, and F. Suchanek. 2013. Mining history with Le Monde. In *Proceedings of the 2013 workshop on Automated knowledge base construction*, pages 49–54. ACM.

J.D. Kim, T. Ohta, Y. Tateisi, and J. Tsujii. 2003. Genia corpusa semantically annotated corpus for biotextmining. *Bioinformatics*, 19(suppl 1):i180–i182.

J. Makhoul, F. Kubala, R. Schwartz, and R. Weischedel. 1999. Performance Measures For Information Extraction. In *In Proceedings of DARPA Broadcast News Workshop*, pages 249–252.

D. Nadeau and S. Sekine. 2007. A survey of named entity recognition and classification. *Lingvisticae Investigationes*, 30(1):3–26.

C. Neudecker and A. Antonacopoulos. 2016. Making Europes historical newspapers searchable. In *Proceedings of the 12th IAPR International Workshop on Document Analysis Systems (DAS2016)*.

D. Nouvel, J.-Y. Antoine, and N. Friburger. 2014. Pattern mining for named entity recognition. *LNCS/LNAI Series*, 8387i (post-proceedings LTC 2011).

C. Parada, M. Dredze, and F. Jelinek. 2011. OOV Sensitive Named-Entity Recognition in Speech. In

Proceedings of the 12^{th} *Annual Conference of the International Speech Communication Association (INTERSPEECH 2011)*, pages 2085–2088, Florence, Italy. International Speech Communication Association (ISCA).

S. Parmesan, U. Scaiella, M. Barbera, and T. Tarasova. 2014. Dandelion: from raw data to dataGEMs for developers. In *Proceedings of the 2014 International Conference on Developers-Volume 1268*, pages 1–6. CEUR-WS. org.

J. Piskorski. 2007. ExPRESS Extraction Pattern Recognition Engine and Specification Suite. In *Proceedings of the International Workshop Finite-State Methods and Natural Language Processing 2007 (FSMNLP 2007)*, Potsdam, Germany, September.

D. Rao, P. McNamee, and M. Dredze, 2013. *Multi-source, Multilingual Information Extraction and Summarization*, chapter Entity Linking: Finding Extracted Entities in a Knowledge Base, pages 93–115. Springer Berlin Heidelberg, Berlin, Heidelberg.

A. Ritter, S. Clark, M. Etzioni, and O. Etzioni. 2011. Named Entity Recognition in Tweets: An Experimental Study. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '11, pages 1524–1534, Stroudsburg, PA, USA. Association for Computational Linguistics.

K. J. Rodriquez, M. Bryant, T. Blanke, and M. Luszczynska. 2012. Comparison of named entity recognition tools for raw OCR text. In Jeremy Jancsary, editor, *Proceedings of KONVENS 2012*, pages 410–414. ÖGAI, September. LThist 2012 workshop.

S. Rosset and C. Grouin. 2011. Entités Nommées Structurées: guide d'annotation QUAERO. Technical report, LIMSI-CNRS.

S. Rosset, C. Grouin, K. Fort, O. Galibert, J. Kahn, and P. Zweigenbaum. 2012. Structured named entities in two distinct press corpora: Contemporary broadcast news and old newspapers. In *Proceedings of the* 6^{th} *Linguistic Annotation Workshop*, pages 40–48. Association for Computational Linguistics.

C. Sporleder. 2010. Natural language processing for cultural heritage domains. *Language and Linguistics Compass*, 4(9):750–768.

R. Steinberger, B. Pouliquen, and E. van der Goot. 2009. An introduction to the Europe Media Monitor family of applications. In F. Gey, N. Kando, and J. Karlgren, editors, *Information access in a multilingual world Proceedings of the SIGIR 2009 Workshop (SIGIR-CLIR)*, Boston, USA, July.

R. Steinberger, B. Pouliquen, M. M. Kabadjov, and E. van der Goot. 2011. JRC-Names: A Freely Available, Highly Multilingual Named Entity Resource. In *Proc. of the* 8^{th} *International Conference Recent Advances in Natural Language Processing (RANLP'2011)*, Hissar, Bulgaria, September.

P. Stenetorp, S. Pyysalo, G. Topić, T. Ohta, S. Ananiadou, and J. Tsujii. 2012. BRAT: A Web-based Tool for NLP-assisted Text Annotation. In *Proceedings of the Demonstrations at the* 13^{th} *Conference of the European Chapter of the Association for Computational Linguistics*, EACL '12, pages 102–107, Stroudsburg, PA, USA. Association for Computational Linguistics.

M. Vilain, J. Su, and S. Lubar. 2007. Entity Extraction is a Boring Solved Problem: Or is It? In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Companion Volume, Short Papers*, NAACL-Short '07, pages 181–184. Association for Computational Linguistics.

SWAN: an easy-to-use web-based annotation system

Timo Gühring[1,2] **Nicklas Linz**[2,3] **Rafael Theis**[2] **Annemarie Friedrich**[1]
[1]Department of Computational Linguistics, Saarland University
[2]Department of Computer Science, Saarland University
[3]German Research Center for Artificial Intelligence (DFKI), Saarbrücken, Germany
timo.guehring@googlemail.com nicklas.linz@dfki.de
s9rathei@stud.uni-saarland.de afried@coli.uni-saarland.de

Abstract

We present the Saar Web ANnotation system (SWAN), a lean web-based annotation system, which is optimized for annotator and project management usability.[1] SWAN is well-suited for various discourse annotation tasks, as arbitrarily large documents can be annotated seamlessly due to dynamic loading and rendering. A graph-based visualization box supports the user by providing both an overview of the existing annotations and a navigation option through the text. The admin view includes the web-based configuration of projects, annotation schemes and users. SWAN is based on JEE technology and compatible with several web browsers.

1 Overview

Manual annotation of text documents is the backbone of natural language processing (NLP) research. Among others, the annotation of linguistic phenomena is necessary for training and evaluating NLP systems. In recent years, NLP research increasingly addresses linguistic phenomena beyond the sentence level, i.e., discourse information (Webber et al., 2012). Text corpora have been annotated with discourse relations (Prasad et al., 2008; Carlson et al., 2002), temporal relations (Day et al., 2003) and coreference (Hovy et al., 2006). Discourse annotation tasks are characteristically different from sentence-level tasks in three major ways: (1) The annotator needs to have an overview of the entire document and potentially create links between spans that are far apart from each other. (2) Spans for annotation can be large, i.e., they may consist of clauses, sentences or even paragraphs. (3) The location of paragraph breaks may be relevant. Hence, unlike other existing web-based annotation systems, SWAN displays the text documents in their original formatting by default.

As the annotators are the main users of the system, we focus on optimizing usability of the annotators' view of the system. One key factor driving the development of our system was the need for an intuitive and usable interface for discourse annotation tasks such as labeling texts with event structures and temporal information, or marking paragraphs with their discourse mode. The latter comprise distinctions like *narrative*, *information*, *report*, *description* or *argumentative* (Smith, 2003). For these tasks, we have developed an editor that is responsive even for large documents, allows for easy and quick selection of arbitrary spans, as well as creating links between annotations. For example, in annotation tasks with a relatively small set of types and labels, SWAN can be configured to display all selectable labels once a span annotation or a link annotation has been created. This is especially useful for annotators who are getting familiar with a task, and allows for fast selection especially in cases where the tag set is limited.

In some of our use cases, large spans corresponding to paragraphs need to be selected. If the annotator decides that the span selection should be slightly different, deleting the annotation and renewing is time-consuming, especially if the annotation is already linked to others. Easily changing the extent of a span annotation is a novel feature developed specifically for these types of discourse annotation.

In addition, SWAN comes with a powerful admin and project management view, which provides for defining annotation schemes, managing users and tracking the progress of annotation projects.

SWAN has the following novel features:

- **Dynamic loading and rendering of the document.** This allows for arbitrarily large documents without having to split the document

[1]Web demo and code available at https://swan.coli.uni-saarland.de and https://github.com/annefried/swan.

into multiple pages, which is essential for our discourse-level annotation tasks.

- **Graph visualization.** The graph structure of the annotations is shown next to the text. When selecting an annotation either in the text or in the graph, annotations are highlighted correspondingly; the graph provides an overview as well as an option for easy navigation through the document.

- **Usability optimized for discourse annotation.** Various new features target the user experience for making changes that arise in discourse annotation tasks, such as easily changing the extent of a span for an annotation. In addition, we aim for simplicity and intuitiveness, keeping the interface and the concepts underlying the annotation scheme as simple as possible for annotators.

SWAN is a web application based on JEE technology, runs on the GlassFish server and uses a PostgreSQL database. The front-end is realized using HTML and JavaScript and is compatible with major recent web browsers. The source code, a pre-packaged WAR ready for deployment, as well as installation and set-up instructions are publicly available via GitHub. SWAN v2.0 is a stable release and the system is already in use in various projects at our department. Issue tracking and support is provided via GitHub.

2 Related work

The most similar system to ours is WebAnno (Yimam et al., 2013), a web-based and configurable annotation tool. It is the state-of-the-art annotation system for many natural language annotation tasks such as part-of-speech tagging, syntactic dependency trees or coreference. WebAnno splits longer documents into multiple pages (configurable by the annotator) and loads the next part of the document if explicitly requested or if the end of a page has been reached. In SWAN, we overcome the efficiency problem underlying this design decision by dynamically reloading content and only rendering the visible parts of the document. WebAnno uses the front-end of BRAT (Stenetorp et al., 2012), the first web-based open source annotation tool, extending its functionality mainly regarding configuration options and supported file formats. In BRAT, links are always displayed on top of the text line, which makes sense for within-sentence annotation tasks. However, if links cross sentence boundaries, as is frequently the case in our discourse annotation tasks, BRAT displays the links as ending at the right side of a line and starting again at the left side of a new line, which is somewhat counter-intuitive. In SWAN, we solve this problem by directly connecting nodes in the text, but graying out non-selected links to improve readability. Related to our work are also GrapAT (Sonntag and Stede, 2014) and rstWeb (Zeldes, 2016), which are both web-based systems focusing on annotating and displaying graph structures on top of text. Some larger-scale discourse annotation projects (Prasad et al., 2008; Carlson et al., 2002; Cassidy et al., 2014) have developed their own annotation-scheme specific tools, which are implemented as locally-running applications.[2,3,4]

3 Annotation schemes

Annotation schemes in SWAN follow a simple concept, in accordance with our intuition that the full complexity of type systems should be represented in the logic of software processing the data, but not necessarily during annotation. Annotators, who often do not have a formal background, thus can focus on a particular task without having to worry about the big picture. Annotation schemes can be configured and modified by project managers using the scheme builder. A full example of an annotation scheme is given in Figure 1.

Span annotations in SWAN consist of spans (any number of contiguous tokens) and are assigned a **span type**, e.g., *NounPhrase*, *Clause*, or *Passage*. The first step after creating an annotation by selecting a span in the text is to choose its span type. **Label sets** are defined as sets of **labels**, and apply to particular span types. They are displayed as soon as a span annotation has been created and its type has been selected. A possible label set for the span type *Passage* would be *DiscourseMode* with labels including *narrative*, *report*, *information* or *description* (Smith, 2003). Another label set applying to the type *Clause* could be *EventType* including the labels *state*, *achievement*, *activity* and *accomplishment* (Vendler, 1957). Label sets can be configured with regard to whether they are **exclusive**, i.e., whether annotators can select only one

[2] www.seas.upenn.edu/~pdtb/tools.shtml
[3] www.isi.edu/licensed-sw/RSTTool
[4] www.usna.edu/Users/cs/nchamber/caevo

label out of a set or assign multiple labels from one set to one span annotation.

Link annotations are edges from one span annotation to another. **Link types** define the type of a link annotation. For each link type, the span types of the start and end annotations need to be defined. For *TemporalRelation*, the type of both the start and end annotations could be *Event*, but the types of the start and end annotations can in general be defined separately and may differ. A link type is also associated with a set of **labels** that can be assigned to a link of this type. Link labels for our *TemporalRelation* link type would be the senses of temporal relations such as *before*, *overlap*, *includes* or *simultaneous* (Day et al., 2003).

```
<?xml version="1.0"?>
<root>
  <name>ExampleScheme</name>
  <spanTypes>
    <spanType>Event</spanType>
  </spanTypes>
  <labelSets>
    <labelSet>
      <name>EventType</name>
      <exclusive>false</exclusive>
      <appliesToSpanTypes>
        <spanType>Event</spanType>
      </appliesToSpanTypes>
      <labels>
        <label>State</label>
        <label>Event</label>
        <label>Generic Sentence</label>
      </labels>
    </labelSet>
    <labelSet> ... </labelSet>
  </labelSets>
  <linkTypes>
    <linkType>
      <name>Temporal relation</name>
    <startSpanType>Event</startSpanType>
      <endSpanType>Event</endSpanType>
      <linkLabels>
        <label>before</label>
        <label>after</label>
        <label>overlap</label>
      </linkLabels>
    </linkType>
    <linkType> ... </linkType>
  </linkTypes>
</root>
```

Figure 1: Example SWAN annotation scheme. *Please check documentation for details / updates.*

4 Functionality

4.1 Roles and projects

SWAN defines two primary roles: **annotator** and **project manager**. A project consists of an annotation scheme, a set of text documents and a set of annotators assigned to it. Project managers can create user accounts for annotators, annotation schemes and projects, while annotators can add annotations to the text documents of projects that they were assigned to. Project managers can see, edit, delete or export only the projects that they have created or that they have been assigned to, but all annotation schemes existing in the database are available to all project managers. Consistency of the underlying database is ensured as the scheme used in a particular project is immutable in the current release. For convenience, however, schemes can be copied and then modified when creating new projects. In addition, the system allows for **admin** users, typically the persons who administrate the installation. Admins have access to all projects, and only they can create or delete project manager accounts, while project managers can manage annotator accounts. In addition, project managers can view the annotations of their annotators using a "non-editable" version of the annotator's view.

4.2 File formats

In order to be able to display documents in their original formatting, SWAN provides for the upload of plain text files, and bases internal representations of annotations on character offsets pointing to spans in the original plain text document. If project managers want to pre-define annotations along with their types as the **targets** for an annotation task, they can upload this information in a separate file along with the plain text documents. Input and export formats for annotations use JSON or XML (for an example annotation scheme, see Figure 1). In addition, annotated documents can be downloaded directly in the UIMA XMI format (Ferrucci and Lally, 2004). Export is tied to projects, i.e., one zip file can be downloaded per project containing all annotations of all annotators.

4.3 Web interface components

The **project explorer** allows project managers to edit projects, i.e., assign annotators to a project or upload text documents optionally along with some pre-defined span annotations. Annotators see the projects that they have been assigned to and which documents they have or have not yet completed. The **scheme builder** allows project managers to create annotation schemes. Once saved and entered in the database, schemes can be modified by creating a copy and editing this copy. Projects need to be assigned an existing scheme at the time of their

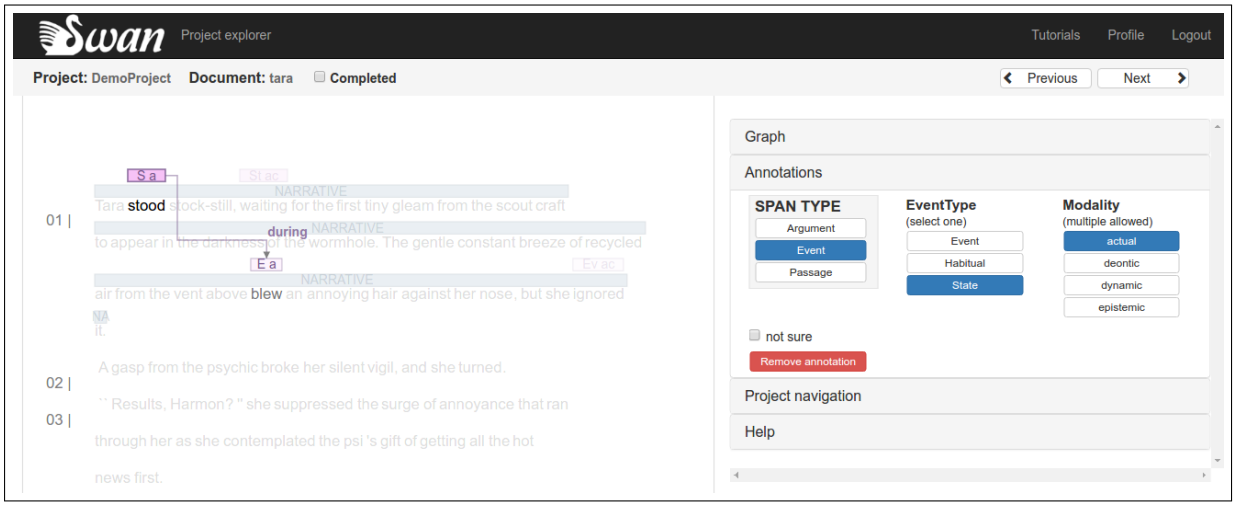

Figure 2: SWAN editor showing type / labels for highlighted annotation ("S a" = "State, actual").

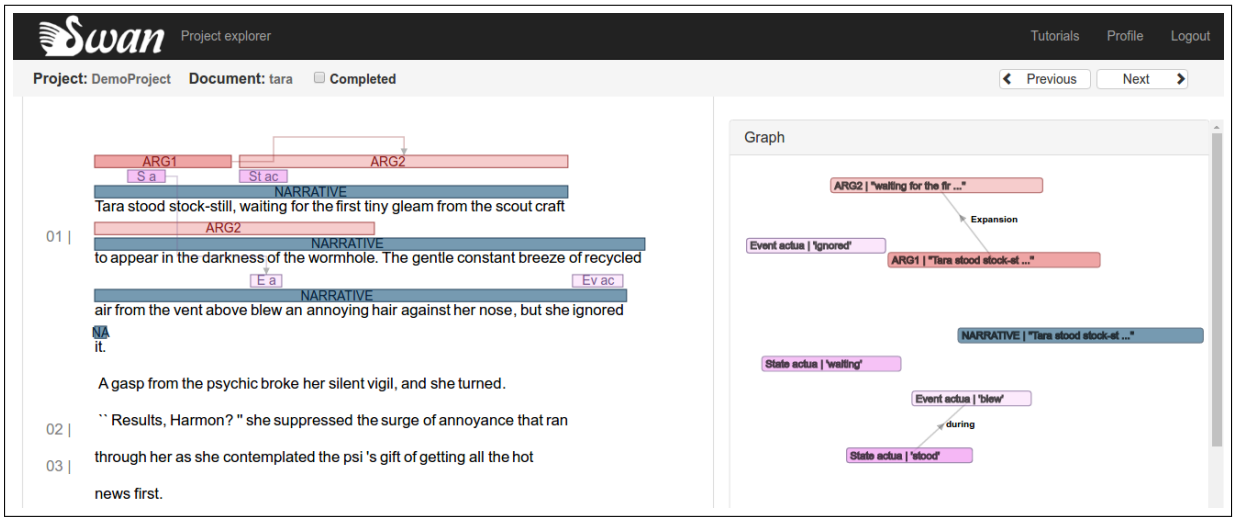

Figure 3: SWAN editor, no annotation selected, graph visualization.

creation. These restrictions ensure that the underlying database is always in a consistent state. In the near future, we will also explore options to add support for modification of schemes that are already in use. The **editor** displays the text document, the labels available for selection and the graph visualization box. Spurious spaces are removed in the editor in order to save space, but they are kept in the background, i.e., the annotation offsets relate to the original uploaded plain text file. Annotations are created by selecting tokens in the text using the mouse or a range of predefined keyboard shortcuts. This includes the quick shortcut-based selection of large spans based on selecting or de-selecting entire lines.

Links are created by selecting the start annotation and dragging the mouse to the end annotation. When removing a span annotation, all links starting or ending at this node are also automatically removed, as links cannot exist without a start and end annotation. In addition, existing annotations can be extended or made smaller token-wise to the left or right using simple keyboard shortcuts. Once an annotation has been created, the available types and respective labels are displayed.

Graph visualization. Links between annotations are visualized only selectively on top of the text by displaying links starting or ending at the selected annotation, and graying out the remaining links (see Figure 2). An optional visualization box (see Figure 3) shows the graph structure of the document. When clicking on a node, the document text view scrolls to the position of the annotation, the corresponding annotation is highlighted in the document, and the local graph structure is also highlighted on top of the text. Thus, in ad-

dition to allowing an overview of the document's discourse structure in terms of the respective annotation scheme, the graph visualization box provides an additional possibility for users to navigate through the annotations they have created for a document. Showing this graph structure directly on top of the text is often not possible as a sensible arrangement of nodes in the graph does not always follow textual order, e.g, for annotating temporal structure. We are also currently working on additional layout options for the graph visualization box such as tree structures and a layout option that is appropriate for temporal relation annotation, i.e., where the selected link labels decide on the node's arrangement in the graph.

5 Software architecture

SWAN is a Java Enterprise Edition (JEE) web-based application (see Figure 4). Being distributed as a Web application ARchive (WAR), it is easily deployable. Back-end and front-end both use lightweight RESTful web services for communication, sending data in a compact JSON format.

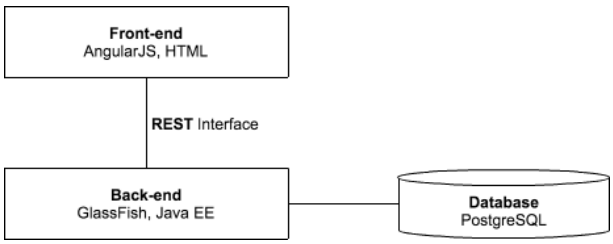

Figure 4: The system architecture of SWAN.

Back-end. The back-end of SWAN runs on GlassFish,[5] which is an open-source application server that supports JEE, and ships with a minimum of configuration effort. All user, text and annotation data is stored in the open-source PostgreSQL database,[6] enabling simple back-up solutions by regularly creating backups of the database. Texts are tokenized using the Stanford PTB Tokenizer (Manning et al., 2014). We currently support English, German, Spanish and French as well as custom white-space-based tokenization. Tokenization options for more languages will follow in future releases.

Front-end. SWAN's front-end, running in a web browser,[7] is based on the AngularJS JavaScript framework[8] and standard HTML. For data visualization, the D3 framework,[9] an industry standard for the visualization of large amounts of data, is used. It is applied to render the text, annotation boxes, graph and timeline as Scalable Vector Graphics (SVG). Using this framework, approximating which parts of the document are visible and rendering only those enables SWAN to display long texts while offering reasonable performance.

6 Discussion and outlook

SWAN is a web-based annotation system focusing on usability for annotation tasks that require the annotator to freely navigate through the entire text document. While being optimized for our annotation projects related to discourse and event structure, SWAN is generally a good option for annotation projects requiring an easy-to-use interface. SWAN does not (yet) provide an adjudication view, as in our own research projects, in order to ensure replicability, we create gold standard data from voting between many annotators rather than simply modeling an adjudicator's view of the data. The near-future development efforts in SWAN will concentrate on implementing additional options for visualizing the document's structure in the graph visualization box, including tree structures and other arrangements useful for quick navigation through the document. Future releases will also include options for monitoring inter-annotator agreement and possibly additional input formats.

Acknowledgments

We thank the anonymous reviewers, Andrea Horbach and Manfred Pinkal for their helpful comments related to this work, and Stefan Grünewald, Julia Dembowski and Janna Herrmann for contributing to SWAN's implementation. We also thank our annotators and project managers Simon Ostermann, Hannah Seitz, Damyana Gateva, Melissa Peate Sørensen, Christine Bocionek and Fernando Ardente for their for their support and useful ideas. This research was supported in part by the Cluster of Excellence "Multimodal Computing and Interaction" of the German Excellence Initiative (DFG).

[5] https://glassfish.java.net/
[6] http://www.postgresql.org/
[7] We support Mozilla Firefox, Google Chrome and Safari.
[8] https://angularjs.org
[9] https://d3js.org

References

Lynn Carlson, Daniel Marcu, and Mary Ellen Okurowski. 2002. RST Discourse Treebank LDC2002T07. Web Download. Philadelphia: Linguistic Data Consortium.

Taylor Cassidy, Bill McDowell, Nathanael Chambers, and Steven Bethard. 2014. An annotation framework for dense event ordering. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 501–506, Baltimore, MD, USA.

David Day, Lisa Ferro, Robert Gaizauskas, Patrick Hanks, Marcia Lazo, James Pustejovsky, Roser Saurı, Andrew See, Andrea Setzer, and Beth Sundheim. 2003. The TimeBank corpus. In *Corpus Linguistics*.

David Ferrucci and Adam Lally. 2004. UIMA: an architectural approach to unstructured information processing in the corporate research environment. *Natural Language Engineering*, 10(3-4):327–348.

Eduard Hovy, Mitchell Marcus, Martha Palmer, Lance Ramshaw, and Ralph Weischedel. 2006. Ontonotes: The 90% solution. In *Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers*, New York City, NY, USA.

Christopher Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven Bethard, and David McClosky. 2014. The stanford corenlp natural language processing toolkit. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 55–60, Baltimore, MD, USA.

Rashmi Prasad, Nikhil Dinesh, Alan Lee, Eleni Miltsakaki, Livio Robaldo, Aravind Joshi, and Bonnie Webber. 2008. The Penn Discourse TreeBank 2.0. In *In Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC)*, Marrakech, Morocco.

Carlota S Smith. 2003. *Modes of discourse: The local structure of texts*, volume 103. Cambridge University Press.

Jonathan Sonntag and Manfred Stede. 2014. Grapat: a tool for graph annotations. In *In Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC)*, pages 4147–4151, Reykjavik, Iceland.

Pontus Stenetorp, Sampo Pyysalo, Goran Topić, Tomoko Ohta, Sophia Ananiadou, and Jun'ichi Tsujii. 2012. BRAT: a web-based tool for NLP-assisted text annotation. In *Proceedings of the Demonstrations at the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 102–107, Avignon, France.

Zeno Vendler. 1957. Verbs and times. *The philosophical review*, 66(2):143–160.

Bonnie Webber, Markus Egg, and Valia Kordoni. 2012. Discourse structure and language technology. *Natural Language Engineering*, 18(04):437–490.

Seid Muhie Yimam, Iryna Gurevych, Richard Eckart de Castilho, and Chris Biemann. 2013. WebAnno: A Flexible, Web-based and Visually Supported System for Distributed Annotations. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 1–6, Sofia, Bulgaria, August. Association for Computational Linguistics.

Amir Zeldes. 2016. rstWeb - A Browser-based Annotation Interface for Rhetorical Structure Theory and Discourse Relations. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations*, pages 1–5, San Diego, CA, USA.

Challenges of error annotation in native/non-native speaker chat

Sviatlana Höhn, Alain Pfeiffer, Eric Ras
Luxembourg Institute of Science and Technology
Esch/Alzette, Luxembourg
sviatlana.hoehn@list.lu, alain.pfeiffer@list.lu,
eric.ras@list.lu

Abstract

This work addresses challenges related to error annotation of conceptually oral learner language such as instant messaging. The analysis is based on a corpus of German longitudinal native/non-native speaker instant messaging dyadic conversations. We show that deviations from language standard in instant messaging can be caused not only by lack of knowledge or high production pace, but also by speakers' competence in selection of interactional resources for the regulation of the social closeness, leading in this case to two contradicting identities of the same speaker: a competent language learner and a competent instant messaging user. We discuss the consequences from the perspective of language understanding and automated error correction in chat.

1 Introduction

Automatic processing of learner language is of interest for applications supporting writing activities, computer-assisted language learning and human-machine communication with non-native speakers. Because statistical natural language processing tools are mainly trained on texts produced by native speakers, their accuracy is much lower if applied for textual data produced by non-native speakers who have not yet fully mastered the language in which they interact. An additional level of complexity of such an analysis is introduced if medially written but conceptually oral (Koch, 1994) learner language needs to be processed automatically. Examples of conceptually oral but medially written interaction are chat and instant messaging (IM).

Chat and instant messaging have been studied as an additional (curricular and extracurricular) resource for language learning (Fredriksson, 2012; Marques-Schäfer, 2013; Tudini, 2010). To support language learning in chat, automatic error recognition and correction may be required. In this article we focus on native/non-native speaker instant messaging communication from the perspective of the error annotation. The error analysis needs to serve two needs: (1) natural language understanding in chat with an artificial conversation partner that helps to practice conversation in a second language, and (2)automatic error corrections in chat with such an agent or an Artificial Conversational Companion (ACC) (Danilava et al., 2013). The ACC should not play a role of a teacher or a tutor, but rather behave like a more knowledgeable and helpful peer. Because of the oral character of chat formulated by the concept of conceptually oral language, we need to discover errors that are potentially addressable in a chat-based Conversation-for-Learning (Kasper, 2004) as recorded and compiled in the dataset used for this analysis. A Conversation-for-Learning brings together participants because of their linguistic statuses of native and non-native speakers and combines elements of an informal conversation and a language classroom (e.g. sub-dialogues focusing on linguistic matters such as error corrections, see (Höhn, 2016) for a detailed analysis of such sub-dialogues). This work aims at answering the following research question: *What are the challenges in error annotation in conceptually oral learner language?* This question may have different answers depending on the purpose of error annotation. For instance learner language understanding and automated error correction performed by an ACC during the talk with a language learner may infer different sets of constraints and requirements for error recognition and error annotation. In addition, this question may have different answers for different speech exchange systems (e.g. informal chat, Conversation-for-Learning and form-focused language classroom).

We use a publicly available corpus of longitudinal dialogues between German native speakers and advanced learners of German as a second language (Höhn, 2015). The speakers in the corpus and in all examples in this article are encoded with N for native speakers and L for non-native speakers (learners). We keep the original turn and speaker numbers as well as orthography. English translations are added in italics. In addition, we had access to the unpublished part of the data collection which includes retrospective interviews with each participant in order to get insights into their linguistic choices. We apply a state-of-the-art error annotation scheme for German to answer the research question. The annotation scheme was created for the FALKO corpus (Reznicek et al., 2012) and is highly reused by other corpora discussed in the next section. In addition, we apply methods of Conversation Analysis (Markee, 2000) to analyse participants' selection of interactional resources in chat, such as specific forms of orthography.

2 Learner corpora and error annotation

Error-annotation of a corpus assumes a non-ambiguous description of the deviations from the norm, and therefore, the norm itself. A creation of such a description may be even problematic for errors in spelling, morphology and syntax (Dickinson and Ragheb, 2015). Moreover, different annotators' interpretations lead to huge variation in annotation of errors in semantics, pragmatics, textual argumentation (Reznicek et al., 2013) and usage of specific non-native language forms (Tetreault and Chodorow, 2008). Multiple annotation schemes and error taxonomies have been proposed for learner corpora, for instance (Díaz-Negrillo and Domínguez, 2006; Reznicek et al., 2012). Because error taxonomies are language-specific, we focus only on error annotation in German learner corpora in this article.

The situation with German error-annotated learner corpora is that there is a very small number of corpora, and only a small part of them are publicly available. The website[1] "Learner Corpora around the World" lists in May 2016 only 11 German learner corpora, 10 medially written and 1 spoken. In addition, there are a few publications about German error-annotated corpora not mentioned on the web page. Table 1 provides an overview on German learner corpora of which we were aware of at the time of writing this article. The table includes only information about the German part for multilingual corpora LeaP (Gut, 2009) and MERLIN (Boyd et al., 2014).

The major conceptual work on the annotation scheme and error taxonomy was accomplished by the FALKO team (Reznicek et al., 2013; Reznicek et al., 2012) and frequently re-used by the followers (German part of MERLIN (Boyd et al., 2014), EAGLE (Boyd, 2010), WHiG (Krummes and Ensslin, 2014)). WHiG is part of FALKO but contains texts from native speakers of British English who are intermediate learners of German.

The error-annotation for the mentioned corpora was approached in the following ways. The LeKo-corpus was created earlier than FALKO by the same principal investigator (Lüdeling et al., 2010). The corpus is accessible though FALKO platform. The researchers elaborated an error taxonomy on a small learner corpus of 30 texts that were written manually and then re-typed to make the resources digitally available and analysable. The difficulties with error annotation that were faced by the annotators of the LeKo corpus were taken into account in the annotation definition phase for the FALKO corpus. Specifically, some of the errors can be tagged differently depending on the *target hypothesis* - how the learners' intention is interpreted by the annotator. Dealing with such ambiguities became an issue for learner corpus annotation.

A multilevel annotation was introduced in the FALKO corpus in order to deal with different target hypotheses (Lüdeling et al., 2005). The minimal, first target hypothesis (orig.: *Zielhypothese*) ZH1 aims at sentence normalisation and is limited to only orthography and morpho-syntax, it is expected to make the sentence or utterance "understandable" for NLP tools. The second target hypothesis ZH2 should address all other types of errors, like semantics, lexical choice, pragmatics and style (Reznicek et al., 2012).

An extension of FALKO annotation schema has been suggested in the EAGLE corpus of beginning learner German where error numbering was introduced to deal with overlapping errors (Boyd, 2010). Multiple target hypotheses were handled by setting a preference for the target hypothesis which minimises the number of annotated errors.

ALeSKo is a corpus of annotated essays of advanced learners of German with Chinese as L1

[1] https://www.uclouvain.be/en-cecl-lcworld.html, retrieved on 31 May 2016

Title	L1	GFL level	Data type	Size	Error-annotated	Available
ALeSKo	Chinese	Different	Written texts	43 texts	Partial	Yes
CLEG13	English	B-C	Written texts	731 texts	NA	Yes
FALKO	Many	Intermed. - advanced	Written texts	Under development	Yes	Yes
WHiG	English	B2	Written texts	279 texts	Yes	Yes
MERLIN		A1-C2	Written examinations	1033 texts	Yes	Yes
LeKo	Many	Different	Written texts	30 texts	Yes	Yes
LeaP	Many	Different	Speech	183 records of 2-20 min	No	Yes
EAGLE		Beginners	Online work book, essays	50 WB & 81 essays	Yes	Yes
LINCS	English	Intermed.-advanced	Written texts, longitudinal	Under development	NA	No
ADS	English	Beginner-intermed.	Threaded discussion, chat, essays, longitudinal	Under development	NA	No
Telecorp	English	Different	Email, IM, essays	1,5 mio words	No	No
deL1L2IM	Russian	Advanced	IM	52000 tokens	Partial	Yes

Table 1: German learner corpora in May 2016

(Zinsmeister and Breckle, 2012). The annotation contains manual marks of topological fields (fields and error marking), referential expressions (definiteness, specificity, target hypothesis) and Vorfeld use. The subject of the ALeSKo study was coherence in learner texts based on the annotation of syntactic, referential and discourse information. German-L1 part of the FALKO corpus were used for L1-L2 comparison. A specific focus of the annotation in ALeSKo lies on referential expressions (Breckle and Zinsmeister, 2010), which are also in general an important area of NLP research.

A specific feature of CLEG13 corpus is that it has a longitudinal core of texts produced by students from their first year to their final exams (Maden-Weinberger, 2015). The corpus is accessible through FALKO platform.

In contrast to the written resources described above, the LeaP corpus includes phonologically annotated speech recordings of German and English learners of German (Gut, 2009). The corpus includes readings of nonsense word lists, readings of a short story, retellings of the story and free interviews.

The corpus KoKo is part of the project Korpus Südtirol, and focuses on German as a first language learned in South Tirol by school pupils (Abel et al., 2014). The corpus of German emails posted to USENET users described in (Becker et al., 2003) consists of ca. 120 000 sentences. An error typology of orthographic, morphological, morphosyntactic, syntactic and syntactic-semantic errors was taken as a basis for the error-annotation, however, only 16 error types from the typology were used for the corpus annotation.

The deL1L2IM corpus used for this work contains 72 dialogues of the duration between 20 and 90 minutes produced by pairs of German native speakers and advanced non-native learners during multiple weeks of IM interaction. Error annotation was performed only for selected types of errors that have been corrected by native speakers. A systematic error annotation of the dataset has been left for a future study (Höhn, 2015).

As (Meurers, 2009) notes, the annotation of learner corpora is mainly focused on annotation of learner errors, however, annotation of linguistic categories in learner corpora is also of interest. To create stable models of learner language for statistical NLP tools, information on occurrences of linguistic categories and their dependencies is required. This need is approached by linguistic annotation of learner corpora, similar as it has been done for native-speaker language. Examples of lin-

guistic annotation in learner corpora are (Amaral and Meurers, 2009) who focused on tokenisation in Portuguese interlanguage, and (Díaz-Negrillo et al., 2010) addressing the problem of POS-tagging in interlanguage. Related to the annotation of conceptually oral language, the challenge of POS-annotation in chat language has been addressed by (Bartz et al., 2014). The concept of grammaticality is applied to approach problems of syntactic annotation in learner language in (Dickinson and Ragheb, 2015).

Most of the error-annotated corpora consist of argumentative essays, and the developed error taxonomy is good for error-tagging in essays, but needs further elaboration to be fitted for conceptually oral language like instant messaging exchange. In contrast to a writing assistance program that has to (ideally) identify and correct *every* error, only a small amount of all errors are usually corrected in an ordinary conversation. Even in a language classroom, not every error is corrected in a fluency context. Therefore, there is a need to distinguish errors that could be potentially corrected in a Conversation-for-Learning from those, which should not be addressed to.

3 Language standard, chat conventions and L2 errors

In chat data, some deviations from the standard German do not count as an error. Sometimes it is even explicitly negotiated by the participants that, for instance, writing everything small will be declared as correct. Therefore, in addition to the objective identification of linguistic errors (difference between the produced language and the language standard), chat language needs to be analysed through the lens of conventions that are valid for the specific communication medium (chat in this case) and accepted by the interaction participants. This means that it cannot be completely defined in advance for chat, what will be an accepted deviation covered by conventions and what will "count" as an error that could be corrected, for instance:

1. Quick typing: everything that speeds up the typing pace does not count as errors: ignore capital letters in sentence and noun beginning, sentence punctuation.

2. Expressivity: word stretches (we found *Tor* with 62 *O*'s in it), uppercase, special symbols, punctuation symbols, quotes and parentheses, as well as various combinations of all of them.

3. Minor misspellings: typos are not important.

4. Oral style: not every utterance is a full sentence, word order is similar to oral.

There are explicit negotiations of typing rules. In all sequences that we found, we observed the following:

1. If participants engage in negotiations of spelling conventions, such negotiations are always initiated by the native speaker.

2. Production pace and conceptual orality of the interaction are the reasons for deviations addressed in chat, but not a lack of knowledge.

3. Deviations from language standard for the purpose of expressivity are not perceived as errors by chat participants.

Participant's linguistic identities (native or non-native speaker) also play a role in the selection of the applied spelling rules. For instance, participant N01 (native speaker of German) saw himself in chat with learners as a role model with respect to orthography. This is analysable in his use of, for instance, capital letters at the beginning of the nouns and utterances. N01 explains in the retrospective interview that he tried to write in according to German standard

> weil ich gegenüber nicht-deutschen-muttersprachlern versuche, die deutsche sprache so gut wie möglich in wort und schrift zu verwenden.
> *because I am trying to use written and oral German language as good as I can in communication with Non-German native speakers.*

However, N01 uses lowercase-only spelling during the interviews as opposed to the standard-compliant spelling that he chose to use in chat with non-native speakers. Thus, the orthography in chat which N01 uses with different partners is *recipient-designed*. Orthography compliance becomes an *interactional resource* in chat.

4 Orthography and social closeness

The presence of a high number of deviations from the language standard in text chat has been usually explained by a pressure to type quickly and demand for a high production pace in CALL studies

(Loewen and Reissner, 2009). However, language learners report that they had (or took) their time to use additional resources (such as dictionaries) for dealing with trouble in comprehension and production. Hence, the production overhead necessary for a standard-conform language in chat might be caused by participant's understanding of their social roles of language novices and language experts, and used for the regulation of social closeness.

Example 4.1. Mutual dependencies between orthography and social closeness.

1	L03	Hallo! Entschuldigung, Ich weiß nicht, wie heißen Sie. Ich bitte um Verzeihung, ich habe total über heutige Unterhaltung vergessen. Ich schäme mich, wirklich, aber ich war beschäftigt, und musste dringend einige Probleme lösen, deshalb habe ich total über den Chat vergessen- ich bitte noch ein Mal um Entschuldigung, und verspreche, dass es nie wiederholen wird. Ich hoffe, dass unser Chat wird uns Spaß machen. mit freundlichem Gruß, L03_FirstName L03_LastName! *Hello! I am sorry, I don't know your [III p. pl.] name. Please forgive me, I totally forgot about [* error: wrong preposition] today's conversation. I feel ashamed, really, but I was busy, and had to solve several problems urgently, this is why I totally forgot about [* error: wrong preposition] the chat - please forgive me again, I promise that it will never happen again. I hope that our chat will be pleasant. best regards, L03_FirstName L03_LastName!*
2	N02	Hallo L03_FirstName, das ist überhaupt kein Problem! Ich hoffe, alle Probleme sind gelöst und wir können ein bisschen chatten. *Hello L03_FirstName, it is absolutely no problem! I hope, all the problems are solved and we can chat a little bit.*
3	L03	Ja, natürlich! wie heißt du? *Yes, of course! what is your [II p. sg.] name?*
4	N02	oh Entschuldigung, ich heiße N02_FirstName, bin 27 Jahre alt und wohne in München. *oh, I'm sorry, my name is N02_FirstName, I am 27 and live in Munich.*
5	L03	sehr angenehm! und ich bin 21 und wohne in Vitebsk, Belarus! *very pleasant! and I am 21 and live in Vitebsk, Belarus!* nice to meet you! and I am 21 and live in Vitebsk, Belarus
6	N02	oh, ich bin schon alt ;) *oh, I am already old [smile]*
7	N02	warst du schon mal in Deutschland? Ich war noch nie in Belarus *have you already been to Germany! I have never been to Belarus*
8	L03	ja, aber ich bin schon verheiretet))) *yes, but I am already married [smile]*
9	N02	oh echt?? wow! seit wann denn, wenn ich fragen darf? *oh really?? wow! may I ask you, how long?*

Example 4.1 presents the very beginning of the talk between L03 and N02. Because the participants have never met before, L03 does not know, who is on the other side of the connection. She comes too late to her first appointment and formulated her first message (turn 1) to her chat partner in a very polite way using a polite German form of address *Sie* (III p, pl., no English equivalent). In addition, she produces an email-like turn - conceptually closer to written than oral language - according to German spelling standard and closes if with a "best regards + name" untypical for instant messaging.

L03 produces multiple morpho-syntactic and semantic errors, however, her phrases start with a capital letter (except of the closing expression), and she is doing her best in positioning herself as a competent German speaker. N02 answers with a "no problem", and her message satisfies the German language standard, too. L03 switches from *Sie* to *du* (you, II p. sg.) in turn 3. In addition, she changes the spelling in the second phrase starting with a small letter instead of a capital. N02 responds with changed applied standard in turn 4 writing only nouns with an initial capital letter.

The participants continue with the rule "write only nouns with a capital letter". Shorter time intervals between turns 5-9 in Example 4.1 show how higher engagement leads to higher talk pace and therefore higher production pace. Deviations from language standard are the price for the typing pace, but in addition, they express a higher grade of engagement and social closeness.

There are mutual dependencies between participants' choices in terms of language standard. A closer look at the native speaker N02 and her partners L03, L04 and L05 helps to understand how participants deal with spelling and punctuation conventions, and how they influence each other. We discuss here only the results, the original data can be obtained from ELDA (Höhn, 2015). N02 behaves differently with her different partners:

L03 Both participants start with the standard-compliant spelling and shift then to a version where they move between standard-compliant spelling and "write-only-nouns-with-a-capital". L03 starts with *Sie* but switch to *du* in turn 3.

L04 starts with a "relaxed" version of spelling: only nouns are written with a capital, a very oral style. N02 starts with a norm-compliant version but adapts to non-native speaker's spelling version after ten turns. Later on, both participants even use lowercase for all

words. L04 starts with *du*. Overall chat of this pair can be characterised as very oral: short phrases, quick, many short turns.

L05 starts with a norm-compliant orthography and *Sie*. L05 makes lexical errors in her first turn. N03 replies with *Sie* but she decides to write the first word in each sentence small. Later on, L03 changes between a norm-compliant spelling and the relaxed "first-letter-small" version. L05 adopts this way of spelling from time to time. In the second chat, L03 start with *du* (first turn in this meeting) using proper spelling, but switches later to the relaxed "first-letter-small" version. It remains an open question if N03 noticed that L05 is not that much an independent language user (compared to the others) and shows her, how to do "chat-in-German".

The other native speakers in the dataset prefer to keep the same orthography style with all their partners: N01 presents himself as a role model, N03 prefers to optimise the spelling to increase the typing pace and types everything with lowercase, and N04 normally types all nouns with an initial capital, but starts all new sentences with a small letter.

5 Learner error annotation

In order to test the error taxonomy, we selected an initial set of data consisting of 481 questions produced by language learners. The error-annotation of the questions was performed according to the annotation guidelines for FALKO Corpus of German learner language (Reznicek et al., 2012; Reznicek et al., 2013). ZH1 was constructed according to the rules of standard German grammar and orthography with Duden dictionary as a reference. Semantics, lexical constructions and pragmatics are the subject of the extended, second target hypothesis ZH2. Example 5.1 shows the two target hypotheses for a sample question.

Example 5.1. Creating target hypotheses for error correction in questions.

402 L08	und um wieviel Uhr gehst gewöhnlich zum Bett?
	and at what time do you normally go to the bed?
ZH1	Und um wie viel Uhr gehst du gewöhnlich zum Bett?
	And at what time do you normally go to the bed?
ZH2	Und um wie viel Uhr gehst du normal ins Bett?
	And at what time do you normally go to bed?

The questions have been annotated by two human annotators: one German native speaker and one non-native speaker with a near-native level of German proficiency. Both annotators had sound background knowledge in corpus annotation and were experienced users of instant messaging. The following issues were faced when annotating errors in chat according to FALKO guidelines. First, special symbolic and orthographic means of expressivity used in chat must be classified as errors according to Duden and FALKO error annotation guidelines. Second, FALKO annotation guidelines do not provide any specific instruction for the cases where the errors in the verb make more than one target for the verb possible.

Example 5.2 illustrates one of the cases. This error has been corrected by the interaction partner of L09 in the dialogue and both possible targets for the erroneous question were addressed in the correction. Therefore, having in mind the application where corrections should be automatically generated in a conversation, we add both target hypotheses to the annotation.

The differences between the original learner's utterance and the two target hypotheses help to classify the errors and to generate corrections. In addition, it allows to analyse empirically what normalisation steps are really required for automated language understanding.

Example 5.2. Ambiguous target hypotheses.

135 L09	gefiel dir das studium leicht?
	Unclear target: Was the study easy for you? or Did you like your study?
ZH1a	Gefiel dir das Studium?
	Did you like your study?
ZH1b	Fiel dir das Studium leicht?
	Was the study easy for you?
136 N04	es gefiel mir, aber es fiel mir nicht immer leicht ;-)
	I liked it but it was not always easy for me
137 N04	("gefallen" = "etwas schön finden",
	("to like" - "to find something pretty",
138 N04	etwas fällt jemandem leicht = man hat keine Mühe damit)
	something is easy for someone = one has no effort with it)

However, chat conventions allow writing everything with small letters only and do not consider typos as errors that need a correction. This is why information about potential correctability of the errors in chat need to be encoded in the error annotation. Additional rules for exceptions need to be specified when deviations in orthography and punctuation are used as a means of expressivity. Therefore, we introduced the "real" error flag with the purpose to identify all errors that are *potentially addressable* in chat. The conventions that we take

into account for the "real" error flag are restricted to orthography and allow to:

1. start an utterance, a new sentence and nouns with a small letter,
2. write lowercase or uppercase or camel-case,
3. use punctuation and special symbols for the purpose of expressivity (emoticons),
4. omit punctuation and to use emoticons to separate turn-constructional units,
5. produce word stretches.

These rules are consciously applied by chat participants while typing. In addition, there are misspellings which are the result of a high typing pace and not lack of knowledge. They also do not qualify as errors in chat and are not considered as "real" errors. There are two exceptions that we annotate as real errors:

1. If a speaker repeats the same misspelling several times and the misspelled word sounds exactly as the correctly spelled word.
2. If it is a special, difficult case where even native speakers often make mistakes, for instance *ziemlich*.

With Duden as a reference for the language standard, 428 questions would contain an error. However, only 136 questions contained "real" errors. Only 21 of all potentially addressable errors in questions have been corrected in conversation by the native speakers. This low number of corrections is mainly explained by the type of the speech exchange system recorded in the corpus. An artificial conversation partner will need to decide in real-time, which of the potentially addressable errors may trigger a correction. This problem has been captured in a correction decision model and discussed in (Höhn, 2016).

As already reported in earlier academic publications, finding a target hypothesis may be a hard problem even for human annotators (Reznicek et al., 2013). We faced the same issue in our work. More specifically, ZH1 and ZH2 may correct different, mutually excluding errors.

Example 5.3 illustrates an error in plural in a non-native-like expression. ZH1 corrects only the error in the plural, confirming the use of the non-native-like expression. ZH2 corrects the non-native-like expression, but does not address the error in the number. Both errors can be hardly address by one target hypothesis.

Dealing with multiple target hypotheses will also be an issue for a computer program that should produce a correction. If only one target hypothesis should be presented in the correction, then criteria for selection need to be specified. As Example 5.2 shows, this is not always possible. If multiple target hypotheses can be presented to the user in a writing assistance program, then the user will have to choose one of them for the correction. This may be less helpful for the user if he or she does not have the necessary level of linguistic competence in the foreign language to make this choice. In both cases the program will need to guess what the user could have meant.

Example 5.3. Different target hypotheses correct different errors. Trouble sources are underlined. Target hypotheses are added in the bottom.

79	L03	)))) ja, wahrscheinlich! Sind die Grenze des Schuljahres von Urlaubssaison in Beiern abhängig? *yes, probably! Do the border of the school year [* errors: verb-subject number congruence, uncommon expression] depend on the holiday season in Beiern [* error: spelling]?* yes, probably! Do the borders of the school year depend on the holiday season in Bavaria?
80	L03	* Bayern * Bayern [self-correction] **Bavaria*
81	N02	ja genau! ist das bei euch auch so? *yes, exactly! is it like this in your place, too?*
	ZH1	Sind die Grenzen des Schuljahres von der Urlaubssaison in Bayern abhängig? *Do the borders of the school year depend on the holiday season in Bavaria?*
	ZH2	Sind die Ferienzeiten von der Urlaubssaison in Bayern abhängig? *Do the school holidays depend on the holiday season in Bavaria?*

6 Findings and discussion

Conceptually oral learner language such as instant messaging and chat introduces additional levels of complexity in error annotation as compared with conceptually written learner language (e.g. essays). In this work we make an attempt to discover these challenges on an example of a German native/non-native speaker instant messaging corpus deL1L2IM (Höhn, 2015).

Section 4 shows that participants of an instant messaging chat use deviations from language standard as a interactional resource to regulate social

closeness and to present themselves as members of specific social categories. Such categories are for instance a native speaker who positions himself as a role model, as well as a competent non-native speaker who is a competent instant messaging user. In this way, this work supports observations described in (Lee et al., 2012) that the concept of language correctness (or grammaticality in (Lee et al., 2012) is not an absolute category, but may vary depending on the level of formality or social proximity.

Using deviations form linguistic standard belongs to the interactional competence in computer-mediated communication and therefore, such deviation should not be always classified as errors. Such deviations in German native/non-native speaker chat mainly include orthography of German nouns and initial letters of an utterance, but also oral verb forms (*hab* instead of *habe* and more oral forms of question (e.g. declarative utterances which are functional questions, see also (Stivers and Enfield, 2010) for question classification).

In addition, the chat conventions covering such deviations (what is allowed) may vary for different pairs of speakers and change for the same pair of speakers with the time. This is the consequence of the variance in the level of social closeness for different pairs of speakers, and changes in the grade of social closeness that may occur with the time for one pair of speakers, since their relationship may change. Nevertheless, the set of potentially correctable errors seems to be quite stable for the specific speech exchange system (here Conversation-for-Learning). It was acceptable even for those native speakers who preferred typing according to German orthography standard, if learners typed with deviations in orthography (e.g. omission of initial capital letters). In addition, the types of deviation produced by the native speakers in the dataset differ from those produced by the learners. For instance, usage of oral forms of verbs (e.g. *hab* instead of *habe*, Engl.: *have* I p. sg) was only observed in utterances by native speakers. This may obscure learner's familiarity with oral German which is not explicitly covered in language classes or by language tests. Thus, some types of deviations may signal higher levels of familiarity with specific aspects of the foreign language use.

With this observations, the error annotation in conceptually oral learner language needs to cover at least one additional layer, namely the layer of potentially correctable errors in order to serve the need of automated error correction in conversation (although the occurrence of a potentially correctable error does not immediately trigger a correction, see (Schegloff, 1988)). We approached the problem of identification of potentially correctable errors by the "real" error flag although we also considered other possibilities which we discuss below.

One possible approach to identify such errors could be a comparison by the chat conventions applied by the native speakers in chat. However, this approach has at least two shortcomings. First, our data show that some native speakers may make their social roles of language experts more important than their roles of proficient IM users, and purposely avoid any deviations from language standard (see Sec. 3, example with N01). This pattern is not necessarily taken up by the learners. In this case, potentially correctable errors would include all those minor deviations that normally do not count as errors in a chat-based Conversation-for-Learning. Second, this approach would automatically put the native speaker in the position of a language expert, and the non-native speaker in the complimentary category of a language novice. However, being a native speaker of a language does not necessarily correlate with high language proficiency. This is why the notion of expertise or *differential language expertise* is suggested by the CA community as more appropriate to describe the socio-linguistic data in native/non-native speaker communication (Hosoda, 2006). For these reasons we suggest to analyse errors and deviations in learners' utterances independently from utterances of their native speaker partners.

Another way to identify correctable errors in IM chat would be looking at those errors that have been corrected by native speakers. The main limitation of this approach is that only a small number of errors received a correction in the dataset, and the number of corrected errors highly varied among different native speakers: some learners produced a high number of errors, but they were not corrected by their partners. An identification of a potentially correctable error in chat does not automatically mean the necessity of a correction, which is also confirmed by the numbers in our dataset (only 21 corrections of 136 "real" errors). Therefore, we relied on the intuitive concept of the "real" error in our analysis.

Deviations from language standard in chat may

occur because they are produced by instant messaging speakers consciously with the purpose of regulation of the social closeness. They can be also produced unconsciously due to high typing pace. The high typing pace, in turn, may be caused by the time pressure, but also by a high participants' engagement in talk. Lack of knowledge is rarely the reason for such deviations. However, it might be important for the language understanding components to find a normalised, grammatical equivalent to learner's utterance. The analysis of the set of learners' questions in Sec. 5 shows that the first target hypothesis ZH1 already serves this need for the analysed dataset. However, the effectiveness of ZH1 for this purpose may be different for less advanced language learners.

Section 5 also shows that learner errors make it sometimes necessary to consider several target hypotheses on each level (Example 5.2). In conceptually oral learner language, this needs to be done not only to guarantee the correctness, but also to maintain intersubjectivity and mutual understanding in the talk. Therefore, additional sub-levels in error annotation may be needed, as suggested in Example 5.2. These additional levels of error annotation can be used by the agent in real-time to capture multiple possible meanings of learner's utterance. Possible responses to such utterances include error corrections with disambiguation like in Example 5.2, or repair initiations (frequently called clarification requests in academic publications in NLP community, see for instance (Schlangen, 2004)).

7 Conclusions and future work

While FALCO annotation guidelines (Reznicek et al., 2012) already provide a comprehensive basis for error annotation in conceptually written learner language, annotating conceptually oral learner language brings the annotation task to a higher level of complexity. Specifically, there is a need to distinguish between deviations from language standard which can be addressed as an error in chat, and all other types of deviations which do not count as error in chat due to chat conventions (produced consciously or unconsciously).

The learners' level of proficiency in the foreign language influences the frequency of errors. However, a high level of familiarity with computer-mediated communication may lead to an increased number of deviations. This makes error annotation in conceptually oral learner language more difficult, namely the decision whether a deviation is caused by a lack of knowledge (and is potentially correctable) or by the competence in language use (and should not be addressed to).

Because chat conventions may change over time and may be different for different pairs of participants, a further question for research may be an automated recognition of the chat conventions and their incremental adaptation.

As argued in this article, orthography (or deviations from it) is an interactional resource in chat used by participants to regulate the social closeness. An open question remains, how these observations may be captured in a computational model for an artificial conversation partner or an ACC aiming at long-term interaction with the user (multiple weeks). Because all of the native speakers in the dataset show different behaviour with this regard, orthography as an interactional resource may be also a means for expression of specific characteristics of agent's individual interaction profile (Spranz-Fogasy, 2002; Höhn, 2016).

Because the identification of a potentially correctable error does not necessarily trigger a correction, one way for an ACC to handle uncertainties in error recognition is to decide against an error correction, first of all. Uncertainties in language understanding (caused by learner errors or other issues) can be either handled in the dialogue using repair practices or making use of *contingency* which is present in talk at virtually every point (Schegloff, 1996). Contingency allows to have more than one options for responses after each utterance, which makes dialogue modelling difficult but allows to introduce "back doors" in dialogue (types of turns that are valid next turns after the turn where an uncertainty with language understanding occurred).

References

Andrea Abel, Aivars Glaznieks, Lionel Nicolas, and Egon Stemle. 2014. KoKo: an L1 Learner Corpus for German. In *Proceedings of the 9th International Conference on Language Resources and Evaluation (LREC 2014)*, pages 2414–2421.

Luiz A. Amaral and W. Detmar Meurers. 2009. Little things with big effects: On the identification and interpretation of tokens for error diagnosis in icall. *CALICO Journal*, 26(3):580–591.

Thomas Bartz, Michael Beißwenger, and Angelika Storrer. 2014. Optimierung des Stuttgart-Tübingen-Tagset für die linguistische Annotation von Korpora zur internetbasierten Kommu-

nikation: Phänomene, Herausforderungen, Erweiterungsvorschläge. *Zeitschrift für germanistische Linguistik*, 28(1):157–198.

Markus Becker, Andrew Bredenkamp, Berthold Crysmann, and Judith Klein. 2003. Annotation of error types for German Newsgroup Corpus. In *Treebanks*, pages 89–100. Springer.

Adriane Boyd, Jirka Hana, Lionel Nicolas, Detmar Meurers, Katrin Wisniewski, Andrea Abel, Karin Schöne, Barbora Stindlová, and Chiara Vettori. 2014. The MERLIN corpus: Learner language and the CEFR. In *LREC*, pages 1281–1288.

Adriane Boyd. 2010. EAGLE: an Error-Annotated Corpus of Beginning Learner German. In *Proc. of LREC*. ELRA.

Margit Breckle and Heike Zinsmeister. 2010. Zur lernersprachlichen generierung referierender ausdrücke in argumentativen texten. *Textmuster: schulisch-universitär-kulturkontrastiv.*, pages 79–101.

Sviatlana Danilava, Stephan Busemann, Christoph Schommer, and Gudrun Ziegler. 2013. Towards Computational Models for a Long-term Interaction with an Artificial Conversational Companion. In *Proc. of ICAART'13*.

Ana Díaz-Negrillo and Jesús Fernández Domínguez. 2006. Error tagging systems for learner corpora. *Revista española de lingüística aplicada*, 19:83–102.

Ana Díaz-Negrillo, Detmar Meurers, Salvador Valera, and Holger Wunsch. 2010. Towards interlanguage POS annotation for effective learner corpora in SLA and FLT. In *Language Forum*, volume 36, pages 139–154.

Markus Dickinson and Marwa Ragheb. 2015. On grammaticality in the syntactic annotation of learner language. In *The 9th Linguistic Annotation Workshop held in conjuncion with NAACL 2015*, page 158.

Christine Fredriksson. 2012. About collaboration, interaction, and the negotiation of meaning in synchronous written chats in l2-german. In Linda Bradley and Sylvie Thouësny, editors, *CALL: Using, Learning, Knowing, EUROCALL Conference, Gothenburg, Sweden, 22-25 August 2012, Proceedings*, pages 88–92. Research-publishing.net.

Ulrike Gut. 2009. *Non-native speech: A corpus-based analysis of phonological and phonetic properties of L2 English and German*, volume 9 of *English Corpus Linguistics*. Peter Lang.

Sviatlana Höhn. 2015. deL1L2IM: Corpus of long-term instant messaging NS-NNS conversations. ELRA http://islrn.org/resources/339-799-085-669-8/.

Sviatlana Höhn. 2016. *Data-driven repair models for text chat with language learners*. Ph.D. thesis, University of Luxembourg.

Yuri Hosoda. 2006. Repair and relevance of differential language expertise in second language conversations. *Applied Linguistics*, 27(1):25–50.

Gabrielle Kasper. 2004. Participant Orientations in German Conversation-for-Learning. *The Modern Language Journal*, 88:551–567.

Peter Koch. 1994. Schriftlichkeit und sprache. In *Schrift und Schriftlichkeit. Ein interdisziplinäres Handbuch internationaler Forschung*, pages 587–604. Walter de Gruyter.

Cédric Krummes and Astrid Ensslin. 2014. What's Hard in German? WHiG: a British learner corpus of German. *Corpora*, 9(2):191–205.

Sun-Hee Lee, Markus Dickinson, and Ross Israel. 2012. Developing learner corpus annotation for korean particle errors. In *Proceedings of the Sixth Linguistic Annotation Workshop*, pages 129–133. Association for Computational Linguistics.

Shawn Loewen and Sophie Reissner. 2009. A comparison of incidental focus on form in the second language classroom and chatroom. *Computer Assisted Language Learning*, 22(2):101–114.

Anke Lüdeling, Maik Walter, Emil Kroymann, and Peter Adolphs. 2005. Multi-level error annotation in learner corpora. *Proceedings of corpus linguistics 2005*, pages 15–17.

Anke Lüdeling, Elena Briskina, Julia Hantschel, Jenny Krüger, Stéphanie Sigrist, and Ulrike Spieler, 2010. *LeKo Lernerkorpus Handbuch*. Humboldt-Universität zu Berlin, Institut für Deutsche Sprache und Linguistik, Philosophische Fakultät II.

Ursula Maden-Weinberger. 2015. "Hätte, wäre, wenn...": a pseudo-longitudinal study of subjunctives in the corpus of learner german (CLEG). *International Journal of Learner Corpus Research*, 1(1):25–57.

Numa Markee. 2000. *Conversation Analysis*. Mahwah, N.J.: Lawrence Erlbaum.

Gabriela Marques-Schäfer. 2013. *Deutsch lernen online. Eine Analyse interkultureller Aktionen im Chat*. Gunter Narr Verlag.

Detmar Meurers. 2009. On the automatic analysis of learner language: Introduction to the special issue. *CALICO Journal*, 26(3):469–473.

Jürgen Quetz. 2001. Der gemeinsame europäische referenzrahmen. *Info DaF*, 28(6):553–563.

Marc Reznicek, Anke Lüdeling, Cedric Krummes, Franziska Schwantuschke, Maik Walter, Karin Schmidt, Hagen Hirschmann, and Torsten Andreas, 2012. *Das Falko-Handbuch. Korpusaufbau und Annotationen*. Humboldt Universität zu Berlin, 2.01 edition.

Marc Reznicek, Anke Lüdeling, and Hagen Hirschmann. 2013. Competing target hypotheses in the falko corpus: A flexible multi-layer corpus architecture. In Nicolas Ballier Díaz-Negrillo, Ana and Paul Thompson, editors, *Automatic Treatment and Analysis of Learner Corpus Data*, pages 101–123. Amsterdam: John Benjamins.

Emanuel A Schegloff. 1988. Presequences and indirection: Applying speech act theory to ordinary conversation. *Journal of Pragmatics*, 12(1):55–62.

Emanuel A Schegloff. 1996. Issues of relevance for discourse analysis: Contingency in action, interaction and co-participant context. In *Computational and conversational discourse: Burning Issues – An Interdisciplinary Account*, pages 3–35. Springer-Verlag Berlin Heidelberg.

David Schlangen. 2004. Causes and Strategies for Requesting Clarification in Dialogue. In *5th Workshop of the ACL SIG on Discourse and Dialogue*.

Thomas Spranz-Fogasy. 2002. *Interaktionsprofile: Die Herausbildung individueller Handlungstypik in Gesprächen.* Radolfzell: Verlag für Gesprächsforschung.

Tanya Stivers and Nick J Enfield. 2010. A coding scheme for question–response sequences in conversation. *Journal of Pragmatics*, 42(10):2620–2626.

Joel R Tetreault and Martin Chodorow. 2008. Native judgments of non-native usage: Experiments in preposition error detection. In *Proceedings of the Workshop on Human Judgements in Computational Linguistics*, pages 24–32. Association for Computational Linguistics.

Vincenza Tudini. 2010. *Online Second Language Acquisition: Conversation Analysis of Online Chat*. Continuum.

Heike Zinsmeister and Margit Breckle. 2012. The alesko learner corpus: design–annotation–quantitative analyses. *Multilingual Corpora and Multilingual Corpus Analysis. Amsterdam: John Benjamins*, pages 71–96.

On "Article Omission" in German and the "Uniform Information Density Hypothesis"

Eva Horch
Universität des Saarlandes
e.horch@mx.uni-saarland.de

Ingo Reich
Universität des Saarlandes
i.reich@mx.uni-saarland.de

Abstract

This paper investigates whether Information Theory (IT) in the tradition of Shannon (1948) and in particular the "Uniform Information Density Hypothesis" (UID, see Jäger 2010) might contribute to our understanding of a phenomenon called "article omission" (AO) in the literature. To this effect, we trained language models on a corpus of 17 different text types (from prototypically written text types like legal texts to prototypically spoken text types like dialogue) with about 2.000 sentences each and compared the density profiles of minimal pairs. Our results suggest, firstly, that an overtly realized article significantly reduces the surprisal on the following head noun (as was to be expected). It also shows, however, that omitting the article results in a non-uniform distribution (thus contradicting the UID). Since empirically AO seems not to depend on specific lexical items, we also trained our language models on a more abstract level (part of speech). With respect to this level of analysis we were able to show that, again, an overtly realized article significantly reduces the surprisal on the following head noun, but at the same time AO results in a *more* uniform distribution of information. In the case of AO the UID thus seems to operate on the level of POS rather than on the lexical level.

1 Introduction

It is well-known (see e.g. Sandig 1971; Stowell 1991; Reich, to appear) that headlines (and some related text types) in principle allow for articleless singular noun phrases (1a) which are strictly ungrammatical in other contexts (1b):

(1) a. Größte Dürre seit einem halben
Biggest aridity since a half
Jahrhundert
century
"Biggest aridity since half a century"
(zeit.de: 10.08.2015)

b. *Er dachte an größte Dürre seit
He thought of biggest aridity since
einem halben Jahrhundert
a half century
"He thought of biggest aridity since half a century"

This phenomenon is called article omission (AO) in the literature (even though it is not clear that there is in fact some kind of ellipsis involved). What we do *not* want to claim in this paper is that information theory (IT) can explain why AO is grammatical in some text types, but not in others. However, in text types which do allow for AO, AO is clearly optional. In other words, in production the speaker / writer needs to make a choice. The crucial question that we want to investigate in this paper thus is whether this choice in production is guided by information theoretic principles like the Uniform Information Density Hypothesis (UID).

2 Background and Aim

In a paper on complementizer deletion, Jaeger (2010) showed that the overt realization of a complementizer like "that" can significantly reduce the information carried by the (following) subject, thus contributing to a more uniform distribution of the information at the left periphery in the case of high surprisal subjects. According to Jaeger (2010) this effect guides the speaker when choosing between two grammatical alternatives. The underlying principle he states as follows:

(2) **Uniform Information Density (UID)**
Within the bounds defined by grammar, speakers prefer utterances that distribute information uniformly across the signal (in-

> formation density). Where speakers have a choice between several variants to encode their message, they prefer the variant with more uniform information density (ceteris paribus). (Jaeger 2010: 24)

The parallels to AO are rather straightforward: In both cases there are two grammatical alternatives which convey essentially the same proposition. In both cases a functional expression precedes a noun (phrase). In both cases the speaker / writer has to opt for one of the alternatives during the production process. Now, building on Jaeger's (2010) results one might suppose, firstly, that functional expressions in general lower the surprisal of the lexical items to follow, and, secondly, that the realization of the functional expression depends (at least to some degree) on whether its realization results in a more uniform (local) density profile.[1]

3 Language Modeling

To test this hypothesis with respect to AO in German we trained trigram language models with the SRI Language Modeling Toolkit[2] (Stolcke 2002) on a corpus consisting of 17 different text types with about 2.000 sentences each and compared the density profiles of minimal pairs like *Kampf der Zeiten* ("Battle of times") vs. *Der Kampf der Zeiten* ("The battle of times"), see figure 1.

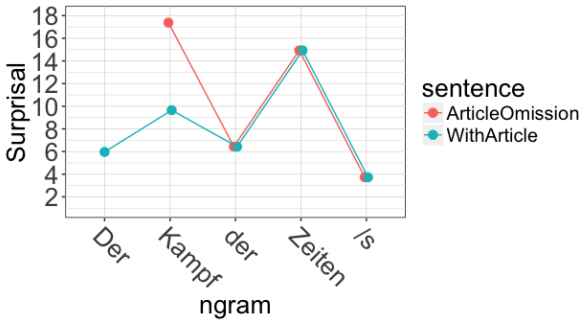

Figure 1: Surprisal profiles (lexical)

Our results show that an article, be it definite or indefinite, does in fact lower the surprisal of the following head noun. This generalizes to sentence-internal positions and to other lexical items of the same syntactic category. In the example chosen, however, the results seem to contradict the UID hypothesis: The original corpus version (*Kampf der Zeiten*) with AO shows a (locally) less uniform profile than the constructed example which overtly realizes the article preceding the head noun.

To get a clearer picture, we abstracted away from the concrete lexical items and trained our language models exclusively on POS structures.[3] The results (trigrams) are shown in figure 2.

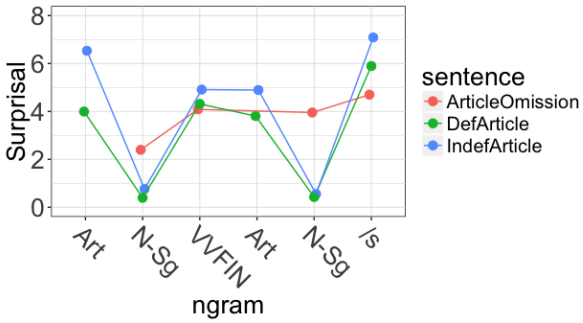

Figure 2: Surprisal profiles (POS)

On this more abstract level, an overtly realized article, whether definite or indefinite, also lowers the surprisal of the following head noun. In contrast to the lexical level, however, an overtly realized article correlates with high surprisal (whether definite or indefinite, whether in sentence-initial or sentence-internal position). As a consequence, the overt realization of an article preceding a head noun results in a peak followed by a trough. Dropping the article, on the other hand, results in a (more) uniform distribution of the information. These results have been confirmed by a 'hybrid' model that combines POS information with information about case, gender and prepositions, see figure 3.

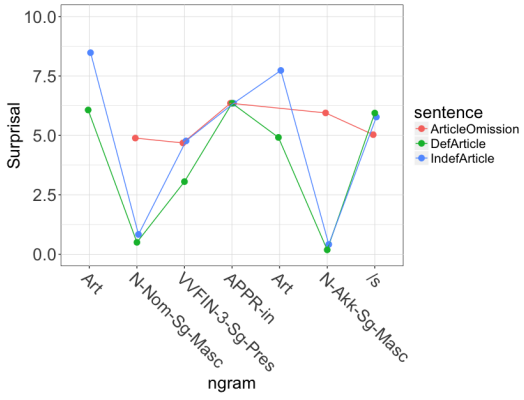

Figure 3: Surprisal profiles (hybrid)

[1] See also De Lange (2008) for a (contrastive) analysis of AO whithin the framework of Information Theory (exclusively based on the number of possible articles in a language).

[2] See `http://www.speech.sri.com/projects/srilm/`. Since smoothing techniques showed no significant effects, we refer to unsmoothed data in this paper.

[3] We used "TreeTagger" by H. Schmid (U Stuttgart) and an expansion of the STTS tagset (see Schmid 1995, 1994).

4 Interpretation

The data suggests two conclusions: Firstly, it seems in fact possible to generalize the observation that functional elements like complementizers and articles systematically lower the surprisal of the lexical item to follow. Secondly, with respect to article omission (and in contrast to complementizer deletion) the UID seems to operate on a more abstract level (POS structure) than on the level of concrete lexical items. This is an important insight into the way article omission (in German) works, and it shows that information theory can in fact contribute to an understanding of that phenomenon.

5 Further Predictions

Given that the two interpretations stated above are essentially on the right track, information theory makes another testable prediction: We expect that if articles are omitted in a sentence, they are in fact omitted across the board. (This is simply because on the level of POS – which has been argued above to be the relevant level for AO – the different surprisal values of different lexical items do not play any role anymore with respect to considerations of uniform information density.) Our corpus suggests that this prediction is in fact correct: The corpus contains a total of 2.127 headlines out of which 308 headlines are in fact subject to AO. Out of those 308 headlines only 137 contain more than one possible target for AO. Out of those 137 headlines, finally, 125 headlines show AO across the board, see (3) (source: SZ.de, 07.06.12) and (4) (source: Bild.de, 04.06.12) for illustration. This is about 91% of the relevant cases, see also figure 4.

(3) Δ *Betrunkene Großmutter schlägt* Δ *Passagier nieder* ('drunken grandma knocks down passenger')

(4) Δ *Fahrer rettet* Δ *Fahrgast aus* Δ *brennendem Bus* ('driver rescues passenger out of burning bus')

As for the remaining 9% it is remarkable that none of them shows the pattern 'overt article followed by null article'. In all of the relevant cases overt articles follow AO. This is in accordance with an observation in Stowell (1991), dubbed "Stowell's Law" in Reich (to appear): In headlines, overt articles must not c-command omitted articles.

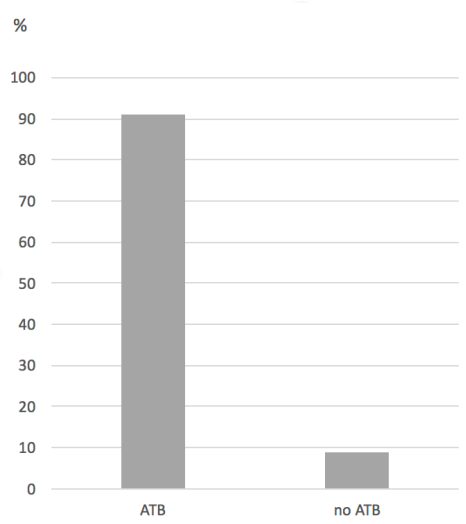

Figure 4: Multiple targets for AO

References

T. Florian Jaeger. 2010. Redundancy and reduction: Speakers manage syntactic information density. *Cognitive Psychology*, 61:23–62.

J. N. De Lange. 2008. *Article omission in child speech and headlines: a processing account*. Ph.D. thesis, Utrecht University, Utrecht.

Ingo Reich. to appear. On the omission of articles and copulae in German newspaper headlines. In D. Massam and T. Stowell, editors, *Register Variation and Syntactic Theory*. Special issue of *Linguistic Variation*.

Barbara Sandig. 1971. Syntaktische Typologie der Schlagzeile. In *Linguistische Reihe*, volume 6. Hueber Verlag, Ismaning.

Helmut Schmid. 1995. Improvements in part-of-speech tagging with an application to German. In *Proceedings of the ACL SIGDAT-Workshop*, Dublin, Ireland.

Claude Shannon. 1948. A mathematical theory of communications. *Bell Systems Technical Journal*, 27(4):623–656.

Andreas Stolcke. 2002. SRILM - an extensible language modeling toolkit. In *Proc. Intl. Conf. Spoken Language Processing*, Denver, Colorado.

Timothy Stowell. 1991. Empty heads in abbreviated English. In *Proceedings of GLOW 1991*.

Automatic cognate classification with a Support Vector Machine

Gerhard Jäger
Tübingen University
Institute of Linguistics
Wilhelmstr. 19
72074 Tübingen, Germany
gerhard.jaeger@uni-tuebingen.de

Pavel Sofroniev
Tübingen University
Institute of Linguistics
Wilhelmstr. 19
72074 Tübingen, Germany
pavel.sofroniev@uni-tuebingen.de

Abstract

Most current approaches in computational phylogenetic linguistics require as input multilingual word lists that are categorized into *cognate classes*. Cognate classification is currently usually done manually by experts, which is time consuming and so far only available for a small number of well-studied language families. Automatizing this step will greatly expand the empirical scope of phylogenetic methods in linguistics, as raw word lists (in phonetic transcription) are much easier to obtain than cognate-coded ones, especially for under-studied language families.

Here we propose a method for automatic cognate classification using supervised learning with a Support Vector Machine. The method outperforms Johann-Mattis List's SCA and LexStat methods (List, 2012; List, 2014b), the current *de facto* standard.

1 Introduction

Computational phylogenetic linguistics has made great strides in recent years. Exciting progress has been made with regard to automated language classification (Bowern and Atkinson, 2012; Jäger, 2015), inference regarding the time depth and geographic location of ancestral language stages (Bouckaert et al., 2012), the identification of sound shifts and the reconstruction of ancestral word forms (Bouchard-Côté et al., 2013; Hruschka et al., 2015), to mention just a few.

Most of the mentioned and related work, especially if Bayesian inference is deployed, relies on multilingual word lists that are manually annotated for cognacy (Bouchard-Côté et al., 2013, being a notable exception). Manual cognate classification is a slow and labor intensive task requiring expertise in historical linguistics and intimate knowledge of the language family under investigation. Also, building automated phylogenetic inference on expert judgments is methodologically problematic as the expert annotators necessarily base their judgments on certain hypotheses regarding the internal structure of the language family in question. In this way, certain assumptions about what is to be inferred is actually fed into the input to the inference process.

The literature contains a variety of proposals to infer cognate classifications automatically from phonetically or orthographically transcribed word lists (Kondrak, 2002; Ellison, 2007; List, 2012; Bouchard-Côté et al., 2013, *inter alia*). In the present paper we will propose a novel approach based on supervised learning. As baselines for comparison we chose List's (2012; 2014b) SCA and LexStat methods since (a) they have been tested on a variety of typologically different language families and (b) a computational implementation is freely available as part of the LingPy software package (List and Moran, 2013; List et al., 2013).

2 Data

We used data from five different sources:[1]

1. the benchmark data from (List, 2014a) (part of the supplementary material accompanying List 2014),
2. the annotated word lists from (Wichmann and Holman, 2013),
3. the part of the IELex data base (`http://ielex.mpi.nl/`, retrieved on 4-23-2013) that contains IPA transcriptions,
4. the part of the ABVD data base (Greenhill et al., 2008, see `http://language.`

[1] The references give the source from where we accessed the data. See the references for the ultimate sources.

psy.auckland.ac.nz/austronesian/; accessed on 12-2-2015) that contains IPA transcriptions, and

5. the Central Asian data set from (Mennecier et al., 2016).

The data from (Wichmann and Holman, 2013) are transcribed in the format of the Automated Similarity Judgment Program (ASJP; see Brown et al., 2013 for the sound class definitions). All other data are transcribed in IPA. Most datasets cover versions of a Swadesh list (see the Supplementary Material for details).

To illustrate the data format, the entries for the concept *woman* in the dataset GER from (List, 2014a) are shown in Table 1.

doculect	*concept*	*transcription*	*cognate class*
Danish	woman	kvenə	160
Dutch	woman	vrɑuu	158
English	woman	uumən	159
German	woman	frau	158
German	woman	vaip	159
Icelandic	woman	kʰɔːna	160
Norwegian	woman	kuinə	160
Swedish	woman	kviniːa	160

Table 1: Entries for *woman* in GER

Two words belong to the same cognate class if — according to historical linguistics scholarship — they descent from the same ancient proto-form.[2]

We split this collection of data bases into three parts, to be used for training (parameter estimation), validation (model selection) and testing respectively in the following way:

- **Training:** data from (List, 2014a) (except the datasets IEL and PAN, as those overlap with the validation data).
- **Validation:** data from (Wichmann and Holman, 2013).
- **Testing:** data from IELex, ABVD and (Mennecier et al., 2016).

This decision is partially motivated from practical consideration. As mentioned above, List's (2012) methods SCA and LexStat will be used as benchmark. As these methods have been developed with the data from (List, 2014a), an informative comparison should be based on the same training data. Furthermore, the data from (Wichmann and Holman, 2013) are only available in ASJP transcription. Our method uses this transcription (all IPA transcriptions are converted into ASJP format by our method), while SCA and LexStat use IPA as input. Therefore the data in ASJP format were used for model selection and the new data in IPA format were held back for final testing.

By way of a further practical consideration, LexStat, in its current implementation from LingPy, can only be applied to datasets comprising at most 169 doculects. The ABVD data comprise 395 doculects. To facilitate the comparison between methods, we split the ABVD data into four equally sized subsets.

3 Methods

To automatically infer cognate classes, we proceed in two steps:

- For each pair of words from the same dataset with the same meaning, the goldstandard data provide a value 0 (different cognate classes) or 1 (same cognate class). We train a binary classifier which predicts probabilities of binary class membership for each such word pair. To this end, we compute a vector of seven quantitative predictors (to be described below).
- For each group of words from the same database denoting the same concept, these pairwise probabilities are transformed into distances. The latter are used as input for hierarchical clustering, leading to an inferred cognate classification.

3.1 PMI similarity

In a first step, all IPA transcriptions are converted into ASJP using the converter from LingPy.

All further steps are based on the *pointwise mutual information* (PMI) between pairs of strings, using the PMI scores and gap penalties from the Supplementary Information of (Jäger, 2015); see (Jäger, 2013) for a detailed description on how those parameters are trained. PMI scores were computed as global pairwise alignment scores as implemented in the function

[2] This criterion is not always clear-cut, even if the etymology of the words involved is known. For instance, English 'woman' descends (according to the Oxford English Dictionary) from Old English 'wife+man'. Only the first of the two components is genuinely cognate with German 'Weib', so the cognacy is only partial.

`pairwise2.align.globalds` of the *Biopython* library (Cock et al., 2009).[3]

The training procedure for PMI scores between different sound classes described in Jäger (2013) ensures that pairs of different sounds frequently participating in regular sound changes have high scores. Therefore cognate word pairs tend to have high PMI similarity even if they are separated by sound changes. An example illustrating this, taken from Jäger (2015), would be the comparison of German *Hand* (`[hant]` in ASJP transcription) to its cognate, English *hand* `[hEnd]` vs. to a non-cognate such as Spanish *mano* `[mano]`. While PMI(`hant, hEnd`)= 4.80 since mismatches such as `a/E` and `t/d` are not very severe, PMI(`hant, mano`)= −11.28 since mismatches such as `h/m` and `t/o` are strongly penalized.

One reviewer suggested to use *longest common subsequence ratio* (LCSR), cf. (Melamed, 1995), or *minimum edit distance* (MED) as basic string similarity measure instead of PMI. These measure are ill-suited for cognate detection though as they both treat all non-identical sound pairs alike. To stay with the example, LCSR(`hant`, `hEnd`) = LCSR(`hant`, `mano`) = 0.5, and MED(`hant`, `hEnd`) = MED(`hant`, `mano`) = 2. On a more general level, the *point-biserial correlation coefficient*[4] between PMI similarity and cognacy is 0.66 for our training data, while it is only 0.58 for MED and 0.57 for LCSR. We therefore conclude that PMI similarity is a good starting point for automatic cognate identification.

Another reviewer remarked that using the same PMI parameters for all comparisons regardless of the languages involved might be sub-optimal as this does not take language-specific regular sound correspondences into account. The benchmark method LexStat does exactly that. As will be shown below, our approach still yields somewhat better results than LexStat. A thorough discussion of this important issue will have to wait for another occasion. The main reason for this discrepancy appears to be though that with the available data, language-specific parameters can be trained on 40 – 200 word pairs only, of which only a fraction is cognate and can therefore provide evidence for regular sound correspondences. This leads to a severe problem of data sparseness. The general-purpose PMI scores from (Jäger, 2015), in contradistinction, were trained on more than one million word pairs, so data sparseness is not an issue.

[3] This implements a modification of the Needleman-Wunsch algorithm (Needleman and Wunsch, 1970), disallowing a gap in one string being directly followed by a gap in the other string.

[4] The *point-biserial correlation coefficient* is a measure of the association strength between a continuous and a binary variable. It is mathematically equivalent to the Pearson correlation coefficient if the binary variable is numerically coded as 0/1.

3.2 Predictors

For a given pair of words (more precisely: a pair of strings of ASJP sound classes) w_1, w_2 (from the same dataset), both denoting concept c, from doculects D_1, D_2, the following (dis-)similarity measures are computed:

1. **PMI similarity.**
2. **Calibrated PMI distances.** Following the procedure described in (Jäger, 2013), the PMI similarities between all pairs of non-synomymous words from D_1, D_2 are computed. The calibrated PMI distance between w_1 and w_2 is the relative frequency of such pairs having a higher similarity than w_1/w_2. This measure can be interpreted as the p-value for the null hypothesis that the similarity between w_1 and w_2 is due to chance. (This measure is monotonically decreasing in the previous measure; it is less fine-grained but less susceptible to chance similarities to similar sound inventories.)
3. The negative logarithm of the previous measure.
4. **Doculect similarity.** The mean value of the previous measure, averaged over all synonymous pairs from D_1/D_2. (This is a measure of the degree of relatedness between D_1 and D_2.)
5. The logarithm of the previous measure.
6. **Average word length.** The average length, measured in the number of ASJP symbols, of all words for concept c (from the same dataset). (This is motivated by Pagel et al., 2007, — where it is shown that frequent words are more resistent against lexical replacement than rare words, together with Zipf's 1935 observation that length of words is negatively correlated with their frequency. It is therefore to be expected that stable concepts are, on average, expressed by shorter words than instable ones.)

7. **Correlation between word distance and doculect similarity.** For each pair of words for concept c, the correlation coefficient between their calibrated PMI distance (measure 2) and the similarity between the corresponding doculects (measure 4) is determined. (We expect this measure to be low for concepts susceptible to borrowing or sound symbolism, and to be high for stable concepts.)

Note that the first three measure quantify the (dis-)similarity between the strings w_1/w_2, the fourth and fifth pertain to the degree of relatedness between the doculects D_1/D_2, while the the last two are related to the diachronic stability of concept c.[5]

3.3 Training a binary classifier

We trained a Support Vector Machine on those vectors, using the Training Set for parameter estimation and the Validation Set for model/feature selection. As criterion to be maximized we chose the *Adjusted Rank Index* (Hubert and Arabie, 1985) as applied to the outcome of the clustering step (see below). Training and prediction was carried out using the svm module from the Python package sklearn `http://scikit-learn.org/stable/modules/svm.html`, which is based on the LIBSVM library (Chang and Lin, 2011).

The test score was maximal with a Radial Basis Function kernel, a kernel coefficient $\gamma = 9 \times 10^{-4}$, and a penalty parameter $C = 0.6$ (both parameter were determined using a grid search). Leaving out any of the seven predictors led to decreased performance.

We observed that using the full collection of vectors computed from the training data led to overfitting. Generalization from the training set to the test set was improved when we randomly selected only one word pair for each data set/concept. This means that out of 111,724 word pairs from the training set, we used only 1,750 pairs (1.6%).

After training, the SVM predicts for each input vector both a categorical class label (0 or 1) and a probability distribution over class labels. Predicting class membership probabilities from a trained SVM was carried out using Platt scaling (Platt, 1999) as implemented in `http://scikit-learn.org`. In the sequel we only use the predicted probability for label 0.

[5]The latter two measures are inspired by (Dellert and Buch, 2016).

3.4 Hierarchical clustering

For each collection of words from the same data set and denoting the same concept, the SVM predicts pairwise probabilities $p(\cdot,\cdot)$ of non-cognacy. These were transformed into pairwise distances according to the formula

$$d(w_i, w_j) \doteq \log p(w_i, w_j) - (\min_{j,k} p(w_j, w_k))$$

UPGMA clustering was performed on these distance matrices. The threshold for forming flat clusters from the UPGMA dendrogram was set at $\log 0.5 - \min_{j,k} p(w_j, w_k)$, i.e., at the distance corresponding to a 50% probability of cognacy.

4 Evaluation

We used two evaluation measures to determine how well an automatically inferred classification confirms to the goldstandard classification: (1) the Adjusted Rand Index (ARI), and (2) the B-Cubed score (Bagga and Baldwin, 1998).

As mentioned above, the performance of our method is compared to List's (2012; 2014b) automatic cognate classification algorithms SCA and LexStat. Perhaps the most significant difference between SCA and LexStat is that the latter automatically detects regular sound correspondences between doculects and utilizes this information to infer cognacy, while the former works with the general-purpose string similarity measures for each pair of doculects. So LexStat incorporates an important insight of the classical comparative method. Our method is closer to SCA in this respect as it also uses the same general-purpose string similarity measures for all language.

The performance of the three methods on the test set are displayed in Table 2.

We found that our method on average outperforms both LexStat and SCA. It also outperforms them for each individual data set according to both evaluation criteria, with one exception (for the Mennecier et al. data set, LexStat achieves a slightly higher B-Cubed score than our method).

5 Conclusion

In this short paper we demonstrated that a combination of linguistically inspired quantitative predictors, modern machine learning techniques and high-quality goldstandard training data achieves state-of-the-art performance for the recalcitrant but important task of automated cognate classification.

	Adjusted Rand Index			*B-Cubed score*		
data set	SVM	LexStat	SCA	SVM	LexStat	SCA
IELex	0.577	0.561	0.541	0.720	0.704	0.695
Mennecier	0.863	0.854	0.828	0.909	0.911	0.894
ABVD-1	0.497	0.451	0.398	0.660	0.642	0.593
ABVD-2	0.551	0.494	0.435	0.692	0.667	0.609
ABVD-3	0.532	0.462	0.406	0.681	0.649	0.598
ABVD-4	0.514	0.469	0.424	0.669	0.652	0.608
weighted mean	0.583	0.542	0.498	0.718	0.700	0.661

Table 2: Evaluation results. "SVM" refers to the method described here

These results are mostly to be understood as a proof of concept. For instance, the idea — implemented in LexStat — to utilize recurring sound correspondences for cognate identification is undoubtedly highly productive. In future research it will be explored whether more and better predictors can be inferred based on this insight.

Acknowledgments

We thank the anonymous reviewers for KONVENS for helpful feedback and the authors of (Greenhill et al., 2008; Mennecier et al., 2016) for the kind permission to use their data. This work has been supported by the ERC Advanded Grant 324246 EVOLAEMP, which is gratefully acknowledged.

References

Amit Bagga and Breck Baldwin. 1998. Entity-based cross-document coreferencing using the vector space model. In *Proceedings of the 17th International Conference on Computational Linguistics-Volume 1*, pages 79–85. Association for Computational Linguistics.

Alexandre Bouchard-Côté, David Hall, Thomas L. Griffiths, and Dan Klein. 2013. Automated reconstruction of ancient languages using probabilistic models of sound change. *Proceedings of the National Academy of Sciences*, 36(2):141–150.

Remco Bouckaert, Philippe Lemey, Michael Dunn, Simon J. Greenhill, Alexander V. Alekseyenko, Alexei J. Drummond, Russell D. Gray, Marc A. Suchard, and Quentin D. Atkinson. 2012. Mapping the origins and expansion of the Indo-European language family. *Science*, 337(6097):957–960.

Claire Bowern and Quentin Atkinson. 2012. Computational phylogenetics and the internal structure of Pama-Nyungan. *Language*, 88(4):817–845.

Cecil H. Brown, Eric Holman, and Søren Wichmann. 2013. Sound correspondences in the world's languages. *Language*, 89(1):4–29.

Chih-Chung Chang and Chih-Jen Lin. 2011. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

Peter J. A. Cock, Tiago Antao, Jeffrey T. Chang, Brad A. Chapman, Cymon J. Cox, Andrew Dalke, Iddo Friedberg, Thomas Hamelryck, Frank Kauff, Bartek Wilczynski, and Michiel J. L. de Hoon. 2009. Biopython: freely available Python tools for computational molecular biology and bioinformatics. *Bioinformatics*, 25(11):1422–1423. doi:10.1093/bioinformatics/btp163.

Johannes Dellert and Armin Buch. 2016. Using computational criteria to extract large Swadesh lists for lexicostatistics. ms., Tübingen.

T. Mark Ellison. 2007. Bayesian identification of cognates and correspondences. In *Proceedings of Ninth Meeting of the ACL Special Interest Group in Computational Morphology and Phonology*, pages 15–22. Association for Computational Linguistics.

Simon J. Greenhill, Robert Blust, and Russell D. Gray. 2008. The Austronesian Basic Vocabulary Database: From bioinformatics to lexomics. *Evolutionary Bioinformatics*, 4:271–283.

Daniel J. Hruschka, Simon Branford, Eric D. Smitch, Jon Wilkins, Andrew Meade, Mark Pagel, and Tanmoy Bhattachary. 2015. Detecting regular sound changes in linguistics as events of concerted evolution. *Current Biology*, 25(1):1–9.

Lawrence Hubert and Phipps Arabie. 1985. Comparing partitions. *Journal of Classification*, 2(1):193–218.

Gerhard Jäger. 2013. Phylogenetic inference from word lists using weighted alignment with empirically determined weights. *Language Dynamics and Change*, 3(2):245–291.

Gerhard Jäger. 2015. Support for linguistic macrofamilies from weighted sequence alignment. *Proceedings of the National Academy of Sciences*, 112(41):12752–12757. doi: 10.1073/pnas.1500331112.

Grzegorz Kondrak. 2002. *Algorithms for Language Reconstruction*. Ph.D. thesis, University of Toronto.

Johann-Mattis List and Steven Moran. 2013. An open source toolkit for quantitative historical linguistics. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, Sofia, Bulgaria, August 4-9.

Johann-Mattis List, Steven Moran, Peter Bouda, and Johannes Dellert. 2013. Lingpy. Python library for automatic tasks in historical linguistics. URL: http://www.lingpy.org. Version 2.2 (Uploaded on 2013-11-22).

Johann-Mattis List. 2012. Lexstat: Automatic detection of cognates in multilingual wordlists. In Miriam Butt and Jelena Prokić, editors, *Proceedings of LINGVIS & UNCLH, Workshop at EACL 2012*, pages 117–125, Avignon.

Johann-Mattis List. 2014a. Data from: Sequence comparison in historical linguistics. GitHub Repository. Release 1.0.

Johann-Mattis List. 2014b. *Sequence Comparison in Historical Linguistics*. Düsseldorf University Press, Düsseldorf.

I. Dan Melamed. 1995. Automatic evaluation and uniform filter cascades for inducing n-best translation lexicons. In *Proceedings of the 3rd Workshop on Very Large Corpora*, pages 184–198, Cambridge, MA.

Philippe Mennecier, John Nerbonne, Evelyne Heyer, and Franz Manni. 2016. A Central Asian language survey: Collecting data, measuring relatedness and detecting loans. *Language Dynamics and Change*, 6(1). in press.

Saul B. Needleman and Christian D. Wunsch. 1970. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *Journal of Molecular Biology*, 48:443–453.

Mark Pagel, Quentin D. Atkinson, and Andrew Meade. 2007. Frequency of word-use predicts rates of lexical evolution throughout indo-european history. *Nature*, 449(7163):717–720.

John C. Platt. 1999. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In *Advances in large margin classifiers*, pages 61–74. MIT Press.

Søren Wichmann and Eric W. Holman. 2013. Languages with longer words have more lexical change. In Lars Borin and Anju Saxena, editors, *Approaches to Measuring Linguistic Differences*, pages 249–284. Mouton de Gruyter, Berlin.

G. Zipf. 1935. *The Psycho-Biology of Language*. MIT Press, Cambridge, Massachusetts.

A Supplemental Material: Data used

dataset	doculects	# doculects	# words	# concepts	# cognate classes	transcription
BAI	Bai dialects	9	1,028	101	205	IPA
GER	Germanic languages and dialects	7	814	110	200	IPA
IDS	Romance and Germanic languages	4	2,429	550	1,602	IPA
JAP	Japanese dialects	10	1,986	200	460	IPA
KSL	various languages (partially unrelated)	7	1,400	200	1,208	IPA
OUG	Uralic languages	21	2,055	110	242	IPA
PIE	Indo-European languages	19	2,172	110	634	IPA
ROM	Romance languages	5	589	110	178	IPA
SIN	Chinese dialects	15	2,789	140	1,025	IPA
SLV	Slavic languages	4	454	110	165	IPA
total		101	15,716	1,750	5,919	

Table 3: Data from (List, 2014a), used for training

dataset	# doculects	# words	# concepts	# cognate classes	transcription
Afrasian	21	829	40	380	ASJP
Huon	14	1,171	84	536	ASJP
Kadai	12	460	40	129	ASJP
Kamasau	8	271	36	60	ASJP
Lolo-Burmese	15	574	40	105	ASJP
Mayan	30	2,896	100	858	ASJP
Miao-Yao	6	223	39	74	ASJP
Mixe-Zoque	10	961	100	300	ASJP
Mon-Khmer	16	1,487	100	775	ASJP
Moroboe	55	2,040	138	582	ASJP
total	187	10,912	617	3,799	

Table 4: Data from (Wichmann and Holman, 2013), used for validation

dataset	doculects	# doculects	# words	# concepts	# cognate classes	transcription
ABVD-1	Austronesian	99	14,198	210	4,592	IPA
ABVD-2	Austronesian	99	14,243	210	4,156	IPA
ABVD-3	Austronesian	99	13,878	210	4,181	IPA
ABVD-4	Austronesian	98	14,155	210	4,435	IPA
IELex	Indo-European	55	8,313	207	1,998	IPA
Mennecier	Central Asian	88	15,904	183	895	IPA
total		138	77,523	1,230	19,707	

Table 5: Datasets used for testing

Parsing Free-Form Language Learner Data: Current State and Error Analysis

Christine Köhn and **Tobias Staron** and **Arne Köhn**
Department of Informatics
Universität Hamburg
{ckoehn,staron,koehn}@informatik.uni-hamburg.de

Abstract

Parsing learner data with high accuracy is important for all systems that want to analyze language learner input, such as computer-assisted language learning software. State-of-the-art parsers are typically trained on news text and not on language learner data since this kind of data is often not available in sufficient quantities. Our contribution is three-fold: We provide gold-standard syntactic annotations for sentences from language learners of German, evaluate the performance of state-of-the-art parser pipelines on this corpus and explore whether augmentation of a parser with weighted constraints to avoid common structural errors could lead to improvements.

1 Introduction

Syntax parsers are usually based on the assumption that the input is well-formed. Their statistical models are trained on large corpora, which are mostly annotated news text and therefore also comparatively well-formed. On the other hand, language learner (L2) data from people, who are learning a language that is not their native one, inherently contains malformed parts. This mismatch between well-formed training data and malformed run-time input may lead to a degradation in parser performance.

Training a parser directly on L2 data is not feasible for several reasons. First and foremost, there is too little annotated L2 data available for most languages. Also, training on L2 data would presuppose that all learners of the respective language make comparable mistakes, which is unlikely.

Extracting the syntactic structure of an L2 sentence is important for different applications, e. g. for assessing answers to reading comprehension questions or for deriving error diagnoses.

We are especially interested in free-form text where the sentence structure is not externally influenced, e. g. the text is not an answer to a question. For this type of input, it is especially hard to extract error diagnoses without syntactic analyses, since the contents of the L2 sentences are not known beforehand.

For the parser evaluation, we annotated 100 L2 German sentences (Falko-100dep) from a subcorpus of the Falko corpus (Reznicek et al., 2012) with dependency trees. The Falko corpus contains target hypotheses, i. e. manually corrected versions of the texts, which can be better automatically analyzed than the original learner sentences (Rehbein et al., 2012). Nevertheless, we evaluate the parsers on the original sentences since we want to assess their performance in a setting where manually created target hypotheses are not available.

2 Related Work

There is already a dependency annotated corpus of L2 German, which uses the same annotation standard as the Falko-100dep corpus: The CREG-109 corpus (Ott and Ziai, 2010), containing answers to reading comprehension questions. We chose to annotate essay texts from the Falko corpus because of their different characteristics. E. g., the sentences in the Falko-100dep corpus are much longer on average (18.9 tokens) than the responses to the respective questions in the CREG-109 corpus (8.3 tokens). Also, the language learner proficiency differs: The CREG-109 sentences were written by learners on the beginning and intermediate level, whereas the Falko-100dep sentences were written by upper intermediate to advanced learners.

Berzak et al. (2016) compiled a corpus of language learner sentences for L2 English, including part-of-speech (PoS) tags and Universal Dependency trees. They annotated the original, ungrammatical sentences as well as their corrected versions. Additionally, they provide a set of annotation

guidelines for syntactically annotating ungrammatical English.

Rehbein et al. (2012) examined the impact of PoS quality on parsing accuracy of language learner sentences. For this purpose, they annotated 100 L2 sentences from the Falko essay subcorpus with constituency structures. They compared the PoS accuracy when tagging the original sentence and a corrected form, the target hypothesis. The target hypotheses were formulated with the purpose of making them suitable for automatic processing. Tagging the target hypothesis of the L2 sentence and projecting the PoS tags back to the original sentence improved the PoS accuracy. Furthermore, they found that the manual correction of automatically assigned PoS tags for some of the taggers does not significantly improve parsing accuracy. Their approach of processing target hypotheses instead of the original sentence is appropriate for analyzing L2 corpora but we perform all experiments on the original learner sentence, simulating a setting where target hypotheses are not available.

Ragheb and Dickinson (2013) developed a multi-layered dependency annotation scheme for learner language and achieved good inter-annotator agreement for L2 English. One dependency layer represents morpho-syntactic information, while the other represents subcategorization information. The PoS tag annotation also consists of two layers: one for morpho-syntactic and one for distributional evidence.

Krivanek and Meurers (2011) compared a transition-based parser, MaltParser (Nivre, 2007), to a rule-based parser, WCDG (Foth and Menzel, 2006), for parsing L2 text by evaluating them on the CREG-109 corpus. They found that, while both parsers have a similar overall accuracy, MaltParser performs better at attaching optional relations, but WCDG is better at identifying the main functor-argument relations.

Hybrid parsing – i. e. incorporating more than one parsing approach – can be performed by using a statistical parser, such as MaltParser, as an additional input source for a rule-based parser, e.g. WCDG (Foth and Menzel, 2006). Khmylko et al. (2009) demonstrated that this approach is beneficial even if the statistical parser is superior to the rule-based one, i. e. the hybrid parser performs better than both its components. Köhn and Menzel (2013) showed that even though a combination of jwcdg, a Java re-implementation of WCDG (Beuck et al., 2013), and MaltParser is beneficial for newspaper text, combining both parsers does not help to improve parsing performance on the CREG-109 corpus.

It is also possible to build a hybrid parser the other way around: Seeker and Kuhn (2013) included morpho-syntactic constraints in statistical parsing to restrict the search space. In addition, morphological disambiguation is performed (which jwcdg also does). In contrast to the previously mentioned approaches, the constraints are not graded. Our approach, described in Section 5, is similar but uses graded constraints and does not perform morphological disambiguation.

Further work has been done on integrating grammars into data-driven parsers. Dhar et al. (2012) used MaltParser and the parses it generates are corrected by grammar rules which, in turn, are inferred from running MaltParser alone and analyzing its errors. This approach was tested for Bangla.

An alternative approach to develop hybrid parsers is to build an ensemble of statistical parsers. The parsers can be combined by n-best parsing and ranking, as performed by Björkelund et al. (2013). This approach yields the currently best results for the shared task on parsing morphologically rich languages.

Even a simple voting by several parsers can outperform the individual parser performances. For example, Sagae and Lavie (2006) combined several shift-reduce parsers similar to MaltParser as well as MST parser (McDonald and Pereira, 2006) by weighted voting for each edge. This approach yielded an increase of 1.7 percentage points in accuracy on the Penn Treebank.

3 The Falko-100dep Corpus

The FalkoEssayL2 corpus contains German essays written by language learners with varying degree of proficiency. Each learner had 90 minutes to write an essay on a given topic without help (neither machine nor human). In addition, each learner completed a C-test to assess their proficiency in German. The C-test scores can be translated into standard CEFR levels[1], which we did to cluster the texts into B2, C1, and C2, with C2 referring to the more advanced learners.

We randomly sampled 100 sentences from the

[1]The **C**ommon **E**uropean **F**ramework of **R**eference for Languages: Learning, Teaching, Assessment

	B2	C1	C2	all
Mean	13.6	20.8	22.6	18.9
Median	11.5	17.0	21.0	17.0

Table 1: Sentence length distribution in the Falko-100dep corpus by language proficiency

FalkoEssayL2 corpus v2.4[2], 33 to 34 for each level (B2, C1, C2) and used the manually corrected tokenization from the level ctok, where the text is otherwise completely untouched. The sentence segmentation was extracted from the level ZH0 (containing already corrected material) and mapped back to ctok, as ctok does not provide sentence segmentation. This way, the sentences we worked on are completely made up of uncorrected tokens, but manually tokenized and segmented.

The essays in the FalkoEssayL2 corpus cover four different topics, which are also represented in Falko-100dep: crime, academic studies, feminism, and wages. The sentence length correlates with language proficiency (see Table 1). The sentences produced by C2 learners are about twice as long as the sentences from B2 learners, suggesting that more experienced learners write more complex sentences.

The sentences contain all kinds of mistakes, e. g. spelling and grammatical mistakes. However, not all sentences contain mistakes. As evidenced by a high inter-annotator agreement (see next section), they do not prevent a reasonable annotation.

3.1 Annotation Process

Ragheb and Dickinson (2011) argue that learner language should be annotated with an annotation scheme specifically tailored to capture the phenomena of the learner's interlanguage and treat it as a system in its own right instead of comparing it to the target language or the learner's L2. We do not use an annotation scheme designed for learner language but use the annotation guidelines by Foth (2006)[3], which were designed for L1 German and are also used by the CREG-109 corpus and the Hamburg Dependency Treebank (HDT) (Foth et al., 2014). We did not use a scheme designed for L2, because this not available for German yet and consequently no parser can produce such an output since no training data is available. We are aware of the fact that an L1 scheme captures less information than the scheme proposed by Ragheb and Dickinson (2012) but we think that the annotations represent the syntactic structure of an L2 sentence well enough to serve as an input for further processing, e. g. error diagnosis.

Our annotation process was as follows: First, three annotators each annotated the first 12 sentences and met afterwards to discuss annotation decisions and to mutually decide on each controversial annotation to gain a higher agreement for the remaining annotation process. To speed up this process, the annotators did not start from scratch but corrected the parses of three different parsers (one parser per annotator: TurboParser, RBGParser, jwcdg - for an overview of these parsers, see Section 4). Two of the annotators continued with a partial overlap (sentences 13 to 22) and decided again on a gold standard annotation afterwards. They annotated sentences 23-100 separately, but again cross-checked the annotations of each other.

For the first 22 sentences, the two annotators achieve an inter-annotator agreement in terms of labeled attachment score (LAS) of 91.48%. The annotators agree on 93.73% of the dependencies and on 95.99% of the labels. This indicates an already high agreement between the two annotators which, in turn, shows that the sentences of our corpus can be annotated fairly well despite the mistakes they contain.

Comparing the annotations of the first 22 sentences with the resulting gold standard leads to a LAS of 94.99% for one annotator (96.99% of the dependencies and 97.49% of the labels remain) and a LAS of 95.49% for the other annotator (96.24% of the dependencies and 97.99% of the labels remain). This shows that only few changes were made when the annotators decided on the gold standard annotation.

Comparing the annotations for the remaining sentences 23 to 100 with the gold standard results in a LAS of 95.29% for the first annotator (agreement on 95.53% of the dependencies and on 97.52% of the labels) and 97.39% for the second one (agreement on 98.46% of the dependencies and on 97.87% of the labels) indicating improved annotations for sentences annotated later, based on comparing and discussing the annotations of the first 22 sentences.

[2]The corpus is available at `www.linguistik.hu-berlin.de/de/institut/professuren/korpuslinguistik/forschung/falko/zugang`

[3]The annotations guidelines are in German. Foth et al. (2014) give an overview of the dependency relations in English.

3.2 Annotation Decisions

The annotators knew what topics were dealt with in the FalkoEssayL2 corpus. The annotation was carried out with respect to an implicit target hypothesis (TH). Our TH is a grammatical correct sentence that makes sense as far as possible while at the same time tries to deviate from the original as little as possible. The TH is different to the ones defined in the Falko manual (Reznicek et al., 2012) since we want to attach as many words as possible to other words. The rules for our TH overlap largely with the rules for the minimal TH in the Falko corpus. The main difference is that we also change the verb if it seems more suitable.

4 Parser Evaluation

We evaluate three parsers: jwcdg (Beuck et al., 2013), TurboParser (Martins et al., 2013), and RBGParser (Zhang et al., 2014). While the first one is rule-based, the others are trained on a treebank. We used the first 100.000 sentences of part A of the HDT to train them.

TurboParser and RBGParser are quite similar in their feature set (at least in our experimental setup). However, their approach to decoding differs fundamentally. the decoding of jwcdg is similar to the one of RBGParser but it uses a hand-written weighted constraint grammar.

Since we want to assess the parses in a setup where gold standard PoS tags and target hypotheses are not available, we use predicted instead of gold standard PoS tags for all parsers. TurboTagger (distributed with TurboParser) assigns the PoS tags for both TurboParser and RBGParser. jwcdg requires multi-tagging and therefore uses TnT (Brants, 2000). TurboTagger was trained on the same data as the data-driven parsers.

4.1 TurboParser

TurboParser translates the parsing problem into a binary integer linear program (ILP) where each possible edge is assigned a variable. The ILP consists of two parts: The linear constraints make sure that each result is a tree while the objective function makes sure that the resulting tree is good. In contrast to the linear constraints, the objective function needs to be learned. To make learning and decoding feasible, the objective function is decomposed into local components (see Martins et al. (2013) for an overview). During decoding, a relaxation of the ILP is solved using dual decomposition.

Because each component of the objective function only scores a fixed set of edges (up to three), global constraints can not be learned. Due to the overlap between the different scoring components (edges of the dependency tree are part of several components), the best scoring tree needs to be locally consistent in each component. However, no component can enforce the existence of a specific construct, e.g. a subject for a verb.

4.2 RBGParser

RBGParser[4] is a data-driven dependency parser (Zhang et al., 2014). It exploits a variety of features: global as well as local features considering up to three connected edges in the dependency structure. Different models based on different subsets of features can be used. In this work, the standard model is used, which uses only local features comparable to the ones of TurboParser.

When using the standard model, RBG applies hill-climbing. It starts with a random parse and reassigns edges until the best parse stops changing. RBG repeats this procedure, each time starting with a newly sampled random parse, until the result converges in order to find a parse as optimal as possible. Because initial random parses for a given sentence are sampled independently from each other, using only first-order features, the scoring of an analysis is largely decoupled from the creation of new parses. Therefore, it is possible to use an arbitrarily complex scoring function. In addition to the TurboParser features, RBG employs a low-rank tensor component which scores single edges (Lei et al., 2014).

4.3 jwcdg

jwcdg (Beuck et al., 2013) implements the weighted constraint dependency grammar formalism (Schröder, 2002). It uses a grammar consisting of weighted constraints, which are used to score analyses, and taboo search (Foth et al., 2000) to find the optimal analysis for a sentence. This approach is comparable to the hill climbing performed by RBG, although the former is more complex.

Additionally, jwcdg is able to evaluate the constraints of its grammar on an already parsed sentence in order to determine constraint violations. Besides generating a score based on the constraint evaluation, the violated constraints can be inspected to analyze the parse.

[4] RBG in the remaining paper

		UAS			
	LAS	all	B2	C1	C2
RBG	80.32	86.70	86.72	84.40	88.79
Turbo	81.83	86.76	85.96	85.23	88.63
jwcdg	77.40	82.02	84.96	79.87	82.18
RBG_h	79.95	86.03	85.96	83.72	88.17

Table 2: Attachment scores for the Falko-100dep corpus (labeled and unlabeled) for RBG, TurboParser, jwcdg and the hybrid parser RBG_h (RBG augmented with constraints); UAS also by learner proficiency.

The grammar was co-developed during the annotation process of the HDT. In contrast to the scoring functions learned by the other parsers, each constraint (and its purpose) can be understood by humans as it is directly linguistically motivated. Since the constraints were created manually, they rely less on the word forms. E. g., differences in distributional attachment preferences for nouns are mostly not modeled. In this paper, we use jwcdg without external predictors, except for a PoS tagger, to assess the quality of the underlying grammar.

In contrast to the previously mentioned parsers, jwcdg co-optimizes dependency structures, dependency labels, and PoS tags and performs lexical disambiguation. jwcdg uses TnT in a multi-tagging mode to obtain weighted suggestions for PoS tags. Due to the lexical disambiguation, the grammar makes extensive use of features such as valence, number, and other morpho-syntactic information.

4.4 Evaluation on Falko-100dep and CREG-109

We performed an evaluation on the 100 syntactically annotated Falko sentences[5]. RBG and TurboParser both produce structures with similar accuracy, but TurboParser is better at assigning dependency labels (see Table 2). jwcdg trails the other two parsers by more than 4 percentage points with respect to the unlabeled attachment score (UAS).

Overall, the performance degrades on L2 text relative to news text. On the HDT, TurboParser achieved an UAS of 93.66% (LAS: 91.35%) and RBG 93.20% (LAS: 90.76%). For both parsers, the attachment errors doubled on the Falko data.

[5]All evaluations exclude punctuation, since punctuation is always attached to 0 with an empty label in the annotation scheme and counting these attachments would only skew the results.

	RBG	Turbo	jwcdg	RBG_h
UAS	90.86	89.83	85.33	89.83
LAS	82.50	80.95	77.86	81.72

Table 3: Unlabeled and labeled attachment scores for RBG, TurboParser, jwcdg and the hybrid parser RBG_h (RBG augmented with constraints) on the CREG-109 corpus.

	LAS	UAS	LA
Falko-100dep	87.36	90.10	93.01
CREG-109	92.79	94.59	95.11
HDT	91.86	93.40	95.90

Table 4: The agreement of RBG and TurboParser on attachment scores (labeled and unlabeled) and label accuracy (LA)

We also evaluated how the language proficiency influences the parsing accuracy. Both RBG and TurboParser achieved the highest accuracy with considerable margin on C2 level data, i. e. data with little grammatical mistakes. In contrast, jwcdg performs best on B2 data, with only a small gap to the other parsers. This indicate a robustness of jwcdg against ill-formed input.

The results on CREG-109 are consistent with our findings on the Falko corpus (see Table 3). Compared to the results reported by Krivanek and Meurers (2011), RBG and TurboParser considerably outperform MaltParser on CREG-109, which is used for the automatically generated syntax layer of the FalkoEssayL2 corpus.

4.5 Analysis

RBG and TurboParser use the same feature sets and have a similar performance on Falko-100dep, CREG-109 and the HDT. However, they commit different errors as can be seen in Table 4. Notably, the difference on Falko-100dep is more pronounced than on the other two corpora.

On Falko-100dep, RBG and TurboParser assign most of the attachments (regent and label) with a similar recall and precision, but there are some major differences. Table 5 shows the attachments where RBG and TurboParser differ most. Moreover, RBG never correctly assigns (regent and label) the infrequent labels `EXPL` (expletive) and `OBJP` (prepositional object), whereas TurboParser at least identifies some of these dependencies cor-

	APP	KOM	OBJD
RBG recall	78.57	64.29	44.44
Turbo recall	71.43	100.00	22.22
RBG precision	91.67	60.00	57.14
Turbo precision	76.92	93.33	40.00

Table 5: Differences in attachments (regent and labels) on Falko-100dep. `APP`: apposition `KOM`: comparison word, `OBJD`: dative object

rectly (28.57% and 47.37% recall, 50% and 81.82% precision). Thus, RBG underrepresents the rare labels in its output.

The tagging error rate of TurboTagger (which is used by both RBG and TurboParser in our experiments) is 5.3%. jwcdg – which co-optimizes the PoS tags – only has an error rate of 4.5%. This difference highlights the benefits of optimizing the PoS tags together with the syntax.

4.6 Relabeling

Not only the syntactic structure but also the dependency labels are important for an analysis of a sentence. TurboParser assigns edge labels before parsing and therefore only uses information from the single edge. In contrast, RBG labels edges after the dependency tree is build, enabling it to use features from the dependency tree. Edge relabeling using information from the dependency tree (e.g. about neighboring edges) has proven to be beneficial for labeling accuracy (Köhn et al., 2014).

We use both the Maximum Entropy relabeler (MELabeler) described in Köhn et al. (2014) as well as TurboDependencyLabeler[6]. The relabelers were trained on part A of the HDT. Interestingly, both labelers actually decrease the labeling accuracy with respect to the original labeling by the parsers, as can be seen in Table 6. Since RBG and the relabelers assign labels after a dependency tree is build, it is not surprising that the labeling accuracy does not improve for RBG. We suspect that the noticeable decrease in labeling accuracy for TurboParser stems from the fact that the overall structure of a sentence is often not well-formed and an ill-formed part can influence more labels if the labeling decision is not made purely local. In a way, the relabelers overfit on well-formed data, whereas the simpler model employed by TurboParser cannot overfit in that way.

[6]which is distributed with TurboParser

	original	TurboLabeler	MELabeler
RBG	87.00	87.00	86.21
Turbo	87.42	85.30	86.03

Table 6: Labeling accuracy (in %) for the relabelers as well as the original labeling of the respective parsers.

5 Constraint-based Augmentation of RBGParser

RBG is able to generate accurate parses. If a sentence contains mistakes, which is the case for sentences acquired from language learners, the accuracy of RBG will decrease though, as we have shown in Section 4. Thus, there is room for improvement regarding the robustness of the parser.

We examined the RBG parses for 30 sentences from the FalkoEssayL2 corpus, that are not part of our Falko-100dep corpus, and for the first 20 sentences of Falko-100dep as follows: First, we evaluated the grammar of jwcdg for German on these parses. Next, we inspected the constraint violations for each word that RBG attached incorrectly. We observed that part of the wrong attachments violate constraints which express essential well-formedness conditions (for a parse) derived from the annotation guidelines. Such constraints e. g. express the requirement that a certain edge label goes together with specific PoS tags.

Because of this observation, we developed the idea to integrate weighted constraints via the scoring function of jwcdg into RBG to obtain parses that adhere to the basic annotation principles. For the remainder of this paper, we call the resulting hybrid parser RBG_h.

RBG is suitable for this approach because of its property that the generation of parses and the computation of their scores is separated and not interwoven as it is the case for TurboPaser.

5.1 Integrating RBGParser and weighted constraints

When generating new parses during hill-climbing, RBG evaluates the parses using its scoring mechanism (see Figure 1). The parses it determines as local maxima are compared to the best global solution up to that point. This is where the grammar integration takes place. When the scoring component is called to evaluate a local maximum parse, jwcdg scores this parse based on a grammar (see

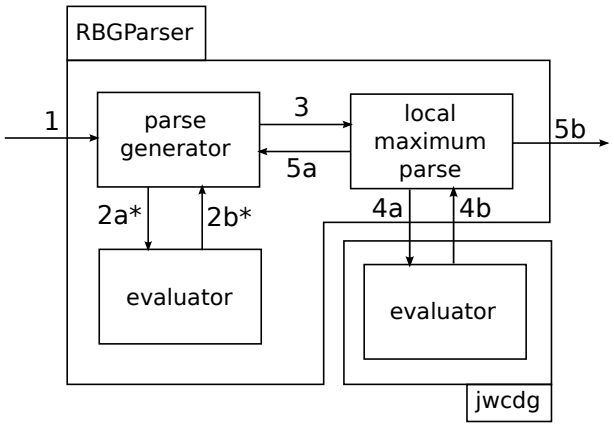

Figure 1: The process of parsing a sentence based on the hybrid approach of augmenting RBG with constraints. A sentence is given to RBG (1) and a parse is generated via hill-climbing performed on an initial random parse, with edges repeatedly passed on to the evaluator of RBG (2a) to receive their scores (2b), resulting in a local maximum parse (3), which is passed on to jwcdg (4a) to evaluate it on the grammar (4b). The RBG score of the local maximum parse is combined with the grammar penalty from jwcdg and is compared to the best parse so far. This procedure is repeated (5a) until the parsing converges (5b).

section 5.2). The score of jwcdg is converted into a penalty, the grammar penalty, which is combined with the rating of RBG into a single score.

In RBG, the syntactic labels are not assigned during decoding. For the hybrid parsing approach, RBG has been modified to assign syntactic labels already to edges of intermediate solutions. Thus, jwcdg has access to those labels in the hybrid setup.

To generate a score, the local maximum parses are passed on to jwcdg. It converts the parses into its internal representation, containing the word forms, their PoS tags and the dependency structure but no further lexical information. Then, jwcdg evaluates the constraints from the grammar and returns a score between 0 and 1, 0 for the violation of constraints that are not allowed to be violated under any circumstances, so-called hard constraints, and 1 if no violations occur at all.

One challenge is to combine the jwcdg and RBG scores since they are from different domains. We combine the two scores as follows: If, according to jwcdg, there are no constraint violations altogether, the RBG score is used without further modification. Otherwise, the jwcdg score is converted into a penalty and subtracted from the score of RBG. The lower the jwcdg score, the higher the respective penalty and vice versa. If a parse violates a hard constraint, the penalty is raised so that it becomes more probable that RBG_h prefers parses not violating any hard constraint over this parse.

5.2 Grammar

To evaluate the constraints RBG is augmented with, jwcdg is being used. Originally, jwcdg has a grammar for German that represents the German language as accurately as possible. It contains 1087 constraints. In this work, a subset of those constraints is used.

The constraints are divided into several groups. All groups for which the constraints are likely to be violated by L2 sentences are excluded. For example, all constraints are excluded from the grammar that are related to the word order or punctuation.

Because of the inherent lexical ambiguity of many word forms, groups of constraints which make use of lexical information were excluded. The reason is that RBG does not provide any form of lexicalization. Thus, jwcdg has to try to find an optimal one every time it receives a parse from RBG. This results in problems regarding the running time due to combinatorial issues. Another reason for omitting lexicalization are possible misspellings leading to wrong lexical information. The constraint groups remaining in the grammar deal with basic structural phenomena and express:

(a) which structure are licensed by the word categories in terms of PoS tags.

(b) which attachments to the root of a parse are allowed.

(c) that the labels of dependents of a word have to be unique for specific dependency labels.

(d) that particular attachments may not cross punctuation marks.

Some of the remaining constraints still depend on lexical information. Since no lexicon was used in the evaluation, those parts of the constraints would have evaluated to false, although no proposition could have actually been made. Therefore, those parts were relaxed so far that they do not influence the evaluation. If this was not possible, the respective constraint was removed from the grammar.

Also, only constraints were used whose violations mark severe mistakes. Less severe violations (which only encode preferences) were disregarded, resulting in a subset of 205 constraints, approximately a fifth of the original grammar. We call the resulting grammar minimal grammar for the remainder of this paper.

5.3 Error Analysis

As can bee seen in Table 2, the grammar integration has a negative effect on parsing performance. We compared the unlabeled attachments of the RBG and the RBG_h parses to find out why RBG_h performs worse. They differ in five parses. None of these five RBG_h parses have a higher UAS than their corresponding RBG parse, four parses have a lower one and one parse has the same. Overall, RBG_h attaches twelve words more incorrectly than RBG.

First, we checked whether the minimal grammar prevents RBG_h from selecting the gold standard parse for these five sentences, which is not the case: The gold standard annotations, including gold standard PoS tags and edge labels, are not penalized by the minimal grammar because none of them violates any constraints.

Next, we evaluated the minimal grammar on these five RBG and the RBG_h parses to answer the question why the grammar prefers the RBG_h parses to the RBG parses. Inspecting the parse with the same UAS shows that a constraint complains about an attachment in the RBG parse, which is indeed incorrect. Although the RBG_h parse has the same UAS, it represents the syntactic structure better than the RBG parse: In the gold standard annotation, the main clause is subordinated to its object clause contrary to the normal case where the object clause is subordinated. The annotation manual stipulates this attachment because otherwise a non-projective structure would arise for this sentence. If we disregard this projectivity rule and subordinate the object clause to the main clause, the RBG_h parse yields a higher UAS on the modified gold standard annotation (UAS: 34/39) than the RBG parse on both the gold standard annotation (32/39) and the modified gold standard annotation (31/19).

In case of the four RBG_h parses with a lower UAS, all of them have a lower grammar penalty than the respective RBG parses, three do not even violate any constraint. The four parses fall into two categories:

(a) The corresponding RBG parses violate hard constraints, even though RBG selected the correct regents for each of the rejected attachments. (3)

(b) An incorrect attachment rightfully violates a constraint in the corresponding RBG parse. (1)

The constraints for the parses under (a) are justifiably violated: One demands that the edge labels are consistent with the PoS tags of the dependent word, which is not the case in one parse due to a tagging error. The others require that a finite verb cannot have two complements of the same type, e. g. two subjects, which is not the case in two parses due to wrong edge labels.

Both PoS tags and edge labels cannot be changed retroactively by RBG_h: The PoS tags are determined beforehand by a PoS tagger and the labels are selected independently of the grammar penalty. Therefore, RBG_h has to change the attachments to achieve a lower penalty and, consequently, a better score in RBG_h itself. Figure 2 shows such a sentence, where RBG_h can not find the correct parse because it would require to change edge labels retroactively. Instead, RBG_h finds a parse with incorrect attachments but with a smaller grammar penalty.

The parse under (b) consists of two disconnected dependency trees although it should be connected. The RBG_h parse is connected at the price of an even lower UAS. The reason why RBG and RBG_h do not find the correct parse is probably due to faulty PoS tags (adjective and past participle are interchanged).

As we have seen before, RBG_h produces different parses only for five sentences and the UAS for these parses are at best the same as the UAS for the respective RBG parses. Thus, one hypothesis why the integration of constraints does not have a positive effect on the parsing performance is that the minimal grammar penalizes structures that occur in the gold standard annotation, and as a result prevents RBG_h from choosing the gold standard parses. To test this hypothesis, we evaluated the minimal grammar on the gold standard parses of the entire Falko-100dep corpus as well as on the RBG_h parses.

The minimal grammar does not prevent the parser from producing the correct parse: There are only two sentences for which the respective

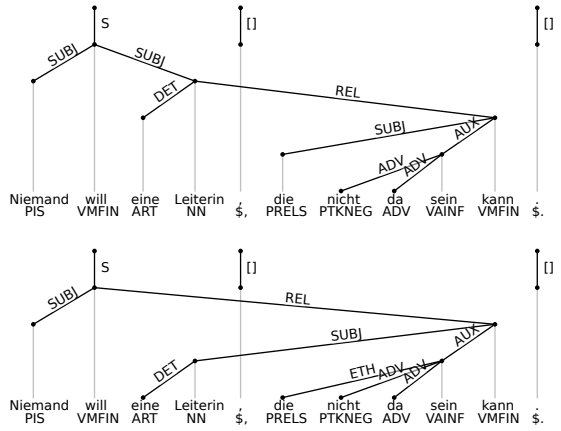

Figure 2: An example where the grammar integration deteriorates a parse ("Nobody wants a manager who cannot be there."). The RBG parse (top, UAS: 9/9) violates constraints of the minimal grammar because it assigns two subjects `SUBJ` to the finite verb "will". The RBG_h parse (bottom) is scored higher by the minimal grammar but has an UAS of only 6/9.

RBG_h parse has a lower grammar penalty than the gold standard. For both sentences the RBG parse is identical to the RBG_h parse.

The gold standard parses violate constraints in 18 sentences, which seems to be a high portion but the RBG_h parses violate constraints in 48 sentences. For 41 of these, the grammar penalty is higher than for the gold standard parse. For 32 of the 48 sentences, the gold standard does not violate any constraint.

The high amount of RBG_hparses violating constraints shows that RBG_h selects parses that are different from the gold standard annotations (including gold PoS tags), even though they are not preferred by the minimal grammar. This indicates that RBG_h can not find the gold standard parse. Reasons for this can be wrong PoS tags that the grammar penalizes or search errors due to the inability of RBG (and of RBG_h) to change dependency labels retroactively. As it turns out, this is indeed the case.

We examined the 32 sentences for which the RBG_h parse violates constraints but the gold standard does not. In 28 RBG_h parses, at least one constraint is violated due to wrong labels or wrong PoS tags. For the other 4 sentences, the constraints are rightfully violated because RBG_h chooses the wrong regent for a word. Why RBG_h did not select a different parse for these 4 sentences has still to be analyzed. Presumably, the reason is that the alternative parses do not have a lower grammar penalty because of PoS tag errors and the labeling issue.

6 Conclusions and Outlook

In this paper, different state-of-the art parsers were analyzed on free-form L2 sentences. For evaluation, we created gold-standard annotations for 100 sentences of the FalkoEssayL2 corpus.

We evaluated three different parsers on this language learner corpus. TurboParser and RBG, the two data-driven parsers, outperform jwcdg, the grammar-based parser. They produce comparable results, with TurboParser performing slightly better. Both parsers have more problems with B2 and C1 than C2 data. jwcdg on the other hand is more robust with respect to learner level. Furthermore, relabeling does not improve label accuracy.

Augmenting RBG with weighted constraints results in a decreased performance despite the grammar preferring the gold standard to the RBG output. Our analysis detected two main sources, namely erroneous PoS tags and wrong syntactic labels provided by RBG, which clash with the grammar because the hybrid parser pipeline cannot change either one retroactively. The future development of this hybrid parsing approach has to tackle the challenge of co-optimizing the syntactic labels and PoS tags during parsing in order to increase the robustness of this approach towards ill-formed data like L2 sentences.

The individual impact of the constraints has not been evaluated yet. Once the constraint-augmented parser co-optimizes, the constraint set can be optimized for the domain it is used for – in this case L2 data. If hand-written constraints only augment the statistical model of a data-driven parser, the effort needed to create these rules is orders of magnitude smaller than creating a full grammar for a rule-based parser. In addition, the constraints could be used for high-level symbolic context integration.

Currently, a detailed analysis of the differences between learner levels is hindered by the small size of annotated L2 sentences for German. For this, a larger corpus of syntactically annotated gold-standard L2 sentences for German needs to be gathered.

Our material can be obtained from `gitlab.com/nats/KONVENS-2016-material`.

References

[Berzak et al.2016] Yevgeni Berzak, Jessica Kenney, Carolyn Spadine, Jing Xian Wang, Lucia Lam, Keiko Sophie Mori, Sebastian Garza, and Boris Katz. 2016. Universal dependencies for learner English. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, pages 737–746, Berlin, Germany, August. Association for Computational Linguistics.

[Beuck et al.2013] Niels Beuck, Arne Köhn, and Wolfgang Menzel. 2013. Predictive incremental parsing and its evaluation. In Kim Gerdes, Eva Hajičová, and Leo Wanner, editors, *Computational Dependency Theory*, volume 258 of *Frontiers in Artificial Intelligence and Applications*, pages 186 – 206. IOS press.

[Björkelund et al.2013] Anders Björkelund, Ozlem Cetinoglu, Richárd Farkas, Thomas Mueller, and Wolfgang Seeker. 2013. (re)ranking meets morphosyntax: State-of-the-art results from the SPMRL 2013 shared task. In *Proceedings of the Fourth Workshop on Statistical Parsing of Morphologically-Rich Languages*, pages 135–145, Seattle, Washington, USA, October. Association for Computational Linguistics.

[Brants2000] Thorsten Brants. 2000. TnT – a statistical part-of-speech tagger. In *Proceedings of the Sixth Conference on Applied Natural Language Processing*, pages 224–231, Seattle, Washington, USA, April. Association for Computational Linguistics.

[Dhar et al.2012] Arnab Dhar, Sanjay Chatterji, Sudeshna Sarkar, and Anupam Basu. 2012. A hybrid dependency parser for Bangla. In *24th International Conference on Computational Linguistics; Proceedings of the 10th Workshop on Asian Language Resources*, pages 55–64, Mumbai, India, December.

[Foth and Menzel2006] Kilian A. Foth and Wolfgang Menzel. 2006. Hybrid parsing: Using probabilistic models as predictors for a symbolic parser. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the ACL*, pages 321–328, Sydney, Australia. Association for Computational Linguistics.

[Foth et al.2000] Kilian A. Foth, Wolfgang Menzel, and Ingo Schröder. 2000. A transformation-based parsing technique with anytime properties. In *4th Int. Workshop on Parsing Technologies, IWPT-2000*, pages 89 – 100, Trento, Italy.

[Foth et al.2014] Kilian A. Foth, Arne Köhn, Niels Beuck, and Wolfgang Menzel. 2014. Because size does matter: The Hamburg Dependency Treebank. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Thierry Declerck, Hrafn Loftsson, Bente Maegaard, Joseph Mariani, Asuncion Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Language Resources and Evaluation Conference 2014*, Reykjavik, Iceland, may. LREC, European Language Resources Association (ELRA).

[Foth2006] Kilian A. Foth, 2006. *Eine umfassende Constraint-Dependenz-Grammatik des Deutschen*. urn:nbn:de:gbv:18-228-7-2048.

[Khmylko et al.2009] Lidia Khmylko, Kilian A. Foth, and Wolfgang Menzel. 2009. Co-parsing with competitive models. In *Proceedings of the International Conference RANLP-2009*, pages 173–179, Borovets, Bulgaria. Association for Computational Linguistics.

[Köhn and Menzel2013] Arne Köhn and Wolfgang Menzel. 2013. Incremental and predictive dependency parsing under real-time conditions. In *Proceedings of the International Conference Recent Advances in Natural Language Processing RANLP 2013*, pages 373–381, Hissar, Bulgaria, September. INCOMA Ltd. Shoumen, BULGARIA.

[Köhn et al.2014] Arne Köhn, U Chun Lao, AmirAli B Zadeh, and Kenji Sagae. 2014. Parsing morphologically rich languages with (mostly) off-the-shelf software and word vectors. In *Proceedings of the 2014 Shared Task of the COLING Workshop on Statistical Parsing of Morphologically Rich Languages*.

[Krivanek and Meurers2011] Julia Krivanek and Detmar Meurers. 2011. Comparing rule-based and data-driven dependency parsing of learner language. In Kim Gerdes, Eva Hajicova, and Leo Wanner, editors, *Proceedings of Depling 2011*, pages 310–317.

[Lei et al.2014] Tao Lei, Yu Xin, Yuan Zhang, Regina Barzilay, and Tommi Jaakkola. 2014. Low-rank tensors for scoring dependency structures. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1381–1391, Baltimore, Maryland, June. Association for Computational Linguistics.

[Martins et al.2013] Andre Martins, Miguel Almeida, and Noah A. Smith. 2013. Turning on the turbo: Fast third-order non-projective turbo parsers. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 617–622, Sofia, Bulgaria, August.

[McDonald and Pereira2006] Ryan McDonald and Fernando Pereira. 2006. Online learning of approximate dependency parsing algorithms. In *EACL*, pages 81–88.

[Nivre2007] Joakim Nivre. 2007. Incremental non-projective dependency parsing. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference*, pages 396–403, Rochester, New York, April. Association for Computational Linguistics.

[Ott and Ziai2010] Niels Ott and Ramon Ziai. 2010. Evaluating dependency parsing performance on German learner language. In Markus Dickinson, Kaili Müürisep, and Marco Passarotti, editors, *Proceedings of the Ninth International Workshop on Treebanks and Linguistic Theories*, volume 9 of *NEALT Proceeding Series*, pages 175–186.

[Ragheb and Dickinson2011] Marwa Ragheb and Markus Dickinson. 2011. Avoiding the comparative fallacy in the annotation of learner corpora. In *Selected Proceedings of the 2010 Second Language Research Forum: Reconsidering SLA Research, Dimensions, and Directions*, pages 114–124, Somerville, MA. Cascadilla Proceedings Project.

[Ragheb and Dickinson2012] Marwa Ragheb and Markus Dickinson. 2012. Defining syntax for learner language annotation. In *Proceedings of the 24th International Conference on Computational Linguistics (COLING 2012), Poster Session*, pages 965–974, Mumbai, India, December.

[Ragheb and Dickinson2013] Marwa Ragheb and Markus Dickinson. 2013. Inter-annotator agreement for dependency annotation of learner language. In *Proceedings of the Eighth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 169–179, Atlanta, Georgia, June. Association for Computational Linguistics.

[Rehbein et al.2012] Ines Rehbein, Hagen Hirschmann, Anke Lüdeling, and Marc Reznicek. 2012. Better tags give better trees – or do they? *Linguistic Issues in Language Technology (LiLT)*, 7(10), 1.

[Reznicek et al.2012] Marc Reznicek, Anke Lüdeling, Cedric Krummes, Franziska Schwantuschke, Maik Walter, Karin Schmidt, Hagen Hirschmann, and Torsten Andreas, 2012. *Das Falko-Handbuch*. http://www.linguistik.hu-berlin.de/de/institut/professuren/korpuslinguistik/forschung/falko/Falko-Handbuchv2.0.pdf.

[Sagae and Lavie2006] Kenji Sagae and Alon Lavie. 2006. Parser combination by reparsing. In *Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers*, pages 129–132, New York City, USA, June. Association for Computational Linguistics.

[Schröder2002] Ingo Schröder. 2002. *Natural Language Parsing with Graded Constraints*. Ph.D. thesis, Universität Hamburg.

[Seeker and Kuhn2013] Wolfgang Seeker and Jonas Kuhn. 2013. Morphological and syntactic case in statistical dependency parsing. *Computational Linguistics*, 39(1):23–55, March.

[Zhang et al.2014] Yuan Zhang, Tao Lei, Regina Barzilay, Tommi Jaakkola, and Amir Globerson. 2014. Steps to excellence: Simple inference with refined scoring of dependency trees. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 197–207, Baltimore, Maryland, June. Association for Computational Linguistics.

Normalising Slovene data: historical texts vs. user-generated content

Nikola Ljubešić[*†], **Katja Zupan**[♢*], **Darja Fišer**[‡*], **Tomaž Erjavec**[*♢]
[*] Dept. of Knowledge Technologies, Jožef Stefan Institute
[♢] Jožef Stefan International Postgraduate School
[†] Department of Information and Communication Sciences,
Faculty of Humanities and Social Sciences, University of Zagreb
[‡] Dept. of Translation, Faculty of Arts, University of Ljubljana
nikola.ljubesic@ijs.si, katja.zupan@ijs.si,
darja.fiser@ff.uni-lj.si, tomaz.erjavec@ijs.si

Abstract

The paper presents two manually annotated Slovene language text normalisation datasets, one of historical texts and the other of tweets, and proposes several variants of character-based statistical machine translation to normalise the spelling of their words. The systems differ in whether they perform token-level or segment-level normalisation and whether they make use of additional language resources. The systems are evaluated automatically against the gold standard as well as manually, against a newly developed typology of errors intended to analyse in detail the effect of different types of data and different levels of data standardness. The evaluations show that segment-level normalisation can be useful given a high enough level of token ambiguity, that the same system can be used regardless of the data type, and that background resources will always prove useful.

1 Introduction

Processing non-standard data has been one of the core NLP challenges in the past decade. This was, in the first instance, due to intensive digitisation efforts of textual cultural heritage, which have resulted in greater access to historical language and language variants. However, accessing such non-standard data is not a straightforward process. It may be difficult for a modern reader to understand it, let alone search through it without background knowledge of historical word forms. Additionally, non-standard language also degrades the performance of off-the-shelf NLP tools, which are typically trained on contemporary standard language. These problems have re-surfaced with the emergence of micro text such as short text messages (SMS) and Twitter as well as a boom in informal communication through social networks.

In this paper we present two Slovene language datasets, one of historical texts and the other of tweets, and propose several variants of character-based statistical machine translation to standardise the spelling of their words. Section 2 overviews related work, Section 3 introduces the datasets and background resources, Section 4 details the experimental setup, Section 5 discusses the automatic and manual evaluation of the results, and Section 6 gives the conclusions and directions for further work.

2 Related work

Numerous methods have been proposed on how to process non-standard textual data, mostly by adding a pre-processing step in the form of normalisation into a standard, canonical word form, which brings the non-standard word forms closer to the readers as well as NLP tools.

Our work falls within the framework of character-based statistical machine translation (CSMT), which was first used for transliteration (Matthews, 2007) and then proposed for word normalization tasks as well.

CSMT on automatic word alignments and small training sets has been successful for modernising both historical Icelandic and Swedish (Pettersson et al., 2013). While Sánchez-Martínez et al. (2013) bootstrapped Spanish historical-to-modern lexica from corpora when no word-aligned training data was available with good results, Scherrer and Erjavec (2013) used a large lexicon of modern Slovene to identify the most similar contemporary equivalent for each unknown historical expression, thus improving tagging and lemmatization performance. Pettersson et al. (2014) obtained consistently superior results with CSMT compared to simplistic filtering or Levenshtein distance for four out of five tested languages.

Non-standard words have been receiving attention also in the context of computer-mediated communication (CMC), where erratic punctuation, misspellings, expressions in foreign languages, colloquial and dialectal expressions make it difficult to process (Sproat et al., 2001). Normalisation of CMC was also approached as a translation task (Aw et al., 2006). Li and Liu (2012) combined a single-step CSMT system with a two-stage character/phone translation method to leverage phonetic information. Pennell and Liu (2011) trained a CSMT model for expanding SMS abbreviations in English. Ljubešić et al. (2014) extended the task to all out-of-vocabulary (OOV) tokens by training a model on a manually normalised lexicon of the most salient, Twitter-specific OOV tokens. De Clercq et al. (2013) showed that a cascaded SMT system of a token-based module followed by translation at the character level gives the best word error rate reduction.

All of the above-mentioned CSMT systems perform normalisation at the token level, thus not taking into account contextual information, which could potentially lead to better performance. Successfully experimenting with token-level as well as segment-level systems is the first contribution of this paper, where, by segment-level, we mean stretches of text longer than a single token, e.g., a line or a sentence of the text. The other contributions are a uniform CSMT method obtaining best results on all datasets, regardless of the type of data or their level of standardness, and a significant positive impact of exploiting additional target-language resources.

3 Datasets

This section details the four datasets used in the experiments. They consist of easy and hard cases for normalising words in a historical setting, and in a social media one. We introduce the diachronic dataset, the user-generated one, define our notion of normalisation, and quantify the datasets. Next, the target language models, i.e. datasets used for modelling contemporary standard Slovene, are introduced.

3.1 The historical datasets

A part of the IMP language resources for historical Slovene (Erjavec, 2015a) is the goo300k manually annotated corpus (Erjavec, 2015b), comprising transcriptions of 1,100 pages (about 300,000 tokens) sampled from 88 books and one newspaper, which were published from 1584 to 1899. Each word token in the corpus is annotated for its normalised (modernised) word form(s), their part-of-speech, lemma, and — for archaic words — its gloss, i.e. contemporary synonyms. The corpus has already been used in several word modernisation experiments (Scherrer and Erjavec, 2016; Etxeberria et al., 2016).

The modern-day Slovene alphabet (called the Gaj alphabet, modelled after the Croatian alphabet by Ljudevit Gaj) was introduced into Slovene print in the 1840s; before that, the Bohorič alphabet, modelled on the German one, was used. The introduction of the Gaj alphabet was also closely preceded by a new grammar and subsequent standardisation of the language, therefore the change in the alphabet makes a convenient split between very non-standard and slightly non-standard historical language. As each text in goo300k is marked for its language variant this split is also trivial technically. After removing 3 outlier texts, we extracted from goo300k the following two datasets:

- **Bohorič**: texts written in the Bohorič alphabet published after 1750, as we have only a handful of pages from older texts which are simultaneously much harder to normalise;
- **Gaj**: texts written in the Gaj alphabet, up to 1899, which are the youngest texts in goo300k.

3.2 The social media datasets

The Janes corpus of Slovene CMC (Fišer et al., 2015) contains texts from various internet and social media platforms including Twitter. This subcorpus collects tweets of 8,750 Slovene users who have posted 7.5 million tweets with over 100 million tokens.

While tweets contain a fair amount of very non-standard text with dialectal forms, removed diacritics, phoneticised English etc., many are also completely or mostly standard. We developed a method (Ljubešić et al., 2015) to automatically classify tweets (and other texts) into three levels of technical and linguistic standardness. Technical standardness (T1, quite standard – T3, very non-standard) relates to the use of spaces, punctuation, capitalisation and similar, while linguistic standardness (L1 – L3) takes into account the level of adherence to the written norm and more or less conscious decisions to use non-standard language,

involving spelling, lexis, morphology, and word order. All tweets in the corpus have been labelled with their two standardness scores, while the authors of tweets have been manually classified into corporate ones – such as news agencies, public institutions, companies etc. – and private individuals.

On the basis of these two criteria we prepared the Twitter easy and hard datasets, both containing only private tweets:

- **L1**: 1,000 randomly sampled T1L1 tweets + 1,000 randomly sampled T3L1 tweets
- **L3**: 1,000 randomly sampled T1L3 tweets + 1,000 randomly sampled T3L3 tweets

These tweets were automatically tokenised and normalised, which was then checked and corrected manually by a team of students. The tokenisation and normalisation guidelines mostly followed the ones from the IMP project, but with some modifications regarding the differences of the medium (e.g. emoticons, urls). The annotation was performed in WebAnno (Yimam et al., 2013) where each tweet was annotated by two different annotators and then curated by the team leader (Čibej et al., 2016). For tweets that had been automatically generated by certain applications or had not been written in Slovene, the annotators had the option to mark them as irrelevant for the task.

3.3 Normalisation

What exactly constitutes a "normalised" word is a complex question, and various approaches have been proposed (Eisenstein, 2013). Most, including ours, normalise a word token only orthographically, in the trivial case into the Gaj alphabet, either from the Bohorič alphabet or from non-diacriticised text (c, s, z instead of č, š, ž), which is a common way of entering text on mobile platforms. More generally, archaic or phonetic spellings are also normalised to their standard equivalent. However, we do not substitute extinct, dialectal or slang words with their standard (near)equivalents, but only modify their spelling. This is a similar approach to Bollmann et al. (2012), who distinguish normalisation from modernisation, with the latter also changing the word to its closest modern standard equivalent as regards its morphosyntax and semantics. In cases of orthographic variation of extinct or non-standard words, we normalise them to their most common form in the relevant corpora.

In our work we map spans of original tokens into spans of normalised tokens, with further linguistic annotation assigned to the normalised ones. In the majority of cases, there is 1-1 mapping between the original and the normalised form but the contemporary standard as regards what constitutes an orthographic word also differs in some cases from past practice or that found on social media. Other approaches have typically taken a more restricted approach to normalisation, either always normalising only 1-1 (Han and Baldwin, 2011), or normalising 1-n, but not n-1 cases (Bennett et al., 2010).

To illustrate, we give in Figure 1 two cases, one from the goo300k corpus and the other from the Janes-Tweet subcorpus, both as encoded in the TEI P5 format we use for encoding our corpora. Note that here both are also lemmatised and PoS tagged, but this information is not used in the current experiments.

```
<w lemma="jagoda" ana="#Ncf">jagod</w>
<c> </c>
<choice>
 <orig>
  <w>nar</w>
  <c> </c>
  <w>več</w>
 </orig>
  <reg>
   <w lemma="veliko" ana="#Rgs">največ</w>
  </reg>
</choice>
<c> </c>
<w lemma="bolan" ana="#Agp">bolnih</w>

<w lemma="@chatek" ana="#Xa">@chatek</w>
<c> </c>
<choice>
 <orig>
  <w>Nene</w>
 </orig>
 <reg>
  <w lemma="ne" ana="#Q">ne</w>
  <c> </c>
  <w lemma="ne" ana="#Q">ne</w>
 </reg>
</choice>
<pc lemma="," ana="#Z">,</pc>
<c> </c>
```

Figure 1: Encoding of the normalised corpora. The first goo300k example maps "jagod nar več bolnih" to "jagod največ bolnih", while the second from Janes-Tweet maps "@chatek Nene, " to "@chatek ne ne, ".

3.4 Dataset sizes

Table 1 quantifies the datasets that will be used in the experiments. The first line gives the number of (sampled) texts, where the Bohorič dataset only contains pages from 15 books, while Gaj has pages from almost 70. With L3 and L1 one text is simply one tweet, so the numbers are correspondingly larger. The next line gives the number of original tokens in each dataset; it should be noted that we count cases where *n* original tokens map to one normalised token as one token. Here, by far the largest is the Gaj dataset with almost 250,000 tokens, while the others are of comparable size of about 50,000 tokens. We next give the numbers of tokens that have been normalised (we do not take into account differences in capitalisation), with the next line giving these numbers as percentages of all the tokens. With Bohorič almost half of the tokens needed normalisation, which is of course also due to the differences in the alphabet. With Gaj only about one tenth needed to be normalised, less than in the L3 Twitter dataset, where the number is almost 17%. Finally, L1 is, of course, the most like standard Slovene, with about 3.3% normalisation. Finally, we also give the number of split or joined words as regards normalisation. These cases pose special technical as well as methodological problems in the process of normalisation, even though the numbers are rather low, with all being less than 1%, while their distribution follows the percentages of normalised tokens.

	Bohorič	Gaj	L3	L1
Texts	15	69	1,983	1,957
Tokens	75,210	249,146	54,694	47,950
Norm.	36,493	29,012	9,203	1,572
	48.52%	11.64%	16.83%	3.28%
Multi.	641	1,093	276	131
	0.85%	0.44%	0.50%	0.27%

Table 1: Sizes of the four datasets.

3.5 Splitting the datasets

For our experiments we split each of the four datasets into training, development, and test parts following a 80:10:10 ratio. Sampling was performed by shuffling on segment, i.e. sentence level.

Having development data was necessary as SMT systems without tuning, i.e. with default parameter values, regularly underperform in comparison to tuned systems.

In the interests of replicability of experiments the pre-processed data with our splits is published via the CLARIN.SI language resource repository, c.f. Ljubešić et al. (2016).

3.6 Target language datasets

While additional parallel data for SMT is expensive and therefore hard to acquire, including bigger target language models, which regularly improves translation quality, is a rather simple task as for most target languages there are monolingual resources available. In our experiments we used two corpora of our target language, standard contemporary Slovene, of different quality, size, and costs of construction.

Web corpora are cheap to acquire and can be quite large, and we used **slWaC** (Ljubešić and Erjavec, 2011), a one billion token corpus crawled from the *.si* top level domain, using language identification to filter out non-Slovene texts.

However, Web corpora are noisy and also contain non-standard language, which e.g. is not diacriticised, potentially leading to low-quality models of standard Slovene. This is the reason we also use **Kres** (Logar Berginc et al., 2012), a 100 million word reference and balanced corpus of contemporary Slovene, which contains, for the most part, proof-read texts.

4 Experimental setup

Our experiments have been carried out with the tools of the standard SMT pipeline: MGIZA[1], a multi-threaded version of GIZA++ (Och and Ney, 2003) for alignment, Moses[2] (Koehn et al., 2007) for phrase extraction and decoding, and KENLM[3] (Heafield, 2011) for language modelling. In particular, we have explored character-based SMT, where a word or a segment of the text is split into individual characters, borders between tokens being encoded with underscores, and the resulting string is then translated.

As will be discussed below, we use two granularities of translation, one of tokens and the other of segments. While segments can, in general, be any contiguous stretch of text, in our experiments segments are sentences.

[1] https://github.com/moses-smt/mgiza
[2] http://www.statmt.org/moses/
[3] https://kheafield.com/code/kenlm/

4.1 Research questions

In this paper we are interested in answering our two main research questions:

1. Is there one single CSMT setting that performs best on text normalisation regardless whether we normalise historical texts or user-generated content?

2. Can we outperform the traditional token-by-token normalisation by translating whole segments at a time, therefore taking into account the context in which a token occurred?

We answer these questions by running the following experiments on each of the four datasets:

- experiment1: comparing token-level and segment-level translators when using language models (LMs) based on training data only;

- experiment2: comparing the token-level and segment-level approaches when including additional LMs.

For token-level systems we use order-7 language models while for segment-level systems we opt for order-10 language models. Our early experiments have shown that these orders yield best results in each of the approaches.

We additionally look into the impact of reordering, traditional part of SMT, and time and memory requirements for each of the approaches.

4.2 Evaluation

We evaluate all our experiments on the level of segments. This means that in case of the token-level translator we translate token by token and then combine these translations into segments before evaluating. During all the experiments we evaluated the segment pairs with two metrics: character-level Levenshtein distance normalised by the length of the reference data and the token-level BLEU metric (Papineni et al., 2002). In the remainder of the paper we report the Levenshtein metric only as it was shown for these two metrics to correlate in all experiments with a Pearson's correlation coefficient greater than 0.99.

We perform statistical significance testing on the Levenshtein evaluation metric by using the approximate randomisation test (Yeh, 2000) with 1000 iterations.

4.3 Baselines and ceilings

In our experiments we use two different baselines: the leave-as-is baseline (LAI), which does not transform the input in any way, and the most-frequent-translation baseline (MFT), which exchanges each token with the token most frequently normalised to in the training data. In the MFT baseline ties are resolved randomly.

We use two ceilings as well, both based on the MFT baseline. These ceilings are informative as to what extent word form transformations are ambiguous, i.e. for what amount of error the only solution is disambiguation in context. The first ceiling, MFT is actually a MFT baseline both trained and tested on the test data. The second ceiling, MFTr is the MFT baseline trained both on training and testing data, while tested on testing data only. We consider the second ceiling to be more realistic as it learns on more than testing data, therefore having a lower probability of measuring rare token transformations as the most frequent ones.

5 Results

5.1 Automatic evaluation

5.1.1 First experiment

In the first set of experiments we train and tune token-level and segment-level translators for each of the four datasets. Additionally, we train translators that use reordering models and those that do not use reordering. Here we report the results for the translators that do not use reordering models as the difference between the systems using and not using reordering has no statistical significance.[4]

Table 2 gives the two baselines and two ceilings along with the results of our eight initial systems.

The LAI baseline draws a clear picture about the level of intervention necessary in each of the texts. While in Bohorič 18% of characters have to be transformed, in the L1 dataset less than 1% needs intervention.

Applying the MFT baseline on hard datasets (Bohorič and L3) resolves more than half of the problems. The lowest error reduction with MFT on the L1 dataset is 17.33%.

The two ceilings show that the level of ambiguity on the token transformation level is actually very

[4] On any of the eight pairs of systems (four datasets, each token- and segment-level), the lowest p-value obtained was 0.076, the second and third being 0.138 and 0.261, in roughly half of the cases reordering was performing better, regardless of the type of translation (token- or segment-level).

	baselines		ceilings		first experiment		
	LAI	MFT	MFT	MFTr	token	segm	Δ
Bohorič	17.63	6.46	0.34	0.44	**1.55**	1.92	-23.9
Gaj	3.13	1.43	0.23	0.29	**1.01**	1.15	-13.9
L3	5.15	2.44	0.37	0.54	2.19	2.12	3.20
L1	0.75	0.62	0.05	0.07	0.41	0.43	-4.88

Table 2: Results of the first set of experiments (no additional LMs) as percentages of character errors. Δ is error reduction (in %) by the segment-level system.

low. While on the L1 dataset there is almost no ambiguity (if we saw enough token transformations, only 0.07 percent of characters would not be normalised correctly), in case of Bohorič and L3 every 200th character would be wrongly normalised.

The results of the first experiments show a very similar performance regardless of whether token-level or segment-level translators were used, with a small but consistent better performance of the token-level systems.

The results on the datasets where a statistically significantly better result was obtained are given in bold. Interestingly, on historical datasets the token-level systems perform significantly better than segment-level systems.

The only dataset in which the segment-level system performs better, although not statistically significant, is the L3 dataset, on which the ceilings are also most distant from a perfect normalisation, i.e. the gain to be obtained by taking into account a token's context is the highest.

5.1.2 Second experiment

We continue our experiments by including additional language models into the translators. The idea behind this second experiment is twofold:

- there is not much training data on which the initial language models are based, and adding easy-to-obtain standard data in the form of additional language models is easy in the case of Slovene as is for most languages;
- segment-level translators need much more target-language data than the token-level ones; our assumption is that the segment-level systems on the datasets where more token-level ambiguity is present (like Bohorič and L3) could win over the token-level systems once they obtain enough context evidence.

The additional language models are built from the Kres and the slWaC corpora, again of order 7 in case of the token-level approach and of order 10 in case of the segment-level approach. We combine language models by adding more entries in the moses.ini file and letting MERT weight each language model on our development data.

We experiment by adding each language model separately to the setting using the existing training data language model, and by using all three language models simultaneously. The results of this set of experiment are given in Table 3.

The results confirmed our assumptions: on Bohorič and L3, where token-level ambiguity is higher, segment-level outperform token-level approaches, while on the Gaj and the L1 datasets the token-level still outperforms the segment-level approach.

On the Bohorič dataset in all three LM settings the segment-level approach outperforms the token-level one, with error reduction spanning from 6% to 12%, in which case the difference between the token- and the segment-level approach is statistically significant. Similarily, on the L3 dataset, once the LM based on web data is added, the segment-level approach obtains better results with error reduction of 7% and 10%, the latter being statistically significant. On the two remaining datasets the token-level approach always performs better, but nowhere with a statistically significant difference.[5]

When comparing best-performing systems using training data only and using additional LMs, regardless of the setting (token- vs. segment-level), the error reduction on the Bohorič dataset reaches 14%, on the Gaj dataset 10%, on the L3 dataset 22% and on the L1 dataset 17%, proving that, regardless of the approach, significant and easy-to-obtain improvements can be achieved by expanding the set

[5]Similar trends were observed in (Scherrer and Ljubešić, 2016) on normalising Swiss German where the MFTr ceiling, calculated as token accuracy, is 93% with an error reduction when moving from token-level to segment-level normalisation of 20%. In our datasets the MFTr ceiling, when calculated as token accuracy, is 94.46% for Bohorič, 96.50% for Gaj, 93.88% for L3 and 98.37% for L1.

	Bohorič			Gaj			L3			L1		
LM	token	segm	Δ	token	segm	Δ	token	segm	Δ	token	segm	Δ
train	1.55	1.92	-23.4	1.01	1.15	-13.4	2.19	2.12	3.0	0.41	0.43	-4.0
+kres	1.50	1.40	7.2	0.90	0.94	-4.2	1.81	1.82	-0.1	0.38	0.43	-12.0
+slwac	1.48	1.39	6.4	0.91	1.04	-13.3	1.76	**1.58**	10.2	0.37	0.38	-3.4
+both	1.51	**1.33**	11.8	0.91	0.93	-3.1	1.77	1.65	6.6	0.34	0.38	-10.8

Table 3: Results of the second set of experiments (additional LMs). Δ is error reduction (in %) by the segment-level system.

of language models used.

During the second set of experiments we also measured the time and space requirements of decoding with the token-level and segment-level decoders. A reasonable assumption is that both space and time requirements of the segment-level decoder will be orders of magnitude higher as both its language models as well as its search space are much bigger. The time necessary to translate each of the test sets was roughly 3 times longer in case of the segment translator. Regarding the memory requirements, the difference became quite drastic with 25 times more memory consumption of the segment-level translator when all three language models (train+kres+slwac) were used.

Regarding our two main research questions, the answers obtained through these experiments are the following:

1. regardless of the type of text to be normalised, using the baseline Moses setting with removed reordering (no lexical reordering model and distortion set to zero) and additional language models yields best results

2. if the level of token ambiguity is high, segment-level translation can give significant improvements in translation quality, but with a heavy hit on time and memory requirements

5.2 Manual evaluation

To obtain a better insight into the errors, we made a manual evaluation and comparison between the best token-based and the best segment-based normaliser.

A sample of random 100 word forms incorrectly normalised by at least one of the two normalisers was selected from each of the four datasets.[6] The errors in these 399 instances were manually categorised into 8 types, with the error types chosen with respect to their potential for introducing improvements in the method of normalisation:

- **XF**: corruption of foreign language words (mostly German, Latin or English; e.g. 'interne' instead of 'intern'); here (word or span-level) language identification would be helpful in preventing such wrong normalisations to contemporary standard Slovene;

- **TR**: transliteration error, either from Bohorič or by failing to rediacriticise for L1 and L3 (e.g. 'tisina' instead of 'tišina' (silence)); a special module for rediacriticisation, such as Ljubešić et al. (2016), could reduce these errors;

- **WB**: a word boundary error (e.g. 'naj lepši' instead of 'najlepši'); these are interesting as they are by definition outside the scope of the token-based normaliser;

- **END**: an error in the inflectional ending (e.g. the normaliser failing to change the archaic adjectival suffix "-iga" into the contemporary "-ega", i.e. "lepiga" instead of "lepega"); such errors could be taken care of by introducing suitable morphological processing into the language model;

- **LEX-D**: an error where the wrong word form was predicted, but this word form does in fact exist, however, it belongs to a different part of speech from the correct one (e.g. preposition 'k' (to) instead of conjunction 'ko' (when)); these errors could be alleviated by having a POS tagger determine the expected POS of the target word;

- **LEX-S**: same error as LEX-D, except that the predicted word has same part of speech as the correct one (e.g. preposition 'o' (about) instead of preposition 'ob' (by)); these are

[6] To be exact, only 99 word form errors were taken from L1, as there were only that many errors in this dataset.

among the more intractable errors, and could be resolved only by having access to some sort of word sense disambiguation;

- **VAL**: a (lexical) validity error, where the predicted word does not exist, although it does follow the spelling conventions of contemporary standard Slovene (e.g. 'izdatelj' instead of 'izdajatelj' (publisher)); such errors could be prevented by having a representative lexicon against which to filter hypotheses;

- **OTH**: multiple errors, or errors that could not be categorized into any of the categories listed above (e.g. 'po semi' instead of 'pozimi' (adverb meaning during the winter); 'Avstri' instead of 'Avstrija' (Austria)); these would probably not be corrected even if all of the above-mentioned extra modules were in place.

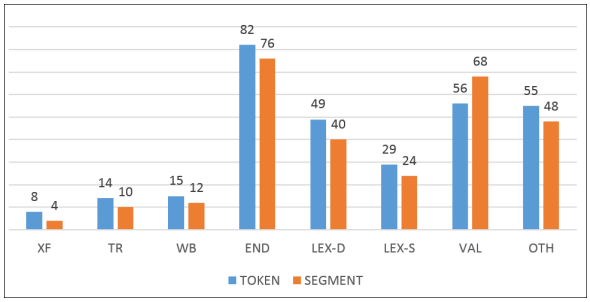

Figure 2: Token vs. segment normalisation on combined datasets.

Figure 2 shows the comparison between the token-based and segment-based systems with all datasets combined. It shows that the segment-based approach outperforms the token-based one in all error types but one, i.e. in the category of validity errors. Only in this case, taking context into account hurts rather than helps: by staying limited to tokens, i.e. word forms, more valid guesses — albeit not necessarily entirely correct — are produced. In total, the token-based system made 308 errors in the analysed sample, while the segment-based one committed 282 errors.

The analysis in Figure 3 shows the distribution of errors made by the segment-based system as the better performing normaliser.

Errors related to foreign words, transliteration, word boundaries, lexical homographs (different POS) and non-categorized errors are more frequent

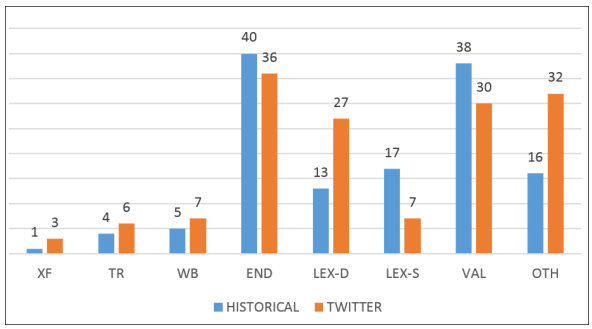

Figure 3: Normalisation on historical and Twitter datasets for the segment-based system.

when normalising tweets, while inflectional endings, lexical homographs (same POS) and validity errors are more typical when normalising the historical datasets. This is to be expected, as many of these errors stem from diachronic changes from historical to modern Slovene.

In both datasets the prevailing type of error is inflectional endings (26.8% of total errors), with the second most common, again for both datasets, being lexical validity, where the normalisers proposed a non-existing word. If we exclude Other errors, the third most common error type is incorrect but existing words with the wrong POS. Interestingly, there are very few errors related to foreign words, transliteration and word boundaries, so from an accuracy point of view, it is not worth investing resources into fixing them.

6 Conclusions

The paper presented experiments in normalising words in historical and user-generated Slovene texts, additionally investigating the differences between cases of easy and hard normalisation. We used CSMT for the task, where we investigated the differences between token-level and segment-level normalisation as well as (not) using additional background resources for better probability estimates regarding the target language.

The experiments show that if token-level ambiguity, measured by training the most-frequent-translation system on both training and testing data and calculating normalised character-level Levenshtein distance, is above 0.04, training a segment-level system could prove to be useful. This, naturally, does not depend on the level of token ambiguity only, but on the amount of parallel and target-side data as well. By applying segment-level approaches on the two datasets with higher token-

level ambiguity, we achieved error reduction of more than 10%. Adding more language models should always be considered as this is not a costly task, and error reductions on our datasets reached between 10% and 22%. Additionally we have also shown that there is no need to use different systems for historical and modern non-standard texts, as the best performing one caters for both datasets.

We performed a manual error classification of the two best performing systems on all four datasets, which showed that about a quarter of all errors are due to poorly normalised inflectional endings, followed by normalising to non-existent words and then by incorrect but existing words with the wrong POS.

Taking into account the most frequent errors of our current systems, there are two main directions how we can improve our result.

The first direction should focus on enriching surface contextual information, either by including larger language models, language models of higher order, language models of higher-order events like tokens, or language models with better abstraction capabilities like neural language models.

The second direction should focus on a higher linguistic abstraction like morphosyntax. Having enough data to train a reasonable part-of-speech tagger over source data could provide us with reasonable morphosyntactic annotation that could be used in the translation process via factored machine translation. Including factors only on the target side, for which very good morphosyntactic annotation can be obtained, should also be investigated.

Acknowledgments

The research leading to these results has received funding from the Slovenian Research Agency within the national basic research project "Resources, Tools and Methods for the Research of Nonstandard Internet Slovene" (J6-6842, 2014-2017), the Young Researcher programme (no. 37487) and the Swiss National Science Foundation grant no. IZ74Z0_160501 (ReLDI).

References

AiTi Aw, Min Zhang, Juan Xiao, and Jian Su. 2006. A phrase-based statistical model for sms text normalization. In *Proceedings of the COLING/ACL on Main conference poster sessions*, pages 33–40. Association for Computational Linguistics.

Paul Bennett, Martin Durrell, Silke Scheible, and Richard J. Whitt. 2010. Annotating a historical corpus of German: A case study. In *Proceedings of the LREC 2010 Workshop on Language Resource and Language Technology: Standards - state of the art, emerging needs, and future developments*, pages 64–68, Paris. ELRA.

Marcel Bollmann, Julia Krasselt, and Florian Petran. 2012. Manual and semi-automatic normalization of historical spelling – Case studies from Early New High German. In *In Proceedings of KONVENS 2012 (LThist 2012 workshop*, pages 342–350.

Jaka Čibej, Darja Fišer, Tomaž Erjavec, and Špela Arhar Holdt. 2016. Razvoj učne množice za izboljšano označevanje spletnih besedil (The development of a training set for better annotation of internet texts). In *Conference on Language Technologies and Digital Humanities*, Ljubljana, September.

Orphée De Clercq, Bart Desmet, Sarah Schulz, Els Lefever, and Véronique Hoste. 2013. Normalization of dutch user-generated content. In *9th International conference on Recent Advances in Natural Language Processing (RANLP 2013)*, pages 179–188. INCOMA.

Jacob Eisenstein. 2013. What to do about bad language on the internet. In *In Proc. of NAACL.*

Tomaž Erjavec. 2015a. The IMP historical Slovene language resources. *Language Resources and Evaluation*, pages pp. 1–23.

Tomaž Erjavec. 2015b. *Reference corpus of historical Slovene goo300k 1.2*. Slovenian language resource repository CLARIN.SI, http://hdl.handle.net/11356/1025.

Izaskun Etxeberria, Iñaki Alegria, Larraitz Uria, and Mans Hulden. 2016. Evaluating the Noisy Channel Model for the Normalization of Historical Texts: Basque, Spanish and Slovene. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Portorož, Slovenia, may. European Language Resources Association (ELRA).

Darja Fišer, Nikola Ljubešić, and Tomaž Erjavec. 2015. The Janes corpus of Slovene user generated content: construction and annotation. In *International Research Days: Social Media and CMC Corpora for the eHumanities*, Rennes, October.

Bo Han and Timothy Baldwin. 2011. Lexical normalisation of short text messages: Makn sens a #twitter. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, HLT '11, pages 368–378, Stroudsburg, PA, USA. Association for Computational Linguistics.

Kenneth Heafield. 2011. KenLM: Faster and smaller language model queries. In *In Proc. of the Sixth Workshop on Statistical Machine Translation*.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the ACL 2007 demonstration session*, pages 177–80, Prague, Czech Republic.

Chen Li and Yang Liu. 2012. Normalization of text messages using character-and phone-based machine translation approaches. In *Proceedings of InterSpeech*, Portland, Oregon, USA.

Nikola Ljubešić, Tomaž Erjavec, and Darja Fišer. 2014. Standardizing tweets with character-level machine translation. *Computational Linguistics and Intelligent Text Processing*, pages 164–175.

Nikola Ljubešić, Darja Fišer, Tomaž Erjavec, Jaka Čibej, Dafne Marko, Senja Pollak, and Iza Škrjanec. 2015. Predicting the Level of Text Standardness in User-generated Content. In *Proceedings of Recent Advances in Natural Language Processing*.

Nikola Ljubešić, Katja Zupan, Darja Fišer, and Tomaž Erjavec. 2016. *Dataset of normalised Slovene text KonvNormSl 1.0*. Slovenian language resource repository CLARIN.SI. `http://hdl.handle.net/11356/1068`.

Nikola Ljubešić and Tomaž Erjavec. 2011. hrWaC and slWac: Compiling Web Corpora for Croatian and Slovene. In Ivan Habernal and Václav Matousek, editors, *TSD*, volume 6836 of *Lecture Notes in Computer Science*, pages 395–402. Springer.

Nikola Ljubešić, Tomaž Erjavec, and Darja Fišer. 2016. Corpus-based diacritic restoration for south slavic languages. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*. European Language Resources Association (ELRA), may.

Nataša Logar Berginc, Miha Grčar, Marko Brakus, Tomaž Erjavec, Špela Arhar Holdt, and Simon Krek. 2012. *Korpusi slovenskega jezika Gigafida, KRES, ccGigafida in ccKRES: gradnja, vsebina, uporaba [The Gigafida, KRES, ccGigafida and ccKRES corpora of Slovene language: compilation, content, use]*. Zbirka Sporazumevanje. Trojina, zavod za uporabno slovenistiko: Fakulteta za družbene vede, Ljubljana, Slovenia.

David Matthews. 2007. Machine transliteration of proper names. *Master's Thesis, University of Edinburgh, Edinburgh, United Kingdom*.

Franz Josef Och and Hermann Ney. 2003. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: A method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, pages 311–318, Stroudsburg, PA, USA. Association for Computational Linguistics.

Deana Pennell and Yang Liu. 2011. A character-level machine translation approach for normalization of sms abbreviations. In *IJCNLP*, pages 974–982.

Eva Pettersson, Beáta Megyesi, and Jörg Tiedemann. 2013. An smt approach to automatic annotation of historical text. In *Proceedings of the Workshop on Computational Historical Linguistics at NODALIDA 2013, NEALT Proceedings Series*, volume 18, pages 54–69.

Eva Pettersson, Beáta Megyesi, and Joakim Nivre. 2014. A multilingual evaluation of three spelling normalisation methods for historical text. *Proceedings of LaTeCH*, pages 32–41.

Felipe Sánchez-Martínez, Isabel Martínez-Sempere, Xavier Ivars-Ribes, and Rafael C Carrasco. 2013. An open diachronic corpus of historical spanish: annotation criteria and automatic modernisation of spelling. *arXiv preprint arXiv:1306.3692*.

Yves Scherrer and Tomaž Erjavec. 2013. Modernizing historical slovene words with character-based smt. In *BSNLP 2013-4th Biennial Workshop on Balto-Slavic Natural Language Processing*.

Yves Scherrer and Tomaž Erjavec. 2016. Modernising historical slovene words. *Natural Language Engineering*, FirstView:1–25, 5.

Yves Scherrer and Nikola Ljubešić. 2016. Automatic normalisation of the swiss german archimob corpus using character-level machine translation. In *Proceedings of KONVENS 2016*.

Richard Sproat, Alan W Black, Stanley Chen, Shankar Kumar, Mari Ostendorf, and Christopher Richards. 2001. Normalization of non-standard words. *Computer Speech & Language*, 15(3):287–333.

Alexander Yeh. 2000. More accurate tests for the statistical significance of result differences. In *Proceedings of the 18th Conference on Computational Linguistics - Volume 2*, COLING '00, pages 947–953, Stroudsburg, PA, USA. Association for Computational Linguistics.

Seid Muhie Yimam, Iryna Gurevych, Richard Eckart de Castilho, and Chris Biemann. 2013. Webanno: A flexible,web-based and visually supported system for distributed annotations. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (System Demonstrations) (ACL 2013)*, pages 1–6, Stroudsburg, PA, USA, August. Association for Computational Linguistics.

Integrating corpora of computer-mediated communication in CLARIN-D: Results from the curation project ChatCorpus2CLARIN

Harald Lüngen
Institut für Deutsche Sprache
luengen@ids-mannheim.de

Michael Beißwenger
Universität Duisburg-Essen
michael.beisswenger@uni-due.de

Eric Ehrhardt
Universität Mannheim
eric.ehrhardt@gmx.de

Axel Herold
Berlin-Brandenburgische Akademie der Wissenschaften
herold@bbaw.de

Angelika Storrer
Universität Mannheim
astorrer@mail.uni-mannheim.de

Abstract

We introduce our pipeline to integrate CMC and SM corpora into the CLARIN-D corpus infrastructure. The pipeline was developed by transforming an existing CMC corpus, the Dortmund Chat Corpus, into a resource conforming to current technical and legal standards. We describe how the resource has been prepared and restructured in terms of TEI encoding, linguistic annotations, and anonymisation. The output is a CLARIN-conformant resource integrated in the CLARIN-D research infrastructure.

1 Introduction

Written language in computer-mediated communication (henceforth CMC) and social media (SM) is an important type of non-standard language usage. Although there has been a lot of research on CMC and SM genres in linguistics and social sciences, most of these studies rely on small datasets or corpora that are not publically available. It would be highly desirable to integrate more CMC and SM corpora in corpus collections and to set up common standards for the representation and annotation of these new forms of communication and their structural and linguistic peculiarities.

The project Chatcorpus2CLARIN aimed to explore the prerequisites for integrating CMC und SM corpora into the CLARIN-D corpus infrastructure by transforming an existing CMC corpus, the Dortmund Chat Corpus, into a resource that conforms to current corpus standards. This integration will allow for a systematic corpus-based analysis of CMC and SM discourse as compared with discourse in edited text (as represented in the text corpora at the CLARIN-D centres Berlin-Brandenburgische Akademie der Wissenschaften (BBAW) and Institut für Deutsche Sprache, Mannheim (IDS)) and to spoken conversations (as represented in the spoken language corpora at IDS). The method of transformation developed for this curation project, which is described in this paper, is regarded as a model for the CLARIN curation of CMC corpus resources in general. (Thus, throughout this paper, the term *curation* is used in the concrete sense of "CLARINification".)

The paper is structured as follows: In the following section we provide information on the curated resource (Chat Corpus 1.0) and discuss some legal issues that had to be considered in the context of the curation. In our main Section 3, we describe how this resource has been restructured to conform to current standards for the representation of corpora in the Digital Humanities context. In Section 4, we describe the resulting resource Chat Corpus 2.0 and outline the added values that will be created by integrating this resource into the CLARIN infrastructure.

2 Resource and conditions

2.1 The resource

The Dortmund Chat Corpus (Beißwenger, 2013) has been collected at Dortmund Technical University between 2000 and 2006 as a resource for researching the peculiarities and linguistic variation in written computer-mediated communication. The corpus comprises 478 chat documents (logfiles) with 140,240 user postings or 1M words of German chat discourse from heterogeneous sources representing the use of chats in a wide range of application contexts (social chats, advisory chats, chats in the context of learning and teaching, moderated chats in the media context). The corpus has been annotated using a homegrown XML format (ChatXML) that describes (1) the basic structure and properties of chat logfiles and postings, (2) selected netspeak phenomena such as emoticons, interaction words, addressing terms, nicknames

and acronyms, (3) selected metadata about the chat platforms and chat users. Since 2005, a large subset of the corpus has been available in ChatXML, for download and offline querying and as an HTML version for online browsing[1]. It has been widely used as a resource for studying and teaching the characteristics of German CMC discourse.

2.2 Legal issues

Prior to the integration of the curated resource in CLARIN infrastructures, we sought a legal opinion to decide on questions regarding the republishability of the material as a whole or in parts, i.e. the provisions needed with respect to questions of copyright and personality rights as well as questions regarding the licensing of the corpus.

The corpus comprises personal communication in both private, educational and public chatrooms. To prevent the public revelation of participants' personal data, the possibility to identify individuals from their utterances (with the exception of public figures) needs to be circumvented as much as possible. This is achieved by means of the anonymisation of names, nicknames, host names and IP addresses, geographical names (e.g. address data) etc. (see Section 3.4 for a technical discussion of the anonymisation performed). In accordance with the legal opinion, some parts of the resources data must not be made available to the public at all, notably those parts where personality rights of participants are strongly affected. This applies to all data obtained from chat-based psycho-sociological counseling services in the original corpus (8 chat logfiles with in sum 88227 tokens). Here, due to the highly personal context represented in the discourse, anonymisation measures alone are unlikely to prevent the identification of individuals. Consequently, these resources were removed from the final corpus.

The legal opinion saw no indication of concerns regarding copyright (German *Urheberrecht,* specifically) as it acknowledges that the collected discourses and the single user contributions in the overwhelming majority of cases do not represent works of art. Protectable under German law however, is the work committed in the course of collection, curation and transformation of the data into the format of the intended linguistic database. Therefore and in accordance with our goal to provide the resource as openly as possible, we followed the lawyers' suggestion and provided the resource with a CreativeCommons licence (CC BY 4.0), which allows for the protection of database creator rights.

[1] from http://www.chatkorpus.tu-dortmund.de

3 Method

One goal of the project was to develop a model for the integration of CMC and SM corpora into the CLARIN-D corpus infrastructures at BBAW and IDS. The Dortmund Chat Corpus served as a use case to demonstrate how such an integration could be accomplished in a way that the target resource (1) conforms to established standards for the representation and linguistic annotation of corpora in the Digital Humanities context and (2) can be used for comparative analyses with other types of corpus resources in CLARIN-D (text and speech corpora). A visualisation of the workflow developed in the project is shown in Figure 1; the steps and resources of the pipeline are described in the following subsections.

3.1 TEI representation

For many years, the guidelines of the Text Encoding Initiative (TEI) have been the de facto standard framework for text (and text structure) encoding in the Digital Humanities. Consequently, the TEI guidelines serve as a suggested best practice in the CLARIN-D corpus research infrastructure (CLARIN-D AP 5, 2012) for different text types, such as historical and contemporary books, newspapers, and other printed resources. However, when trying to model CMC in TEI, there are two fundamental challenges: Firstly, as argued above, CMC shares characteristics with both text and spoken conversation. On the one hand, CMC constitutes dialogic interaction in which each communicative move creates or changes the context for follow-up moves. On the other hand, written CMC is organised through the exchange of stretches of written text which have been completed before they are transmitted and read. A basic model for the representation of user contributions to written CMC (post, s.b.) should reflect these properties. The second challenge is that a basic schema for CMC should be flexible enough to represent multimodal CMC interactions as well, such as the interactions of teachers and students on an e-learning platform. So far, the official TEI P5 Guidelines do not include features that model these basic characteristics, see also Beißwenger et al. (2012).

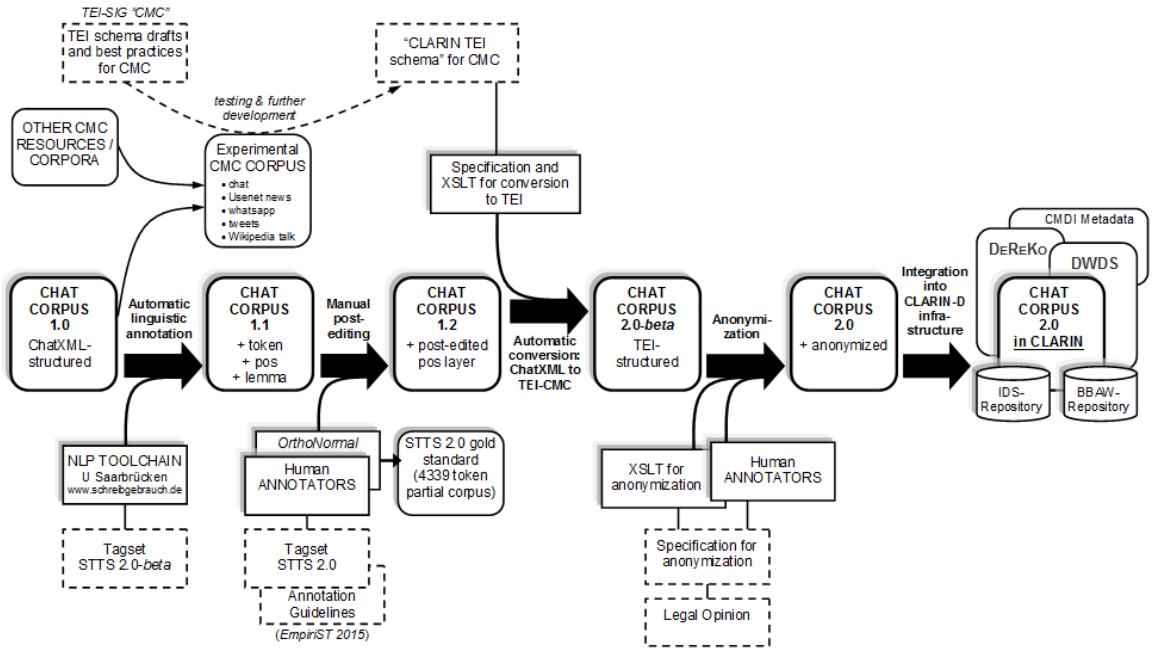

Figure 1: CMC corpus curation pipeline

As a consequence, several corpus initiatives represented in the TEI Special Interest Group Computer-Mediated Communication (TEI CMC SIG) have put forward TEI customisations for different types of CMC and social media genres in the past few years. Two schema drafts resulting from corpus projects in Germany and France have been published by Beißwenger et al. (2012) (DeRiK schema) and Chanier et al. (2014) (CoMeRe schema). The main features of these proposals are the introduction of the elements <post> for written user contributions to CMC interactions, thus combining features of text divisions and spoken utterances (Beißwenger et al., 2012), and <prod> for the representation of non-verbal acts (Chanier et al., 2014). The elements <post>, <prod>, and <u> (the latter marking a spoken utterance in TEI) have been (re-)defined such that they may be combined within one interaction (ibid.).

In the CLARIN-D project, we tested the suitability of the CoMeRe schema for our project by compiling an experimental corpus of German CMC data consisting of chat data (two chat logfiles from the Dortmund Chat corpus), Usenet news (94 news messages from one newsgroup of the Usenet corpus in DeReKo, cf. Schröck & Lüngen (2015)), Wikipedia discussions (five talk pages with 10148 tokens), twitter data (1412 tokens of donated tweets from two different twitter channels), and What'sApp data (1907 messages from the data collected in the project "What's up, Deutschland?"[2]. We then manually annotated the experimental corpus according to the CoMeRe TEI schema and as a result identified a set of CMC features that could not be encoded using it, i.e. for which we had to find new solutions within TEI. Hence, we went for a new, project-specific TEI customisation, dubbed CLARIN CMC-TEI. Our focus was to customise features and to describe best practices for representing the chat data, while the other CMC genres in the experimental corpus were used as supporting or additional evidence.

We decided that for the present project, lexical CMC phenomena such as action words, acronyms, emoticons, and addressing terms are more appropriately annotated on the part-of-speech level, as the tagset STTS 2.0 with corresponding extensions for CMC has recently been introduced cf. Section 3.2), and one tagging system has already been trained for it using CMC data (Horbach et al., 2014), with excellent results on chat data. The POS tags were included in the @type attribute of the <w> elements which mark the tokens. Thus, no TEI customisation would be needed for accommodating these anymore.

The features and solutions of our CLARIN CMC TEI schema are of three types with respect to their relation to the generic TEI P5 guidelines (version 2.9.0):

[2]http://www.whatsup-deutschland.de

1. Additions of new models for the elements <post>, <prod>, <signatureContent>, and for the two model classes model.floatP.cmc, and model.divPart.cmc.
2. Modifications of existing TEI P5 models so that they fit certain CMC phenomena (e.g. adding @who, @auto; changing <post>, <p>, <s> and <quote>, to include the new model class model.floatP.cmc).
3. "Best practice" solutions using existing TEI P5 models according to specific CMC practice, e.g. use of <w> and <phr> and their attributes for representing word tokens and phrases, respectively, and their POS tags and lemma information.

The first two types are true TEI customisations and have been implemented in our Chat2CLARIN TEI schema. The third type is entirely based on the existing TEI framework and effectively suggests restrictions for the use and application of generic TEI models.

In the following, we explain in more detail one example of each type of solution.

3.1.1 Addition of a new model: <post>

The element <post> models a written contribution to an ongoing CMC interaction which (1) has been composed by its author in its entirety as part of a private activity and (2) has been sent to the server en bloc (Beißwenger et al., 2012).

From the perspective of its addressees/readers, a post is a passage of text that has been composed in advance. Posts occur in a wide range of written CMC genres: as user messages in chats and WhatsApp dialogues, as SMS messages, as tweets in Twitter timelines, as individual comments following a status update on Facebook pages, as posts in forum threads, as contributions on Wikipedia talk pages or in the comments section of a weblog.

The <post> element is provided with five post-specific, optional attributes that serve to model a small set of post metadata: Firstly, @correspAction, which is used to encode the 'sent'/'delivered'/'read' status of a post as in WhatsApp dialogues; the name of the attribute follows the element of the same name in the TEI standard. Secondly, @replyTo indicates to which previous post the current post replies or refers to. The remaining three, @revisedBy, @revisedWhen, and @indentLevel are adapted from the DeRiK-Schema (Beißwenger et al., 2012).

3.1.2 Modification of existing TEI P5 models: The attributes @who and @auto

In the TEI guidelines, the attribute @who indicates the person, or group of people, to whom the element content is ascribed. Besides its application in the <teiHeader>, it is most notably used for references from individual utterances (<u>) to discourse participants (or fictional characters in the case of literary works). As the equivalent to <u> in our schema is <post>, we allow @who for posts in order to indicate post creators. The participants metadata are recorded in a participant list (<particDesc>) within the <profileDesc> section of the <teiHeader> (see Section 3.4 for issues concerning anonymisation) providing each participant with a unique xml:id. The xml:id is then used to establish the reference from posts to participants.

Not all participants in a CMC discourse are necessarily humans. Introduction of automatic chatbots is unproblematic in the adopted framework as they differ from human participants only in their metadata properties but not in their formal behaviour in discourse. However, many mediating systems are able to generate messages or parts of messages on their own, e.g. to indicate that a participant entered or left a chat room or by automatically providing time stamps or signatures in posts. This behaviour of the mediating system is typically triggered by specific actions of the discourse participants. To account for automatically generated parts of messages, an additional attribute @auto with a binary value domain (true, false) was introduced. By combining @who with @auto it becomes possible in principle to model different scenarios of human-machine interaction, including phenomena such as automatic correction of words during typing or the substitution of textual emoticons by their graphical equivalents (@who="HUMAN_PARITICIPANT", @auto="true").

3.1.3 Best practice for CMC: Modelling further aspects of posts in TEI

As can be seen in the example of a post in TEI in Listing 1, certain aspects of a post are modelled using available TEI attributes and elements: The creator of a post is given in the @who attribute, which contains a pointer to the creators entry (<person> element in the participant description in the metadata). Similarly, the posting time (extracted from the timestamp) is given through the reference in the attribute @synch which refers to a point in the

Listing 1: A post element and its annotations

```
<post xml:id="m645" who="#A02" synch="#t058" type="standard" auto="false">
 <note auto="true" who="#A02">for all</note>
 <anchor type="sentence_start"/>
 <ref type="addressingTerm" corresp="#A27">
  <w xml:id="m645.t1" type="ADV" lemma="nun">nun</w>
  <w xml:id="m645.t2" type="VVFIN" lemma="bitten">bitte</w>
  <w xml:id="m645.t3" type="NE" lemma="[_FEMALE-STUDENT-A27_]">[_FEMALE-STUDENT-A27_]</w>
  <w xml:id="m645.t4" type="$." lemma="!">!</w>
 </ref>
 <time> 16:48 </time>
</post>
```

timeline in the metadata section. Note that a timestamp as part of the text is represented in a <time> element, and the string indicating the private/public mode as shown in the original message is annotated by the <note> element. (Similarly, a signature stamp as e.g. used in Wikipedia discussions, would be represented in a <signed> element.) In accordance with the TEI Guidelines, the tokens in our chat corpus (derived from the tokenisation of the Saarland tagging pipeline, cf. Section 3.2.1, are represented by <w> elements. For the inclusion of token-related PoS analyses (including lemma information), there are two basic options offered the TEI by the TEI P5 Guidelines (ch. 17): as inline annotations, i.e. in attributes of <w>, or alternatively, as standoff annotations using the @ana attribute indicating span or feature structure elements elsewhere that contain the analysis. In this project we chose the first method. At <w>, the @lemma attribute contains the lemmatisation info, and the @type attribute contains the POS, see Listing 1. For occurrences of nicknames, chat room names, and addressingTerms, which had been marked up in the original ChatXML, we used the TEI <name> and <ref> elements, with a set of suitable values of their @type attribute ('roomname', 'nickname', 'addressingTerm', and 'url'). In a similar vein, we have introduced many more usage conventions for regular TEI elements and their attributes for the encoding of CMC phenomena.

3.1.4 Best practice for CMC: Metadata

In contrast to the customisations needed for the markup of the primary discourse data, we did not modify the existing TEI metadata model. All metadata provided in the original version of the corpus could be modelled using their TEI equivalents within the teiHeader. Special attention was paid to the modeling of a text classification scheme which is associated with the texts by means of the TEI's generic textClass/catRef mechanism. This model can be easily extended to a broader range of text and/or discourse properties to account for more detailed classifications, such as the one proposed by Herring (2007).

However, the TEI guidelines for metadata modeling are currently unable to account for crucial information about properties of the (software) system used to mediate the communication. There are very few means to informally describe the recording equipment used. For CMC systems, a fine-grained formal description of their properties is highly desirable to trace the system's influence on the discourse, especially in large and heterogeneous CMC corpora, possibly comprising multi-modal and/or multi-channel communication. Due to the rapid evolution of CMC systems, it will be difficult for future researchers to take into account relations among the properties and modes of use of a CMC system and properties of the discourse constructed using this system (e.g. communication channels available vs. actually used, automatic transformations of participants' utterances, exact time delays between utterances and their receptions etc.). The discussion of solutions to this problem will be taken up by the TEI special interest group on CMC.

The final CLARIN TEI schema for modeling CMC data according to the solutions developed in our project is publicly available in the form of a documented ODD customisation on the public website of the TEI special interest group on CMC[3].

3.2 Linguistic annotation

Linguistic annotation of the corpus comprised tokenisation, lemmatisation, and part-of-speech (PoS) tagging. While the original ChatXML resource already included annotations for selected CMC phenomena such as emoticons, interaction words, nicknames and addressing terms, one goal of the curation project was to systematically add a layer with PoS annotations in order to extend the

[3] http://wiki.tei-c.org/index.php?title=SIG:CMC/clarindschema

possibilities for linguistic queries.

For this purpose, we used the STTS-IBK tag set ('STTS 2.0') from the GSCL shared task on automatic linguistic annotation of CMC and SM genres (EmpiriST2015[4]) which had been defined as a result from discussions in the DFG scientifc network Empirikom[5] and in the context of three workshops dedicated to the adaptation and extension of the canonical version of the Stuttgart-Tübingen-Tagset STTS (Schiller et al., 1999) to the peculiarities of "non-standard" genres (cf. the volume Zinsmeister et al. (2013)). STTS-IBK is a customisation of the canonical STTS version as it introduces two types of new tags: (1) tags for phenomena which are specific for CMC and social media discourse, (2) tags for phenomena which are typical of spontaneous spoken language in colloquial registers. The resulting tag set is still backwards compatible with STTS (1999) and therefore allows for interoperability with other corpora that have been tagged with STTS. In addition, the tag set extensions defined in STTS-IBK are compatible with the extensions used at the IDS for the PoS annotation of FOLK, the Mannheim "Research and Teaching Corpus of Spoken German"[6] (Westpfahl, 2014). The tag set is described in an annotation guideline (Beißwenger et al., 2015a) and has been tested with data from several CMC genres in advance. A tabular overview of tags which have been added to the STTS in STTS 2.0 is given in Beißwenger et al. (2015b).

The linguistic preprocessing of the corpus was done in two steps: (1) an automatic step using a toolchain developed at Saarland University (including a basic sentence annotation, tokenisation, PoS and lemma annotation) and (2) a manual step in which the PoS tags resulting from step 1 were post-edited and made compatible with STTS_IBK by two human annotators, cf. Figure 1.

3.2.1 Automatic annotation

The automatic step was carried out by the team of the chair for computational linguistics at Saarland University using the tools for sentence segmentation, tokenisation, PoS tagging and lemmatisation developed in the BMBF project *Schreibgebrauch*[7] and described in (Horbach et al., 2014). These tools were already adapted to the processing for chat and forum data. For the PoS layer they had been trained for assigning the categories of the draft version of STTS 2.0 described in (Bartz et al., 2013) plus some additional categories defined by the developers at Saarland University (tag set STTS 2.0-beta). The result of this automatic tagging process was represented in an extended ChatXML format including token, lemma and PoS information (Tagged ChatXML).

Specific tags in STTS 2.0-beta	Target tags in STTS 2.0
AW	AKW
AWIND	$(
ERRAW	XY
ERRTOK	XY
PROAV	PAV

Table 1: Mapping from STTS 2.0-beta to STTS 2.0

3.2.2 Post-editing of the PoS results

Post-editing included (1) an upgrade of the PoS annotations resulting from step 1 to the STTS 2.0 tag set as described in (Beißwenger et al., 2015b) and, (2) a manual correction of tagging errors in the results from step 1 for a sample of parts of ten chat logfiles, comprising 4,339 tokens altogether.

The manual post-editing of the tagged ChatXML was carried out using the normalisation editor OrthoNormal in FOLKER from the FOLK-Tools Suite (Schmidt, 2012), which was originally developed for the manual normalisation and correction of PoS-tagged spoken language transcripts in the IDS FOLK corpus. For this purpose, Thomas Schmidt (IDS) provided an import and export interface for PoS-tagged ChatXML as part of FOLKER (version 1.2).

In work package (1), the upgrade of the tags used by the Saarland toolchain to STTS 2.0, we mapped specific tags from the Saarland tag set to tags in our target tag set (Table 1). On this basis, all occurrences of the tags in the left column were replaced by the tags in the right column.

Work package (2), the manual correction of tagging errors in the results from step 1, was done independently by two annotators who had been trained on assigning the STTS 2.0 categories beforehand. Based on the EmpiriST2015 guidelines for PoS tagging CMC (Beißwenger et al., 2015a), both annotators checked the PoS tag for each token in a sample comprising approximately 1,000 tokens of data from each of the four top-level text classes of the corpus (social chat, advisory chat, chat in

[4]https://sites.google.com/site/empirist2015/ and cf. Beißwenger et al. (2016)

[5]http://www.empirikom.net

[6]http://agd.ids-mannheim.de/folk.shtml

[7]http://www.schreibgebrauch.de

the context of learning, chats in the media context, N=4,339 tokens in ten different logfiles). The tagging results of the two annotators were used for calculating Cohens Kappa ($\kappa = 0.92$). The cases were the annotators had assigned different PoS tags (N=347) were extracted from the ChatXML and presented to the project leaders with a context size of one user posting per token. The project leaders decided the differing cases; on the basis of these decisions, we created the final version of the PoS-tagged ChatXML sample.

An evaluation of the 347 cases in which the tags of the two annotators differed showed that 25,9% of all cases (N=90) could easily be solved with additional restrictions for the use of tags from the canonical STTS (especially of tags for punctuation); the lions share of the remaining cases concerns the distinction between adverbs, modal and gradation particles. Based on these results, further specifications about assigning the STTS 2.0 categories for modal and gradation particles were added to the annotation guidelines.

3.3 ChatXML to TEI Conversion

We implemented a "ChatXML2TEI" XSLT stylesheet to convert the chat documents in Chat Corpus 1.2 (cf. Figure 1), including all metadata and image references, to the CLARIN CMC-TEI format as described in Section 3.1. We also implemented a wrapper script to generate a containing <teiCorpus> element with an appropriate <teiHeader>, combining all the individual chat documents in one large TEI corpus file. The result is the Chat Corpus 2.0 beta, TEI-structured, as indicated in Figure 1.

For quality assurance, we generated a log file of the conversion process, logging e.g. image references, nicknames not matched in the participant list, unusual timestamp formats, unusual element configurations, and the like, and checked it carefully, modifying the XSLT if necessary. We also performed a "primary data diff", i.e. we checked that the raw text contained in the ChatXML files was identical to the raw text of the resulting TEI files, to ensure that the conversion was complete in every case. The single TEI chat files and the combined TEI large corpus file were successfuly validated against the CLARIN CMC-TEI RNG schema using the jing validator[8].

[8] http://www.thaiopensource.com/relaxng/jing.html

NE category	Meaning
PER	Person
ORG	Organisation
LOC	Geographic location
GPE	Geopolitical entity
OTH	Other

Table 2: NE categories according to Telljohann et al. (2004)

3.4 Anonymisation

The obtained legal opinion (cf. Section 2.2) confirmed what is generally known about linguistic data and personality rights: Elements of the data that can be connected to a person (either a chat participant or mentions of a chat-external person) by any means likely reasonably to be used must be anonymised before the data can be published. In practice, this means that linguistic units such as names of persons, places, organisations, but also referring expressions such as URLs or email addresses, including parts that occur only in the metadata such as chatroom names or platform names, need to be obscured. Even indirect sensitive references, such as mentions of the rare hobby of a person, should be anonymised. However, names of politicians and celebrities such as "Sabine Christiansen" (the name of a political talk show host) need not be removed. In order that a corpus can still be reasonably used by linguists after anonymisation, it is recommended that such references are not simply removed but *categorised*, i.e. replaced with a placeholder string expressing the category of the element that has been replaced or even, when more effort can be invested, *pseudonymised,* i.e. "replacing a reference with a variant of the same type", cf. (Medlock, 2006). In the present project, we realised anonymisation as categorisation. Since most of the elements to be anonymised in the chat corpus are names, we used the named entity class set that was used in the Tüba-D/Z treebank (Telljohann et al., 2004), see Table 2.

Since the five categories of this set are rather broad, and in some cases the annotation of the original Chat Corpus 1.0 contained more specific information, we extended the set by the categories NICKNAME (subcategory of PER), and ROOMNAME. The majority of the chat nicknames mentioned is connected via @who or @corresp to the list of creators given in the <particDesc> of the TEI header, so wherever possible we used this entry to derive a more meaningful replacement string,

consisting of a.) the info in the @sex attribute of the participant (i.e. the corresponding <person> in participant description in the TEI header), if available; b.) the info in the @role attribute of the participant, if available, or, if unavailable, the string 'PARTICPANT'; c.) the @xml:id of the participant. The fancy replacement strings such as "FEMALE-TEACHER-A08" were used as replacements in the primary textual data, and in the @lemma and @normal (normalised form) attributes of the <w> elements. A result of this anonymisation procedure can also be seen in Listing 1.

Apart from the content of <name>, we use the the NE replacement categories also for the content of <ref type=addressingTerm>. We have defined further replacement strings for other types of references, e.g. 'WWWURL' and 'EMAIL' for mentions of URLs, or email addresses, respectively.

Anomymisation (i.e. replacing occurrences of names and similar references by the replacement strings described above) was performed in two steps (see Fig. 1):

1. **Automatic** anonymisation using an XSLT stylesheet that operated on the names that had already been annotated and in most cases linked to the creator list of the original resource (using the TEI elements and attributes <name>, <ref>, @who, @corresp, and the <person>s in the header's <particDesc>).
2. **Manual** anonymisation of the remaining occurrences of names that had not been annotated in the source, or that could not be matched in the participant list by the automatic procedure. – However, note that this time-consuming process has only been completed for the tokenised, normalised, and PoS-tagged subset of ten logfiles described in Section 3.2 so far.

4 Result: CLARIN-conformant Resource

The resulting resource is dubbed Dortmund Chat Corpus 2.0, and it contains 470 chat logfiles, containing 131,033 posts, containing 1,005,166 tokens altogether. The file (pretty printed XML) has a size of 100MB. The Dortmund Chat Corpus 2.0 will be ingested in TEI format into the CLARIN repositories at the IDS[9] and the BBAW [10]. At IDS, the chat corpus will become a corpus within the German Reference Corpus archive DeReKo and as such will be integrated in the corpus query platform COSMAS II[11], at BBAW, the corpus will be integrated in the corpus query platform DWDS (Digital Dictionary of the German Language[12]) as of autumn 2016. In addition, access will be provided to it through CLARINs federated content search, e.g. for NLP toolchains such as WebLicht.[13] However, the resource will be fully accessible and downloadable for academic use only when it is completely anonymised. Its complete anonymisation is currently undertaken as a separate effort.

5 Conclusion and prospects

Compared with the previous version of the resource, the Chat Corpus 1.0, the CLARIN-integrated version Chat Corpus 2.0 will allow for advanced queries using the additional linguistic annotations (sentences, tokens, PoS, lemmas). Due to the remodeling of the resource in TEI and the compatibilty of the PoS annotations with STTS, the corpus will be interoperable with other TEI-/STTS-annotated language resources. The integration in the CLARIN-D corpus infrastructures at BBAW and IDS will facilitate the comparative analysis of the chat corpus with the BBAW and IDS text and speech corpora. These features will not only increase the value of the resource for language-centered CMC research and variational linguistics but also the possibilities to use it in language teaching and higher education. Last but not least, the schemas, guidelines and best practices developed in the project which are all documented online will be useful resources for the curation of other CMC and SM corpora and their integration in the CLARIN infrastructure. The produced gold standard with PoS-tagged chat data may be used as an additional resource for the further adaptation of NLP tools to the peculiarities of CMC and SM data and corpora. It is planned to apply the pipeline described in this paper (Figure 1) for the remodeling, preprocessing and integration of further CMC and SM corpora in CLARIN in the near future.

[9] https://repos.ids-mannheim.de/
[10] http://clarin.bbaw.de/en/repo/
[11] http://cosmas2.ids-mannheim.de/
[12] http://www.dwds.de/
[13] https://weblicht.sfs.uni-tuebingen.de/weblicht/

References

Thomas Bartz, Michael Beißwenger, and Angelika Storrer. 2013. Optimierung des Stuttgart-Tübingen-Tagset für die linguistische Annotation von Korpora zur internetbasierten Kommunikation: Phänomene, Herausforderungen, Erweiterungsvorschläge. *Journal for Language Technology and Computational Linguistics. Special edition "'Das STTS-Tagset für Wortartentagging – Stand und Perspektiven"', edited by Heike Zinsmeister, Ulrich Heid & Kathrin Beck*, 28(1):157–198. http://www.jlcl.org/2013_Heft1/7Bartz.pdf.

Michael Beißwenger, Maria Ermakova, Alexander Geyken, Lothar Lemnitzer, and Angelika Storrer. 2012. A TEI Schema for the Representation of Computer-Mediated Communication. *Journal of the Text Encoding Initiative*, Issue 3. http://jtei.revues.org/476 (DOI: 10.4000/jtei.476).

Michael Beißwenger, Thomas Bartz, Angelika Storrer, and Swantje Westpfahl, 2015a. *Tagset und Richtlinie für das Part-of-Speech-Tagging von Sprachdaten aus Genres internetbasierter Kommunikation. Guideline-Dokument aus dem Projekt "'GSCL Shared Task: Automatic Linguistic Annotation of Computer-Mediated Communication / SocialMedia"' (EmpiriST2015)*. https://sites.google.com/site/empirist2015/home/annotation-guidelines.

Michael Beißwenger, Eric Ehrhardt, Andrea Horbach, Harald Lüngen, Diana Steffen, and Angelika Storrer. 2015b. Adding Value to CMC Corpora: CLARINification and Part-of-Speech Annotation of the Dortmund Chat Corpus. In *Proceedings of the 2nd Workshop on Natural Language Processing for Computer-Mediated Communication / Social Media (NLP4CMC2015)*, pages 12–16, Essen. https://sites.google.com/site/nlp4cmc2015/program.

Michael Beißwenger, Sabine Bartsch, Stefan Evert, and Kay-Michael Würzner. 2016. EmpiriST 2015: A Shared Task on the Automatic Linguistic Annotation of Computer-Mediated Communication and Web Corpora. In *Proceedings of the 10th Web as Corpus Workshop (WAC-X) and the EmpiriST Shared Task*, volume W16-26 of *ACL Anthology*, pages 44–56. Association for Computerational Linguistics, Stroudsberg.

Michael Beißwenger. 2013. Das Dortmunder Chat-Korpus. *Zeitschrift für germanistische Linguistik*, 41(1):161–164. (Extended version online: http://www.linse.uni-due.de/tl_files/PDFs/Publikationen-Rezensionen/Chatkorpus_Beisswenger_2013.pdf).

Thierry Chanier, Céline Poudat, Benoit Sagot, Georges Antoniadis, Ciara Wigham, Linda Hriba, Julien Longhi, and Djamé Seddah. 2014. The CoMeRe corpus for French: Structuring and annotating heterogeneous CMC genres. *Journal of language Technology and Computational Linguistics*, 29(2):1–30. http://www.jlcl.org/2014_Heft2/1Chanier-et-al.pdf.

CLARIN-D AP 5. 2012. CLARIN-D User Guide. http://de.clarin.eu/de/hilfe/benutzerhandbuch.

Susan C Herring. 2007. A Faceted Classification Scheme for Computer-Mediated Discourse. *Language@Internet*, 4(1):1–37. http://www.languageatinternet.org/articles/2007/761.

Andrea Horbach, Diana Steffen, Stefan Thater, and Manfred Pinkal. 2014. Improving the Performance of Standard Part-of-Speech Taggers for Computer-Mediated Communication. In *Proceedings of KONVENS 2014*, pages 171–177.

Ben Medlock. 2006. An introduction to nlp-based textual anonymisation. In *Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC 2006)*, pages 1051–1056.

Anne Schiller, Simone Teufel, Christine Stöckert, and Christine Thielen. 1999. Guidelines für das Tagging deutscher Textcorpora mit STTS (Kleines und großes Tagset). Technical report, Universität Stuttgart, Institut für Maschinelle Sprachverarbeitung. http://www.sfs.uni-tuebingen.de/resources/stts-1999.pdf.

Thomas Schmidt. 2012. EXMARaLDA and the FOLK tools – two toolsets for transcribing and annotating spoken language. In *Proceedings of the Eight conference on International Language Resources and Evaluation (LREC12)*, pages 236–240. European Language Resources and Evaluation (ELRA). http://www.lrec-conf.org/proceedings/lrec2012/pdf/529_Paper.pdf.

Jasmin Schröck and Harald Lüngen. 2015. Building and Annotating a Corpus of German-Language Newsgroups. In *Proceedings of the 2nd Workshop on Natural Language Processing for Computer-Mediated Communication / Social Media (NLP4CMC2015)*, pages 17–22, Essen. https://sites.google.com/site/nlp4cmc2015/program.

Heike Telljohann, Erhard Hinrichs, and Sandra Kübler. 2004. The Tüba-D/Z Treebank: Annotating German with a Context-Free Backbone. In *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC 2004*, pages 2229–2235.

Swantje Westpfahl. 2014. STTS 2.0? Improving the Tagset for the Part-of-Speech-Tagging of German Spoken Data. In Lori Levin and Manfred Stede, editors, *Proceedings of LAW VIII – The 8th Linguistic Annotation Workshop*, pages 1–10, Dublin, Ireland. Association for Computational Linguistics and Dublin City University. http://www.aclweb.org/anthology/W14-4901.

Heike Zinsmeister, Ulrich Heid, and Kathrin Beck, editors. 2013. *Das STTS-Tagset f/"ur Wortartentagging – Stand und Perspektiven*. Themenheft, Journal for Language Technology and Computational Linguistics 28(1). http://www.jlcl.org.

Annotation of Lexical Cohesion in English and German: Automatic and Manual Procedures

Jose Manuel Martinez Martinez
Universität des Saarlandes

Ekaterina Lapshinova-Koltunski
Universität des Saarlandes

Kerstin Kunz
Universität Heidelberg

kerstin.kunz@iued.uni-heidelberg.de
{e.lapshinova,j.martinez}@mx.uni-saarland.de

Abstract

The present paper describes procedures to annotate lexical cohesion in GECCo, a corpus of English and German texts that includes both written and spoken data. Lexical cohesion is an important linguistic component of meaningful discourse and contributes to the overall coherence and thematic continuity of a text. Aiming at a highly precise, fine-grained annotation and avoiding time-consuming procedures, we combine automatic and manual annotation procedures. In this paper, we present the main concepts underlying the annotation and outline the encoding scheme that we apply. We describe the annotation principles and the classification of the sense relations included in our scheme. We also present both automatic and manual procedures and evaluate them in terms of their performance and inter-annotator agreement.

1 Aims and Motivation

This paper describes the annotation of lexical cohesion in GECCo, a corpus of English and German texts that includes both written and spoken data. Lexical cohesion is one of the major types of cohesion contributing to the overall coherence and thematic continuity of a text. It therefore is an important linguistic component of effectively organised and meaningful discourse.

Our overarching goal is an empirical analysis of the realisation of cohesive strategies in English and German and also in written and spoken registers. For this reason, one of the major challenges is defining fine-grained categories that permit the identification of commonalities and differences in terms of various cohesive aspects across the languages and registers under analysis.

As our interest lies in the linguistic properties of lexical cohesion, another challenge is to obtain a highly precise annotation without wasting to much time and labour. Therefore, we start the annotation process with semi-automatic procedures that help to identify candidates of lexical chains and assign their semantic relations. For the sake of convenience, this annotation step was performed on the English texts only. We then proceed with the manual annotation of the English texts. On the one hand, this provides us with a precise annotation of lexical cohesion, and on the other hand, it allows us to test and evaluate the automatic procedures. As the evaluation results indicate unsatisfactory performance of the automatic procedures, we decide to apply only manual annotations for the German texts. In the final step, we evaluate the manual annotation of both English and German texts by calculating inter-annotator agreement. Both automatic and manual procedures are evaluated at three levels: 1) candidate identification, 2) chain construction, and 3) sense relation assignment.

The paper is structured as follows. We provide the theoretical background and state of the art in Section 2 and describe the principles underlying and the categories included into our annotation scheme in Section 3. The annotation procedures are described in Section 4, and their evaluation is presented in Section 5. In Section 6, we summarise and discuss our results.

2 Theoretical Background

Lexical cohesion is regarded as one major type of cohesion contributing to the overall coherence and thematic continuity of a text. The concept was introduced by Halliday and Hasan (1976), whose main focus was on textual relations between linguistic expressions beyond the level of the clause. Halliday and Hasan posit lexical cohesion alongside four other major types of cohesion: coreference, substitution, ellipsis and conjunction. As

illustrated by example (1), lexical cohesions differs from them in terms of **structure** and **semantics**.

(1) *I live in a town called Reigate. It's between London and the countryside which is quite nice. It takes us about 25 minutes to get to London on the train. I say it's a town, it's more of a village. It's quite small. It's very nice actually, it's a nice place to live. And I grew up in a place called Banstead which is fairly close to Reigate.*

2.1 Structure

Contrary to Halliday and Hasan's other four types, the cohesive devices signalling a relation to other expressions in the text are not grammatical items such as proforms, determiners or conjunctions. As the term suggests, the cohesive relation is triggered by lexis, as between *village*, *town* and *place* in example (1). The focus of our project is on the extraction and annotation of nominal elements, although Halliday and Hasan also include relations between verbs, adjectives and adverbs.

2.2 Semantics

The conceptual relation set up by lexical cohesion differs from co-reference and also from what is called bridging in the literature. **Co-reference** and **bridging** are both based on information **instantiated** in the text, the former evoking a relation of identity and the latter a relation of similarity between individual referents in the same text. Our concept of lexical cohesion concerns context-free **sense relations** such as meronymy, hyponymy, synonymy, as described in Lyons (1977) or Winston and Herrmann (1987). Hence what is created from a semantic or conceptual perspective is a relation of similarity between **types** of referents, see also Tanskannen (2006) and Berzlanovich (2008). We also account for **cohesive chains**, which span all nominal elements belonging to the same semantic field (see below). Quite often, devices of lexical cohesion are preceded by co-referential devices, such as the definite article or demonstrative determiners. The interaction of co-reference and lexical chains is assumed to be a major indicator of coherence (Hasan, 1985a; Hasan, 1985b; Martin, 1992). This interaction is left aside here, although it was demonstrated, for instance, by Kunz et al. (2016).

2.3 State of the art in annotation of lexical cohesion

Lexical chains have often been used in natural language processing to solve tasks like text summarization (Doran et al., 2004), or forum thread linking (Wang et al., 2011). However, fewer proposals have tried to use such chains for the study of lexical cohesion (see Teich and Fankhauser (2005) or Bartsch et al. (2009)).

According to Teich and Fankhauser (2004), an automatic lexical chain builder is desirable to reduce human effort devoted to lexical chain annotation and to obtain more consistent results.

Most automatic algorithms rely on either thesauri (Doran et al., 2004; Wang et al., 2011; Fankhauser and Teich, 2004) or statistical associations between words (Wang et al., 2011). The former approach allows one to create not only chains but also to establish various types of semantic relations at the cost of a recall, which is limited to the coverage of the thesaurus.

The annotation of lexical chains is a complex task and so is the operationalization of its evaluation. Some authors (Wang et al. (2011) and Doran et al. (2004)) assess the quality of their techniques to automatically produce lexical chains using an extrinsic approach. This is to test the performance of the system in terms of improving an extrinsic task (forum thread linking and text summarization, respectively). Teich and Fankhauser (2004) carry out an intrinsic evaluation on the methodology used under a purely linguistic point of view, comparing the output of their system with a manually annotated gold standard. However, their evaluation is qualitative.

Further works on lexical cohesion related to natural language processing include Morris and Hirst (1991) and Barzilay and Elhadad (1999).

There exist some works focusing on the comparison of automatic and manual annotations. For instance, Hollingsworth and Teufel (2005) present an approach to directly evaluate the quality of lexical chains, in comparison to a human gold standard. This approach differs from previous evaluation efforts which adopted extrinsic methods relying on word sense disambiguation or on the final application result (the summary or the text segmentation), rather than the focusing on the properties of the lexical chains themselves. The authors also perform a meta-evaluation to compare the best of the metrics used for the evaluation.

3 Annotation Scheme

In order to guarantee consistency throughout the whole process of annotation, detailed descriptions and disambiguation rules had to be defined. They concern the segmentation of nominal elements, the textual distance allowed between nominal elements, the account of different word senses, the classification of the type of sense relation between two nominal elements, and the grouping of several nominal elements in one lexical chain. We can only provide an overview in this paper, details can be found in our annotation guidelines (Kunz, 2014).

Segmentation A nominal element may consist of one noun only, or it may be a compound such as *private teacher*, or a term pattern, such as *head of faculty*. Annotations are based on entries in standard dictionaries (e.g. Cobuild[1] or Longman[2] for English and DWDS[3] for German).

Distance Sense relations are always analysed in linear order, between the two closest elements in a lexical chain. According to Halliday and Hasan's concept, only relations between nominal elements in different clauses, clause complexes, or larger textual passages are cohesive. This would imply that the sense relation between *town* and *Reigate* in example (1) in the first clause is not cohesive but the relation between *town* and *London*. We however decide to annotate all adjacent elements within and outside clause boundaries in order to enhance research on intra- and inter-clausal relations with the help of additional annotation layers available in the corpus.

Word sense If one nominal element occurs more than once in a text and refers two different semantic concepts, and if each of these occurrences enter into a relation to other nominal elements (e.g. *bank* and *financial institution*; *bank* and *building*), two separate lexical chains are established. The assignment of semantic relations follows those defined in WordNet (Fellbaum, 1998) for English, and DWDS for German. The latter integrates GermaNet (Hamp and Feldweg, 1997; Henrich and Hinrichs, 2010) and OpenThesaurus (Naber, 2005).

Sense relations We include the following sense relations:

- repetition: orthographical repetition of nominal expressions such as *London* and *London*, or *place* and *place* in example (1) above. In case of compounding, the second element is the determining factor (*stem cell* and *pluripotent cell*, but not *stem cell research* and *stem cell maintenance*).
- antonymy: relation of contrast, as with *inflation* and *deflation*
- synonymy: total synonymy but also near synonymy, such as between technical and common-language terms (e.g. *belly* and *abdomen*).
- hyperonymy: in case the superordinate term follows the more specific term as with *village* and *place*
- hyponymy: in case the specific term follows the superordinate one
- co-hyponymy: between two elements on the same level of specification, such as *town* and *village*
- holonymy: relation, where the whole follows the part (e.g. *quarter* and *town*
- meronymy: part-whole relation, where the part follows the whole (e.g. *town* and *quarter*)
- co-meronymy: succession of two parts that belong to a whole (e.g. *square* and *quarter*).
- type: relation between a common noun and a named entity (e.g. *place* and *Reigate*).
- instance: relation where the named entity follows the common noun (e.g. *Reigate* and *town*).
- co-instance: relation between two named entities (*Reigate* and *London*)

Lexical chains A nominal element can be assigned to a lexical chain if its word sense matches all other elements in the chain, i.e. if one of the types of relations described above could be assigned to the nominal element and each of the other elements in the chain. As a consequence, one nominal element may be a part of several different lexical chains in the same text. See example (2) and the paragraph **Sense relations** above for further discussion.

[1] http://dictionary.reverso.net/english-cobuild
[2] http://www.ldoceonline.com
[3] http://www.dwds.de

4 Annotation Procedures

4.1 Data

The data under analysis includes English and German texts that belong to a variety of registers on a continuum from written to spoken discourse (understood as a sub-dimension of register variation under mode of discourse). The written subcorpus was extracted from the CroCo corpus (Hansen-Schirra et al., 2012), and the spoken subcorpus – from the spoken part of GECCo (Lapshinova-Koltunski et al., 2012).

The registers and the size (in tokens) of annotated subsets are listed in Table 1. ESSAY (political essays) and POPSCI (popular scientific texts) represent written discourse, INTERVIEW (transcribed interviews on various topics) represents spoken discourse, whereas FICTION (fictional texts) contains spoken passages in the form of dialogues and is, in this way, on the borderline between spoken and written registers.

	EO		GO	
register	**texts**	**tokens**	**texts**	**tokens**
ESSAY	23	27171	20	31407
FICTION	10	36996	10	36778
INTERVIEW	9	30057	12	35036
POPSCI	8	27055	9	32639
TOTAL	50	121279	51	135860

Table 1: Information on the corpus size per register

Further annotation layers available in this corpus data include tags on parts of speech, chunks, clause and sentence boundaries, cohesive devices (cohesive reference, conjunction, substitution, ellipsis) triggering coherence in a text, and also chains of relations (for co-reference and ellipsis). The procedures for the annotation of cohesive devices are described by Lapshinova-Koltunski and Kunz (2014).

4.2 Automatic annotation procedure

As starting point for our automatic procedures we used the Little Cohesion Helper (LCH)[4], a piece of software written in Python inspired by Fankhauser and Teich (2004). The authors introduced constraints to filter relevant ties related to WordNet (distance of a word from a root in the WordNet, kind of semantic relationship, minimum depth, etc.) and the text (distance between two words in terms of number of intervening sentences and parts of speech), and the chain themselves (maximum length). Similar strategies are reported by Doran et al. (2004) to weight relations (kind of semantic relationship, and type of match for repetitions – exact, partial, fuzzy) and to discard irrelevant chains (length as number of members in the chain, homogeneity –type-token ratio of chain members–, number of repetitions and type of WordNet relation).

A simplified and schematic expression of the typical algorithm to build chains based on thesaurus look-ups is provided in Figure 1.

The original script takes as input a plain text file using NLTK (Bird et al., 2009) to process the text. First, it tokenizes and splits the text into sentences with `Punkt Tokenizer` (Kiss and Strunk, 2006). Second, it adds POS annotation with `Unigram Tagger` trained on the Brown corpus. And third, it performs a semantic analysis with `WordNet` for all nouns which is the basis for building lexical chains. For each noun, all possible relations with other nouns are checked in reverse order of apparition in the text. This yields cohesive tuples for each word pair. Then, any of the two components of the cohesive tuple is checked as to whether it is already in an existing chain. If yes, the tuple is added to the chain, if not, a new chain is created including this tuple. If the tuple has no relation with the direct preceding word, a look up with other previous items is done, until it finds a related term, adding the information about this relationship. Finally, it saves the result as a MMAX2 project for subsequent manual revision (see Section 4.3 for more details).

We modified LCH with the following goals:

- to port it to Python 3
- to circumvent NLTK tokenization and POS tagging (since this information was already encoded and, more importantly, the original token stream had to be preserved to incorporate this new layer of annotation into the corpus)
- to use lemmas instead of word forms to increase recall using WordNet/GermaNet (specially in German)
- to identify WordNet's multi-word expressions in our texts to increase precision and recall
- to improve file handling and character encoding

[4] `http://lch.sourceforge.net`

- to improve generation of well-formed XML (MMAX2 projects)
- to restore the original token stream to integrate the annotation in the corpus

The final workflow is made up of three steps:

1. corpus preprocessing:
   (a) text boundary identification,
   (b) nominal MWE extraction from WordNet,
   (c) extraction of word forms, lemmata and POS tags for each text,
   (d) identification of WordNet's MWEs in texts.
2. annotation with LCH, for each text:
   (a) obtaining lemmas for nouns
   (b) identification of repetitions of unknown nouns (not found in WordNet)
   (c) extraction of all possible pairs representing semantic relations
   (d) building of lexical chains
   (e) generation of chain links (sorting by consecutive elements)
   (f) serialization of results as MMAX2 project.
3. project postprocessing:
   (a) restoring original tokenization
   (b) updating lexical cohesion annotation accordingly
   (c) replacing lemmas by their word forms.

We describe and discuss the evaluation of this annotation procedure in Section 5.1.

4.3 Manual annotation procedure

For the manual annotation of lexical cohesion in our data, we use MMAX2, a tool for manual annotation (Müller and Strube, 2006) facilitating this process. Texts are annotated by four human annotators with linguistic background. The annotation process consists of three main steps: (1) identification of the candidates for lexical chain members, (2) assignment of links between chain members, (3) assignment of sense relations to chain members.

Candidate identification For the texts in English, we partly keep the automatic pre-annotation of candidates for lexical chain members. However, we remove the sense relations to avoid the influence of automatic assignment on the decision of human annotators. The MMAX2 visualisation allows annotators to decide whether the candidates tagged by LCH belongs to a lexical chain.

As our annotation scheme includes nominal cohesion only, all nouns and noun phrases can be considered as candidates for chain members. However, our analyses show that not every nominal element is included into a lexical chain: 60,84% of all nouns in the English texts and 59,56% of all nouns in the German text are members of lexical chains. For this reason, we decide against the automatic annotation of all nouns as candidates.

Link assignment Human annotators not only identify members of lexical chains and assign their sense relations, but also link the chain members. The MMAX2 tool allows visualisation of links between two or more elements. The annotated information is then encoded as `<lexicalcohesion>` for every member. Each member (markable) is automatically provided with an identification number (ID). Every expression which belongs to the same lexical chain is also assigned to the same ID. This information is saved for every text, and then imported into the corpus. The information on the chains can then be extracted with the help of these IDs.

Sense relation assignment As mentioned above, we analyse the sense relations linking two adjacent chain elements. For this purpose, the type of relation is tagged on the second element of each link. For instance, *place* in example (1) is an hyperonym of the preceding nominal expression *village*, and *place* is a repetition of the preceding nominal expression *place*, and so on. The first element in every chain obviously has no sense relation.

The same word may belong to several lexical chains, and therefore may have several markables with different sense relation assignments. This is especially relevant for words within multiword expressions. For an illustration, see *broadcast industry* and *broadcast legislation* in (2-a) and (2-b), which are elements in long lexical chains.

(2) a. *and Ofcom who is the watchdog for the broadcast industry, to, instead of having it 10 per cent over 10 years, we*

reduce that to 10 per cent over 5 years. (...)

b. *I think that is built into broadcast legislation but it is not there for the cinema legislation, for film legislation. There is no film legislation. (...)*

The whole multiword expression *broadcast industry* in (3-a) is a member of the lexical chain *industry – broadcast industry – industry – industry – industry* tagged as a hyponym of *industry*. At the same time, the multiword expression *broadcast legislation* in (3-b) is also a member of another lexical chain with the head *legislation*: *legislation – broadcast legislation – cinema legislation – film legislation – film legislation – legislation.*

In the process of manual assignment of sense relations, human annotators rely on their intuition. However, they are also allowed to consult various resources to solve problematic cases, e.g. WordNet for English and DWDS, GermaNet and OpenThesaurus for German.

The information on the sense relation is also integrated into the structure `<lexicalcohesion>`, see Figure 5. In this example, the items indexed with 'set_49' belong to the "*legislation*" lexical chain mentioned above. The chain contains nine elements and starts with the word *chain* which is, however, outside the text span provided in Figure 5. *Broadcast legislation* is its hyponym, and *cinema legislation* is the co-hyponym of *broadcast legislation*, whereas *film legislation* is the co-hyponym of *cinema legislation*. The second mention of *film legislation* is a repetition. The other set (set_113) in the example in Figure 5 is represented by the lexical chain *UK – Europe – UK*, and is a case of holonymy-meronymy relations.

4.4 Annotation statistics

We summarise the statistics on the structures annotated for lexical cohesion in our data in Table 2. Whereas Table 3 provides statistics on the annotated relations classified per relation.

	EO	GO
nr of chains	2598	1783
nr of relations	11814	11568

Table 2: Manually annotated structures in GECCo

	EO	GO
repetition	6925	6191
hypernym	1046	1104
hyponym	1033	1159
synonym	579	608
co-hyponym	520	570
meronym	436	340
holonym	426	308
antonym	292	465
instance	190	238
type	175	203
co-instance	172	307
co-meronym	75	100
gennoun	1	3

Table 3: Manually annotated sense relations in GECCo

5 Annotation Evaluation

In the evaluation step, we compare automatic and manual annotations (for English texts only), as well as the annotations produced by different annotators (on a sample of English and German texts). The comparison is performed for the following features: 1) markables representing candidate identification, 2) chains representing link assignment, 3) semantic relations representing sense relation assignment.

Markables from both annotation versions are aligned on the basis of their token IDs. Each markable pair containing at least one token in common is considered a markable alignment. We use Jaccard distance to take into account perfect (all tokens in both markables were the same) and spurious (only some tokens were in common) agreement.

Chain alignment is done by retrieving the chain IDs of the markables aligned in the previous step. We consider a chain alignment any pair of chains having at least one markable in common. Upon identifying the alignments, chain members are retrieved. We use Jaccard distance again to take into account perfect (all markables in both chains are the same) and spurious (only some markables are shared across both chains) agreement.

To evaluate the assigned relations, we collect subsets of aligned markables which share the preceding member in a chain. If the condition is satisfied, the semantic relation assigned to the selected markable is compared. Since only one label is provided, we used binary distance to calculate the agreement. If the relation label is the same on both aligned markables, the agreement is 1, if they are

different agreement is 0.

For each level of analysis we provide the following measures: precision ($P = \frac{|M \cap A|}{|A|}$ where M is the reference dataset –for **M**anual– and A is the test –for **A**utomatic), recall ($R = \frac{|M \cap A|}{|M|}$), and the F-score ($F$, the harmonic mean of P and R, weighted by $\alpha = 0.5$) of the elements annotated by the automatic system, together with the Jaccard coeficient of similarity $J = \frac{I}{U}$, which accounts for the proportion of elements present in both data sets ($I = |M \cap A|$) over the total number of elements being compared ($U = |M \cup A|$). Moreover, we report on the level of agreement for the intersection of elements with the manual reference annotation (I) using Cohen's Kappa (κ) as implemented in NLTK `nltk.metrics.agreement` (Bird et al., 2009) and described by Artstein and Poesio (2008).

We plot a confusion matrix to visualize the quality of sense relation assignments produced by the automatic system (or by human annotators) in relation with a reference annotation. Such a plot depicts the prediction on the X axis (e.g. automatic annotation), and the reference on the Y axis (e.g. manual annotation). The resulting diagonal displays the instances where there is agreement, which means that the predicted label is equal to the true label. The off-diagonal cells represent mislabelling or disagreement. A diagonal with high values is an indicator of many correct predictions or a good agreement.

5.1 Automatic vs. manual procedures

We firstly report on the comparison of the output of the automatic procedures explained in Section 4.2 with its manual annotation. As previously mentioned, the automatic procedures were applied on the English subcorpus only. Table 4 summarises all the measures calculated to evaluate the quality and agreement of the annotation.

	markables	chains	relations
U	23832	5700	11884
I	11884	4089	3262
J	0.50	0.72	0.28
P	0.60	0.65	0.17
R	0.77	0.69	0.22
F	0.67	0.67	0.19
κ	0.90	0.38	0.47

Table 4: Evaluation measures for automatic annotation of lexical cohesion chains.

Markables A total of 23832 markables are compared (U), the intersection of markables present in both annotation sets (I) amounts to 11884 items, what represents a 50 % of them showing some kind of overlap (J). Precision ($P = 0.6$), recall ($R = 0.77$) and the F-score ($F = 0.67$) are low in comparison with the human performance (see Section 5.2). The agreement between both versions at markable level is $\kappa = 0.90$. However, if we extrapolate this measure to the total number of chain members annotated in both versions, the agreement sinks to a mere 45 %.

Chains A total of 5704 chains are compared (U). 72 % of the chains overlap (J). Precision ($P = 0.65$), recall ($R = 0.69$) and the F-score ($F = 0.67$) are lower than human performance. The agreement between both versions regarding the overlapping chains is $\kappa = 0.38$. This clearly indicates that chains share a very low proportion of members in common. If we extrapolate the agreement to the total number of chains, the agreement falls to 27 %.

Relations From the 11884 markables aligned across both versions (U), only 28 % of the markables refer to the same anteceding member in their respective chains (J)). Precision ($P = 0.17$), recall ($R = 0.22$) and the F-score ($F = 0.19$) are very low indicating that the internal arrangement of members within the automatically assigned chain is very different from the human reference. The agreement in the assignment of the relation labels is $\kappa = 0.47$. Nevertheless, this subset of relations represents just 14 % of all the relations annotated. If we extrapolate the agreement to the total number of relations annotated with both methods the agreement drops to 11 %.

The confusion matrix displayed in Figure 2 shows a very low precision of the automatic system (light shadowed diagonal). The automatic system seems to assign many repetitions to instances where humans chose other categories. This can be explained by the nature of the automatic procedures assigning repetitions to the nouns that are not covered by WordNet. Another influential factor is the difference in the subset of the relations used by the automatic system (antonym, holonym, hypernym, hyponym, meronym, synonym, repetition) and the one used by humans who had six additional relations at their disposal (co-hyponym, co-instance, co-meronym, instance, type).

5.2 Inter-annotator agreement in manual procedures

The inter-annotator agreement is calculated on a subset of texts containing 5925 tokens in English and 7102 in German roughly representing a 5 % of the manually annotated subcorpus. The proportion of lexical chains revised for both English (see Table 5) and German (see Table 6) also reaches a 5 %, what in turn amounts to around a 7 % of all sense relations.

	markables	chains	relations
U	1123	175	903
I	903	146	465
J	0.80	0.83	0.52
P	0.92	0.95	0.49
R	0.86	0.82	0.45
F	0.89	0.88	0.47
κ	0.94	0.59	0.62

Table 5: Evaluation measures for IAA in manual annotation of lexical cohesion chains for English.

	markables	chains	relations
U	1169	215	821
I	821	169	350
J	0.70	0.79	0.43
P	0.89	0.98	0.39
R	0.76	0.88	0.33
F	0.82	0.93	0.36
κ	0.97	0.55	0.63

Table 6: Evaluation measures for IAA in manual annotation of lexical cohesion chains for German.

Markables We compare a total of 1123 markables in English (U) of which 80% showed an overlap (J). Precision ($P = 0.92$), recall ($R = 0.86$) and the F-score ($F = 0.89$) are higher than the values for the automatic system. The agreement between both annotators at the markable level is $\kappa = 0.95$. If we extrapolate this measure to the total number of markables annotated in both versions, the agreement remains close to 75 %, which is much better than the IAA achieved with the automatic procedure.

As for German, a total of 1169 are compared (U) showing an overlap of 70 % (see J in the table). The agreement is $\kappa = 0.97$, but extrapolated to the total number of markables it goes down to about 68 %.

Chains A total of 175 chains are compared in English (U) showing an overlap of 83 % (J). Both annotators reached an agreement of $\kappa = 0.59$ for the overlapping chains. If the number is extrapolated for all chains, the agreement still reaches 48 %.

78 % of the 214 chains in German are overlapping. Their agreement amounts to $\kappa = 0.55$ for these subset of chains, what represents only 43 % of the agreement for the total number of chains.

Precision, recall and the F-score are very similar for both languages at this level.

Relations From the 903 markables aligned across both of annotators in English (U), 52 % refer to the same preceding member across chains (J). The agreement regarding the assignment of a semantic relation for these pairs is $\kappa = 0.62$. This subset of relations represents in turn 41 % of the total number of markables annotated. If we extrapolate this agreement to the total number of relations annotated by both annotators, the proportion of relations showing agreement amounts for 30 %.

The confusion matrix plotted in Figure 3 shows a fairly good agreement for most categories, except co-hyponyms competing with co-instances, and co-meronyms with co-hyponyms, as well as synonymy.

43 % of the 821 markables in the German sample aligned across both annotators refers to the same preceding member across chains. The agreement for these pairs is $\kappa = 0.63$. If we extrapolate this agreement to all the markables analyzed it decreases to 21 %.

The confusion matrix in Figure 4, enables the examination of the results broken down by semantic relations. We observe that the relations of antonym, co-hyponym, hyperonym, hyponym, repetition and synonym display a fairly good agreement. However, it is weaker for the rest of the relations as indicated by the lighter grey tones of the cells in the diagonal, and darker grey shadows out of the diagonal.

Taking into account all indicators, annotators of English texts show a slightly higher IAA than those of the German ones. This may be due to the higher number of repetitions in English than German, which can be more easily identified than e.g. relations of synonymy. Moreover, lower inter-annotator agreement in German may go along with a higher degree of lexical specification.

6 Conclusion and Discussion

In the present paper, we have provided an insight into the annotation procedures underlying our analysis of lexical cohesion in English and German spoken and written texts. Our initial approach included the combination of automatic and manual procedures. The automatic annotation procedure employs basic heuristics linking nouns and noun phrases greedily if a certain type of a link can be found in WordNet. In some respects, this is a high-recall strategy desirable in scenario combining an automatic pre-annotation and manual post-correction, which was originally our intention.

We have performed a thorough evaluation of the automatic annotation calculating IAA between the automatic system and the human annotators. The main challenges for the evaluation are unitisation issues hindering comparability (see Wacholder et al. (2014) for a similar scenario) and the complexity of assessing multiple annotation choices (candidate identification, chain membership, link assignment, and sense relation assignment) which are comprised in the task of building lexical chains. Our evaluation goes beyond previous intrinsic evaluations of this task.

Although the number of markables and chains seems to be similar in both datasets, the representation of lexical cohesion by means of lexical chains and the internal structure of the chains differs in automatic and human annotations, as shown in the evaluation of chains and relations. The performance of the automatic system presented in Section 4.2 is much lower than the human reference quantified in terms of precision, recall and IAA. These differences have an important effect on higher level dimensions such as topic development and overall semantic variation. Their correction turned out to be even more time consuming than a purely manual procedure. This was confirmed by the feedback provided by human annotators, who considered it much easier to build lexical chains and annotate relations from scratch than post-editing the system's output.

The evaluation of our manual procedures show that overall, we achieve a good IAA in the annotation of both German and English texts. The agreement scores however show that annotating lexical cohesion chains is a difficult task even for humans. Annotators showed a higher degree of agreement in English than in German across all levels of comparison. The challenge not only arises from the high conceptual level of the linguistic analysis but also from the complexity of the annotation which is made up of different subtasks.

Acknowledgments

This paper is based on work carried out in the frame of the GECCo[5] project funded by the German Research Foundation (DFG) under GZ STE 840/6-2 and KU 3129/1-2 *Kohäsion im Deutschen und Englischen – ein empirischer Ansatz zum kontrastiven Vergleich*.

References

Ron Artstein and Massimo Poesio. 2008. Survey Article Inter-Coder Agreement for Computational Linguistics. *Computational Linguistics*, 4(34):555–596.

Sabine Bartsch, Stefania Degaetano, Tomek Grubba, Nina Petrychka, David Sullivan, Christoph Tragl, and Claudio Weck. 2009. ObamaSpeeches.com Building and Processing a Corpus of Political Speeches. In Elke Teich, Andreas Witt, and Peter Fankhauser, editors, *Poster at Proceedings of GSCL Workshop: Linguistic Processing Pipelines.*, pages 41–42, Potsdam, sep.

Regina Barzilay and Michael Elhadad. 1999. Using lexical chains for text summarization. *Advances in automatic text summarization*, pages 111–121.

Ildikó Berzlanovich. 2008. *Lexical cohesion and the organization of discourse. First year PhD report*. University of Groningen, Groningen.

Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural Language Processing with Python*. O'Reilly Media Inc.

William Doran, Nicola Stokes, Joe Carthy, and John Dunnion. 2004. Comparing lexical chain-based summarisation approaches using an extrinsic evaluation. In P Soijka, K Pala, P Smrz, C Fellbaum, and P Vossen, editors, *Proceedings of the 2nd Global WordNet Conference, 20 - 23 January*, page 112, Brno, jan. Masaryk University.

Peter Fankhauser and Elke Teich. 2004. Multiple perspectives on text using multiple resources: Experiences with XML processing. In *Proceedings of LREC Workshop on XML-based richly annotated corpora, 4th Conference on Language Resources and Evaluation (LREC)*, pages 15–20, Lisbon, Portugal, may.

Christiane Fellbaum, editor. 1998. *WordNet: An Electronic Lexical Database*, volume 71. MIT Press, Cambridge, MA.

[5] http://www.gecco.uni-saarland.de

M.A.K. Halliday and Ruqaiya Hasan. 1976. *Cohesion in English*. Longman, London, New York.

Birgit Hamp and Helmut Feldweg. 1997. GermaNet - a Lexical-Semantic Net for German. In *Proceedings of ACL workshop Automatic Information Extraction and Building of Lexical Semantic Resources for NLP Applications*, pages 9–15, Madrid, jul.

Silvia Hansen-Schirra, Stella Neumann, and Erich Steiner. 2012. *Cross-linguistic Corpora for the Study of Translations. Insights from the Language Pair English-German*. de Gruyter, Berlin, New York.

Ruqaiya Hasan. 1985a. The structure of a text. In M.A.K. Hallday and R. Hasan, editors, *Text and context: aspects of language in a social-semiotic perspective*, pages 52–96. Oxford University Press, Oxford.

Ruqaiya Hasan. 1985b. The texture of a text. In M.A.K. Hallday and R. Hasan, editors, *Text and context: aspects of language in a social-semiotic perspective*, pages 70–96. Oxford University Press, Oxford.

Verena Henrich and Erhard Hinrichs. 2010. Gernedit – the germanet editing tool. In *The Seventh Conference on International Language Resources and Evaluation (LREC 2010)*, pages 2228–2235, Valletta, Malta.

Bill Hollingsworth and Simone Teufel. 2005. Human annotation of lexical chains: coverage and agreement measures. In *he ACM International Workshop on Methodologies and Evaluation of Lexical Cohesion Techniques in Real-World Applications (ELECTRA 2005) held at SIGIR 2005*, volume 39, New York, NY, USA. CM SIGIR Forum Homepage archive.

Tibor Kiss and Jan Strunk. 2006. Unsupervised multilingual sentence boundary detection. *Computational linguistics*, 32:485–525.

Kerstin Kunz, Ekaterina Lapshinova-Koltunski, and José Manuel Martínez Martínez. 2016. Beyond Identity Coreference: Contrasting Indicators of Textual Coherence in English and German. In *Proceedings of CORBON at NAACL-HLT2016*, San Diego, jun.

Kerstin Kunz. 2014. Annotation guidelines for lexical cohesion. , Universität des Saarlandes.

Ekaterina Lapshinova-Koltunski and Kerstin Kunz. 2014. Annotating cohesion for multillingual analysis. In *Proceedings of the 10th Joint ACL - ISO Workshop on Interoperable Semantic Annotation*, Reykjavik, Iceland, May. LREC.

Ekaterina Lapshinova-Koltunski, Kerstin Kunz, and Marilisa Amoia. 2012. Compiling a multilingual spoken corpus. In Tommaso Raso Heliana Mello, Massimo Pettorino, editor, *Proceedings of the VIIth GSCP International Conference: Speech and corpora*, pages 79–84, Firenze. Firenze University Press.

John Lyons. 1977. *Semantics*, volume 1–2. Cambridge University Press, Cambridge, UK.

James R. Martin. 1992. *English Text. System and Structure*. John Benjamins, Amsterdam, Netherlands.

Jane Morris and Graeme Hirst. 1991. Lexical cohesion computed by thesaural relations as an indicator of the structure of text. *Computational linguistics*, 17:21–48.

Christoph Müller and Michael Strube. 2006. Multilevel annotation of linguistic data with MMAX2. In Sabine Braun, Kurt Kohn, and Joybrato Mukherjee, editors, *Corpus Technology and Language Pedagogy: New Resources, New Tools, New Methods*, pages 197–214. Peter Lang, Frankfurt a.M., Germany.

Daniel Naber. 2005. Openthesaurus: ein offenes deutsches wortnetz. In *Sprachtechnologie, mobile Kommunikation und linguistische Ressourcen: Beitrge zur GLDV-Tagung 2005*, pages 422–433, Bonn. Peter-Lang-Verlag, Frankfurt.

Sanna-Kaisa Tanskannen. 2006. *Collaborating towards Coherence*. John Benjamins, Amsterdam, Netherlands.

Elke Teich and Peter Fankhauser. 2004. WordNet for lexical cohesion analysis. In P Soijka, K Pala, P Smrz, C Fellbaum, and P Vossen, editors, *Proceedings of the 2nd Global WordNet Conference, 20 - 23 January*, pages 326–331. Masaryk University, Brno, Czech republic.

Elke Teich and Peter Fankhauser. 2005. Exploring lexical patterns in text: lexical cohesion analysis with WordNet. In *Heterogeneity in focus: Creating and using linguistic databases. Interdisciplinary studies on information structure*, volume 2, pages 129–145. Universität Potsdam.

Nina Wacholder, Smaranda Muresan, Debanjan Ghosh, and Mark Aakhus. 2014. Annotating Multiparty Discourse: Challenges for Agreement Metrics. In Lori Levin and Manfred Stede, editors, *Proceedings of LAW VIII - The 8th Linguistic Annotation Workshop*, pages 120–128, Dublin, aug.

Li Wang, Diana Mccarthy, and Timothy Baldwin. 2011. Predicting Thread Linking Structure by Lexical Chaining. In Diego Mollá and David Martinez, editors, *Proceedings of the Australasian Language Technology Association Workshop 2011*, volume 9, pages 76–85, Canberra, dec. Australasian Language Technology Association.

Chaffin R. Winston, M.E. and D. Herrmann. 1987. A taxonomy of part-whole relations. In *Cognitive Science*, number 11, pages 417–444.

A Figures

```
candidates = []
for token in tokens_of_text:
  if token == noun:
    append.candidates(token)
ties = []
for candidate in candidates:
  all_pairs = get_all_pairs(candidate,
    ↪  candidates)
  all_pairs = filter_pairs(all_pairs)
  append.ties(all_pairs)
chains = []
for chain in chains:
  for tie in ties:
    if tie[0] in chain or tie[1] in
      ↪  chain:
      append.chain(tie)
    else:
      new_chain = [tie]
      chains.append(new_chain)
for chain in chains:
  chain = link_ties(chain)
chains = filter(chains)
```

Figure 1: Pseudo-code for lexical chain building algorithm

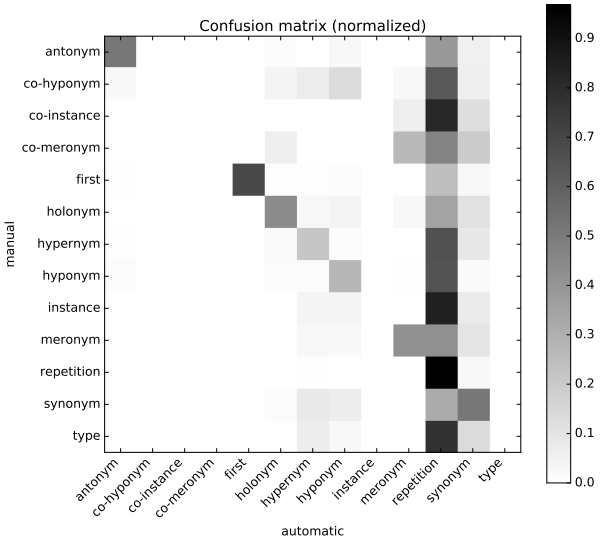

Figure 2: Confusion matrix for annotation of semantic relations manual vs. automatic in English.

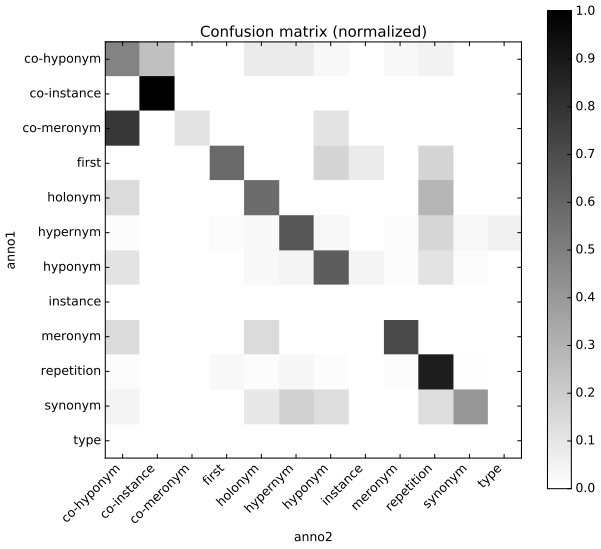

Figure 3: Confusion matrix for annotation of semantic relations Annotator 1 vs. Annotator 2 in English.

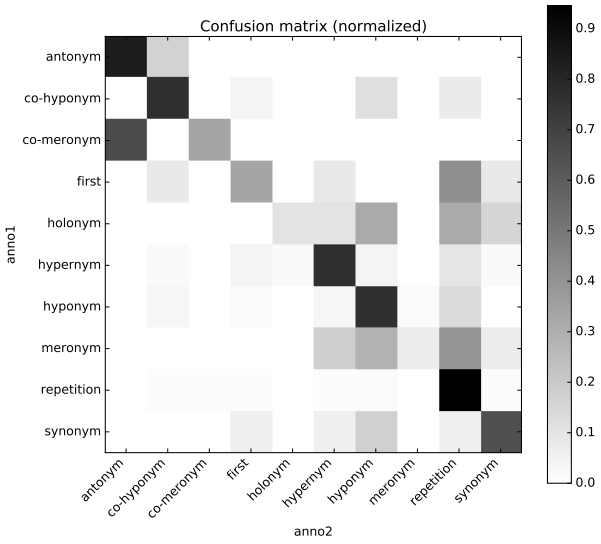

Figure 4: Confusion matrix for annotation of semantic relations Annotator 1 vs. Annotator 2 in German.

```
I think that is built into <lexicalcohesion id="markable_313" lexical_type="hyponym"
  ↪ lexical_chain="set_49"> broadcast legislation </lexicalcohesion> but it there
  ↪ it is not there for the <lexicalcohesion id="markable_377"
  ↪ lexical_type="co-hyponym" lexical_chain="set_49"> cinema legislation
  ↪ </lexicalcohesion>, for <lexicalcohesion id="markable_378"
  ↪ lexical_type="co-hyponym" lexical_chain="set_49"> film  legislation
  ↪ </lexicalcohesion>. There is no <lexicalcohesion id="markable_316"
  ↪ lexical_type="repetition"  lexical_chain="set_49"> film legislation
  ↪ </lexicalcohesion>. I know that the <lexicalcohesion id="markable_312"
  ↪ lexical_type="repetition"  lexical_chain="set_113"> UK </lexicalcohesion> 's
  ↪ quite advanced, isn't it, in terms of audiodescription compared with the rest
  ↪ of  <lexicalcohesion id="markable_318" lexical_type="holonym"
  ↪ lexical_chain="set_113"> Europe </lexicalcohesion> , for example . What do you
  ↪ think it is that makes us, or makes the <lexicalcohesion id="markable_319"
  ↪ lexical_type="meronym"  lexical_chain="set_113"> UK </lexicalcohesion> UK, so
  ↪ good at doing this?
```

Figure 5: Annotated lexical chains in the corpus

Automatic authorship attribution based on character n-grams in Swiss German

Rahel Oppliger
University of Zurich
rahel.oppliger@uzh.ch

Abstract

Automatic authorship attribution aims to train computers to identify the author of a disputed text based on idiolectal language features. When confronted with non-standard data – in the present study Swiss German instant messages – language-specific NLP toolkits are often unavailable, limiting the availability of features to classify texts. Thus, the approach I propose for Swiss German is based on character n-grams, which not only avoids the problem of a lack of available NLP tools, but – in addition to being a proven successful feature for authorship attribution – allows the capturing of orthographical idiosyncrasies. It thus allows the exploitation of Swiss German's lack of standardised spelling rules, turning the challenge that Swiss German presents as non-standard data into an advantage. Different lengths of n-grams as features of a Naïve Bayes classifier combined with varying sizes of training and test corpora were tested, and 6- and 7-grams were found to faultlessly identify authors for all combinations considered. The number of distinctive n-grams in an author's data set was found to be a determining factor for the classifier's success, highlighting the benefits of exploiting Swiss German's non-standard nature for authorship identification.

1 Introduction

Identifying the authors of texts purely based on stylometric evidence has been of interest to linguists since the 19th century, when Augustus DeMorgan suggested that authors can be identified according to the average word length in their texts, an idea taken up by Mendenhall (1887, p. 237). Since these early explorations, which have since been discovered to lack sufficient empirical foundation (Holmes, p. 88), various measures have been employed to determine authorship, ranging from sentence length, over vocabulary peculiarities, to the use of particular syntactic structures. Such methods have found application not only in determining the idiolectal styles of authors, but they have prominently been used to determine authorship of the Federalist Papers (Mosteller and Wallace, 1964), and they are predominantly employed in forensic linguistics – to aid in criminal cases that contain evidence of disputed authorship (Olsson and Luchjenbroers, 2013, pp. 7-9).

Over the last decades, large efforts have been undertaken to automate the process of authorship attribution, as well as to render it more empirically founded. Automatic authorship attribution, as this method has been termed, aims to determine the author of a disputed text reliably by identifying features indicative of the author's writing style utilising a corpus of known texts.

Since this new attribution paradigm relies solely on computers, the availability of a variety of natural language processing (NLP) tools directly influences the availability of features. The tools needed to process language for feature extraction range from tokenisers, over part-of-speech taggers, to syntactic parsers and semantic annotation tools. However, such extensive NLP toolkits are only available for the major languages that are regularly subjected to computational linguistic research.

As this study will be focused on Swiss German, the language independence of the method is vital considering this language is vastly under-researched, especially with regards to NLP. The difficulties of processing Swiss German lie predominantly in the fact that it is not a standardised language, and only recently have there been efforts to develop NLP tools for it. Hollenstein and Aepli (2014) have presented first steps towards develop-

ing a part-of-speech tagger for Swiss German, but more progress needs to be made until such a tool is fully dependable. Thus, the challenge of the present study is finding a reliable, standardised method for such non-standard data; in an attempt to achieve the best possible result despite the restrictions of software availability for Swiss German language data, the approach presented in the following will be based on character n-grams – a selection justified in Section 2.

As much as authorship attribution within a non-standard language like Swiss German poses problems in terms of software availability, there may also be peculiarities of a non-standard language to be exploited: in this study, the fact that Swiss German does not have standardised spelling and thus encourages individual spelling styles will be used to distinguish authors. The hypothesis I propose is that based on character n-grams that are able to capture the idiolectal orthography in written Swiss German, automatic authorship attribution is possible.

In the following, I will first review the literature on Swiss German orthography and n-gram-based automatic authorship attribution – justifying the selection of n-grams as a stylometric measure for my data – followed by a discussion of the processing necessary to train a successful classifier. I will analyse the group conversation of four authors on the instant messaging app WhatsApp, selecting a portion of the data to function as a training corpus from which to extract features to be used to automatically attribute authors to the messages in the second portion. This attribution process is carried out by a Naïve Bayes classifier that is trained on a feature set of character n-grams. Finally, I conclude with a discussion of how distinctive features interact with the success of the n-gram classifier.

2 Literature review

2.1 Feature selection

As part of the paradigm shift to computer-based methods in authorship attribution discussed in the introduction, the research community's aim has been to find features that are both easy to extract automatically and are maximally indicative of an author's written idiolect. Stamatatos (2009, pp. 540-544) presents a selection of such stylometric features that have been applied in authorship attribution, including lexical, character, syntactic, and semantic features. As he describes, the depth of text analysis and thus the complexity of NLP systems differs vastly from feature to feature: for instance, a lexical aspect such as word frequencies requires only a tokenised text, whereas the pre-processing needed to allow for the extraction of sentence and phrase structure features is extensive.

In an early overview of computer-based approaches, Holmes (1994) lists various features using elaborate pre-processing to differentiate authors: he questions the suitability of features such as word-length, sentence-length, and word frequencies, but he argues that approaches combining multiple features show particular promise. Such methods that incorporate various features are arguably able to capture an individual's idiolectic style more comprehensively. Over ten years later, Grieve (2007, pp. 266-67) reaches a similar conclusion: in his analysis of different approaches to automatic authorship attribution, he finds that by combining measurements, test accuracies above 93% are achieved for four possible authors – the number of authors considered in the present study – and for two authors, the accuracy is as high as 97%.

While the state of the art automatic authorship attribution approach is one that combines a variety of features, from a processing point of view it is preferable to focus on less features. The best-performing individual algorithm that Grieve (2007) tested for is one that distinguishes individuals according to how often they use various punctuation marks relative to the number of words in their texts. The success rate of this approach is 95% for two possible authors and 89% for four.

Both the construction of a word and punctuation profile and the combination approach require at least the availability of a tokeniser in the former case (to determine the number of words in a text) and a variety of NLP tools depending on the features being combined in the latter. Thus, an alternative approach must be found for non-standard data that lacks reliable or fully automatic tools. Houvardas and Stamatatos (2006, p. 78) identify language independence as one of the major advantages of using character n-grams for authorship attribution; since n-grams simply consist of consecutive strings of characters, no pre-processing is needed.

In addition to the practical advantages of extracting character n-grams, they have also been proven to perform very well in a variety of studies. Grieve (2007) examines thirty-nine different methods of

quantitative authorship attribution, and in applying them all to the same data set he finds that n-grams – particularly bi-, tri-, and 4-grams – are only surpassed in accuracy by the aforementioned feature-combining approaches and word and punctuation profiles. Specifically, he finds that both bi- and trigrams achieve 94% and 88% accuracy for two and four possible authors respectively, while 4-grams perform slightly worse at 93% and 85%.

While Kešelj et al. (2003) confirm the high success rates of the character n-gram approach, they find larger n-gram sizes to perform better: using sequences of 4 to 8 characters produces the best outcome. Yet more strikingly, 6- and 7-grams achieve 100% accuracy in distinguishing seven authors for feature profile sizes of 500 to 3,000 features. It has to be considered, however, that the pool of seven possible authors that Kešelj et al. distinguish between is quite varied: it ranges from 16th century Shakespeare to 19th century Lewis Carroll. In such a diverse pool of authors, the language samples considered can be expected to vary not only in idiolect but also rather drastically with respect to the historical period they were produced in, and the topic they are written about. The 100% accuracies that this study presents thus have to be regarded with caution. Nevertheless, it is of interest that they observe that 6- and 7-grams outperform other n-gram sizes.

The circumstances that are encountered in authorship attribution are often challenging: many authors may have to be considered, and the amount of available data may be very limited. The suitability of the n-gram approach in such difficult cases is further attested in Houvardas and Stamatatos (2006), who find that character n-grams work particularly well for multiple authors (p. 78). Moreover, as n-grams make it possible to extract a large number of features from even short texts (Layton et al., 2012, p. 299), they are well suited to registers usually producing brief messages, such as the instant messages considered here.

The approach based on character n-grams has also proven to be successful for social media data. In his application of a variety of features to the authorship disambiguation in Dutch tweets, van Halteren (submitted) finds that trigrams perform better than lexical features. He finds the success of trigrams to be particularly apparent in Twitter data – a data type that shares characteristics like short length of messages and a low degree of formality with instant messages. One potential reason he provides for the success of the feature is its ability to capture many characteristics of a tweet, such as capitalisation, spelling variation, user mentions, or URLs.

This success of n-grams in the classification of short computer-mediated texts may thus be attributed to their ability to capture different elements of language. Layton et al. (p. 298) argue that character-level n-grams are able to represent peculiarities on every level of language, whether the information be morphological, lexical, orthographical, or syntactical. In particular the benefits of n-grams being able to capture information on orthography will be discussed in the following section.

2.2 Swiss German's lack of standardised orthography

To further justify the selection of character n-grams as the ideal feature to differentiate between authors in Swiss German, we must consider how the language is written. Swiss German does not exhibit standardised spelling rules: every user of the language develops an individual set of spelling conventions. Ruef and Ueberwasser (2013, pp. 61-62) attribute this lack of uniform orthography to the large diversity in regional dialects, as well as to the low number of texts being written in this predominantly spoken language – Standard German is used for written communication in Switzerland, presenting a model case of diglossia.

In a corpus of Swiss text messages, sms4science, Ueberwasser (2013, p. 8) finds that almost two thirds of the German messages were in fact written in Swiss German dialect, thus rendering text messages a register where Swiss German appears in a written form. However, Ruef and Ueberwasser (2013) report that despite a growth of written Swiss German due to its uses in computer-mediate communication, virtually no standardisation has taken place, which may be partly caused by the register being largely informal, as official communication still uses Standard German.

The data set used in this study is indeed taken from a very informal context, namely a instant messaging conversation between friends; thus, the use of Swiss German is favoured. However, the dialectal heterogeneity that Ruef and Ueberwasser (2013) cite as a cause for lacking spelling norms is not a large issue in the present data set: three out

of four participants grew up in the same town, and although there are slight differences in the participants' spoken dialects, they are overall very similar. For such a homogeneous group, Scherrer and Rambow (2010, p. 98) voice concerns whether character n-grams are able to distinguish between very similar dialects. Yet, the hypothesis of this study is based on my observation that even between the four participants in this group chat – who all talk very similarly – there are considerable differences in spelling.

Contrary to Scherrer and Rambow's (2010) concerns, I argue that it is precisely with n-grams that we can successfully exploit the Swiss German particularity of exhibiting large variety in orthographic idiolects for the purpose of automatic authorship attribution. This non-standard feature of Swiss German lends itself to be captured in n-grams, and in the following, I will illustrate the degree to which spelling differs among individuals even of the same or a very similar dialectal background.

English / German		**P. 1**	**P. 2**	**P. 3**	**P. 4**
write, text / schreiben	schriebe	22	0	6	6
	schriibe	0	0	0	22
	schribe	1	10	0	0
Friday / Freitag	Fritig	2	0	0	4
	fritig	17	6	11	0
	Friitig	0	0	0	14
not / nicht	nöd	63	152	115	165
	ned	71	0	0	1
then / dann	denn	23	105	21	235
	den	127	1	77	0
now / jetzt	jetzt	5	12	2	136
	ez	52	0	79	0
	ezt	0	43	0	0

Table 1: Frequencies of orthographic alternatives in the WhatsApp training corpus.

Table 1 shows a selection of such spelling differences, presenting inconsistencies in spelling both between and within speakers. While Ueberwasser (2013, p. 20) suggests that an individual's spelling is often consistent, Table 1 illustrates that this certainly does not always apply. For example, in spelling 'now/jetzt', the four participants each prefer one of three spelling variants: participants 1 and 3 favour *ez* very strongly, while participant 2 prefers *ezt* and participant 4 *jetzt*. Participant 2 moreover exhibits considerable variation, using *jetzt* in approximately a fifth of instances. Showing even more intra-speaker variation, participant 1 uses both *nöd* and *ned* at almost equal frequency. However, in many cases, speakers indeed show a tendency to use a specific variant, as exemplified by the other participants' clearly preferred use of *nöd* over *ned*.

Furthermore, as Ueberwasser also notes (2003, p. 20), one speaker may be consistent in representing identical sounds with the same letter or letter combination. This phenomenon can be observed in the spellings for the mid-word vowel [i:] in 'write/text' and 'Friday': participant 4 prefers to represent the long vowel as /ii/, whereas participant 2 favours /i/. At the same time, however, participants 1 and 3 do not follow this pattern, using /ie/ in 'write/text' and /i/ in 'Friday'. These differences in spelling as well as potential consistencies in how individuals choose to represent certain sounds can be captured by character n-grams. Additionally, the distribution of how often the variants are used by each author is also represented in the n-gram feature profiles. In their ability to incorporate these particularities of non-standardised orthography, character n-grams form the ideal basis for an authorship attribution classifier for Swiss German. To sum up, this approach not only avoids the difficulties in working with this type of non-standard data by requiring no pre-processing, but it crucially exploits the data's idiosyncrasies. In the following, the extraction of the n-gram features and their use in a Naïve Bayes classifier will be discussed.

3 Data and method

3.1 Data

The data used in this study was obtained from a group chat on the instant messaging app WhatsApp between four Swiss females, aged 20 to 24 at the time of production. All participants have given their consent for me to use the data in this study. The four participants all share a similar spoken dialect. The conversation is conducted in Swiss German with occasional occurrences of English, Standard German, and French. The training corpus consists of 5,141 messages, varying in length and with different participants having contributed to various degrees; the exact size of the training (TR) and test (TE) corpora is presented in Table 2.

		P. 1	P. 2	P. 3	P. 4
TR	*msg.*	1,069	1,384	1,384	1,443
	char.	53,995	45,255	47,157	85,489
TE	*msg.*	330	405	393	285
	char.	19,271	17,016	13,571	19,186

Table 2: Size of the training and test corpora, in messages and characters per participant.

The split into training and test corpora was made at a specific point in time, resulting in the uneven sized corpora. This choice was motivated by an aim to control for the influence of topic of conversation; by splitting the data at the same point in the conversation for every participant, similar topics should be located in the training and test corpora of all participants. However, it is worth noting that the varying sizes of training material per participant may have an influence on the performance of the classifier. The test corpus that was used was a smaller part of the same conversation; roughly, the test corpus for each participant is 15-25% the size of the respective training corpus.

While the general Machine Learning principle of 'more data is more' applies to this task, too, Layckx and Daelemans (2010) set out to define the desired data size. They suggest that the ideal training set size lies above 10,000 words per author, but they acknowledge that with the use of n-grams, satisfactory results can be obtained on much smaller data sets (p. 53). How well a classifier based on character n-grams performs on data sizes below that desirable threshold will be explored in the present study. Namely, in addition to training and testing the classifier on the full data set, it is trained on half the training data size and tested on half, a fifth, and a tenth of the testing data size.

3.2 Method

In Section 2, I outlined the practicality and proven success of character n-grams as a feature for authorship attribution. The task is then to extract n-grams from the WhatsApp data – the conversations can be downloaded in the app as a plain text file. I extracted the messages for each participant and created n-grams of variable length for all messages, storing both the n-grams and how frequently they occur for every participant in feature dictionaries.

As the data is taken from an informal computer-mediated register – specifically instant messaging – we can observe an extensive use of emojis. Since these form a potentially defining part of an author's idiolect, they were included as part of the n-grams.

After the collection of n-grams of varying lengths, the resulting feature dictionaries that represent the language of each participant are used to train a classifier, specifically a Naïve Bayes classifier. Juola (2006, p. 285) cites the relative ease of training as one of the chief advantages of Naïve Bayes classifiers. In fact, Bird et al.'s (2009) Natural Language Toolkit (NLTK) contains a module that I use in this study to train a Naïve Bayes classifier and apply it to data.

In order to determine how well the classifier copes with different amounts of data, I test the classifier on a number of combinations of training and test data; I aim to determine whether it can still provide accurate results with lower amounts of data. Additionally, I attempt to add my results to the studies described in Section 2.1 that have sought to determine what size n-grams deliver the best results, thus testing bigrams to 10-grams, as Kešelj et al. (2003) did.

4 Results and discussion

4.1 Naïve Bayes classifier performance

I trained and tested Naïve Bayes classifiers on a number of training and test data set combinations, noting simply how many of the authors were correctly identified for each n-gram size and data size combination. These results are presented in Table 3, with the numbers in each instance referring to how many of the four authors were correctly identified by the classifier. It is evident that the Naïve Bayes classifier overall performs above chance for all sizes of training and test corpora examined here. In fact, with a vast amount of data available to both train and test the classifier on, i.e. the full training and test set, the performance is near faultless, with only bigrams and 4-grams failing to deliver fully correct results.

However, perhaps more interestingly, certain sizes of n-grams appear to produce wholly accurate results for all sizes of data sets tested: classifiers trained on 6-grams and 7-grams succeed in identifying the correct authors in all categories of data size, while the 8-gram classifier only decides incorrectly for one author in one category. These results match the findings of Kešelj et al. (2003), who also find 6- and 7-grams to be the most effective. The results of my study thus support their argument that these length n-grams are most indicative of individual

	Full training Full test	0.5 training 0.5 test	Full training 0.2 test	Full training 0.1 test	0.5 training 0.2 test	0.5 training 0.1 test
2-g	3	3	0	0	0	1
3-g	4	4	1	0	1	0
4-g	3	3	1	1	2	1
5-g	4	4	3	2	4	4
6-g	4	4	4	4	4	4
7-g	4	4	4	4	4	4
8-g	4	3	4	4	4	4
9-g	4	3	3	4	4	4
10-g	4	3	3	4	4	4

Table 3: Naïve Bayes classifier results, split by n-gram size and size of training and test set.

writing style.

More faulty performances can be seen from the classifiers trained on bigrams, trigrams, and 4-grams, where the classifiers perform at, or below, chance level when trained and tested on less data. Thus, Grieve's (2007) findings that short n-grams perform best could not be confirmed. However, it has to be noted that text type may play a considerable role in what size n-gram is most successful – the results presented here should therefore be regarded as potentially being particular to the instant messaging register and their transferability to other text types considered with caution. Nevertheless, it can be said that – with the right selection of n-gram size and sufficient data – character-based classifiers work very well in determining authorship in Swiss German instant messages.

Although the classifier evidently has trouble when trained and tested on shorter n-grams, namely bi- to 4-grams, it has to be noted that when the full size and half size training and test sets are used, the performance for these size n-grams is still well above chance level. Only when either the test or training corpus are substantially smaller does the classifier struggle. In the following, I will attempt to uncover the source of this problem.

4.2 Sparse data problem with short n-grams

As an explanation for the comparatively bad performance of shorter n-grams (bi- to 4-grams), I suggest that the issue in the present experiment is one of sparse data. A lack of sufficient training or test data has frequently been identified as one of the main problems Machine Learning approaches to authorship attribution face in forensic linguistic cases (Coulthard, 2004, p. 432; Totty et al., 1987, pp. 16-17). However, it is important to note that the sparsity of the data does not simply relate to the number of features, but to the number of distinctive features, as will be outlined in the following.

The shorter n-grams' success in the larger data sets may be attributed to their ability to capture the writer's language behaviour on a level that allows the classifier to compute a language profile for them. With less data, the creation of such a profile is seemingly not possible for the classifier, as simply not enough of the individual's habitual language behaviours may be present. Moreover, the profiles might be too similar as they are based more on frequent words and character sequences in the language rather than the individual's language choices. Longer n-grams, on the other hand, may be able to capture habitual language features even within a small amount of data, as they are less likely to produce identical features for all participants but will rather find a sufficient number of distinctive features.

	Part. 1	**Part. 2**	**Part. 3**	**Part. 4**
$2-gram$	11	7	2	18
$6-gram$	79	52	102	218

Table 4: Number of distinctive 2-grams and 6-grams found by the classifier for the half-sized training set and tenth-sized test set.

To illustrate this type of sparse data problem, I compare bigrams and 6-grams – the best and worst-performing n-grams – within the smallest data set in this study. In order for the classifier to be effective, distinctive features have to be found; a feature is distinctive if it appears both in the training and test corpus of one author, but not in the training corpora of the other authors. Table 4 shows that 6-grams provide far more distinctive features than bigrams. An examination of the distinctive bigrams reveals that they predominantly include emojis. To

sum up, the sparse data problem in n-gram classification has to be considered not at the level of how many features are available to train the classifier with, but how many of those features are distinctive.

4.3 Distinctive features

Following from the hypothesis that the success of an n-gram classifier is dependent on how many distinctive n-grams are available for each author, I now aim to illustrate the connection between number of distinctive n-grams and performance of the Naïve Bayes classifier.

According to my hypothesis, we would expect the n-gram sizes that produce the best results in the automatic authorship attribution task to produce the most distinctive n-grams. And indeed, as is shown in Figure 1 below, 6-grams exhibit the most distinctive features for three out of four authors, with participant 4 producing more unique 5-grams.

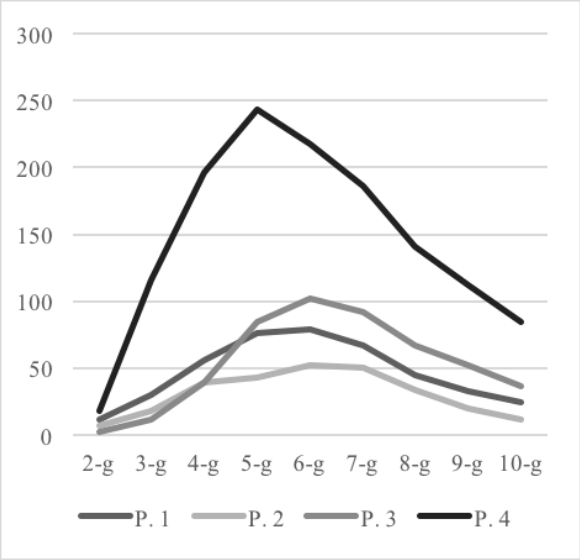

Figure 1: Distinctive n-grams for different n-gram lengths.

The n-gram sizes in Figure 1 reveal that the distribution for distinctive features peaks around 5- and 6-grams. As can be expected, participant 4, who provides the most training material, and thus allows for the extraction of more features, produces the most distinctive features for every n-gram size. However, the larger size of the training corpus shows the most benefits around its peak, while the very short and very long n-grams show similarly bad performances as with less data. I conclude from this case study that even with more training data available, the most efficient n-gram size, at least for this particular register, will be in the range of 5- to 8-grams, agreeing with Kešelj et al. (2003).

4.4 Distinctive spelling features

To illustrate how idiolectic spelling is reflected in the 6-grams that are found to be distinctive features, I will now take a look at participant 1's distinctive n-grams. A closer investigation of this participant's 79 distinctive 6-grams reveals that 23 involve spellings that are either highly indicative of this author or at least often used by her. Looking back at Table 1, where I demonstrated that participant 1 is the only author within this group to regularly – indeed over half of the time – use 'ned' to express *not/nicht*, it is perhaps not surprising that eight of the orthographically distinctive 6-grams contain this idiolectic spelling.

A further three distinctive sequences contain the lemma DERFEN, meaning *can/dürfen*, which is habitually spelled with a second vowel /e/ by participants 1 and 4, while participants 2 and 3 represent this vowel with /ö/. The /e/ vs. /ö/ distinction here is a similar one to that between 'ned' and 'nöd', and it is of interest to remark that participant 1 chooses the option /e/ in both cases. This observation supports Ueberwasser's (2013) hypothesis that identical sounds are often habitually represented by identical grapheme sequences.

Overall, consistent idiolectic spelling choices such as the aforementioned are here shown to be indicative of authorship, especially if they are characteristic of only the specific author. N-grams similar in length to the 6-grams investigated here are successful in capturing these orthographical idiosyncrasies. In this way, this characteristic orthographical freedom in Swiss German can be exploited as an effective feature for automatic authorship attribution.

5 Conclusion

In this paper, I have demonstrated that n-gram-based Naïve Bayes classifiers are successful in identifying authorship within four Swiss German speakers' instant messages. The outcome of this study leads me to conclude that character-based authorship attribution in Swiss German is a promising method, even for such small data sets as instant messages provide. I have suggested that the success of n-grams as features for identifying authorship in Swiss German may be amplified by the language's lack of standardised orthography,

encouraging each individual to develop their own spelling habits, which in turn may lead to a greater number of distinctive n-grams for the classifier to base its authorship attribution on.

Naïve Bayes classifiers based on different length n-grams delivered promising results in these authorship application tests, particularly for 6- and 7-grams, where perfect results for all data sets were achieved. These tests also revealed that the success of this method of authorship attribution lies in the classifier being able to find distinctive features within the data, creating a sparse data problem for shorter n-grams which fail to produce such distinctive features. Therefore, 5- to 7-grams proved to be the most suitable for the task, as they provide a sufficient number of distinctive features even within smaller data sets. Indeed, the number of distinctive features for any given n-gram size was found to correlate with how well the classifier performs.

While this study has presented promising results for character n-gram classifiers for automatic authorship attribution in Swiss German, further tests with a larger amount of authors and data will have to be undertaken in order to ensure the method's validity. Furthermore, rates of success will have to be tested more rigorously, particularly for forensic linguistic application.

Perhaps the most valuable finding of this study is that the non-standard nature of data is not merely a challenge to overcome, but that the particularities of a non-standard data set can be exploited. In this case, I have shown that Swiss German's characteristic lack of spelling rules – causing idiolectal orthography among its users – presents an opportunity to use this trait as an effective feature for automatic authorship attribution.

References

Steven Bird, Ewan Klein, and Edward Loper. (2009). *Natural Language Processing with Python*. Sabastopol, CA: O'Reilly Media, Inc.

Malcolm Coulthard. (2004). *Author identification, idiolect, and linguistic uniqueness. Applied Linguistics, 25*(4), 431-447.

Jack Grieve. (2007). Quantitative authorship attribution: An evaluation of techniques. *Literary and Linguistic Computing, 22*(3), 251-270.

Hans van Halteren. (submitted). Large scale authorship recognition on productive Dutch-speaking Twitter users.

Nora Hollenstein and Noëmi Aepli. (2014). Compilation of a Swiss German dialect corpus and its application to PoS tagging. Paper presented at the *Proceedings of the First Workshop on Applying NLP Tools to Similar Languages, Varieties and Dialects*, (pp. 85-94).

David I. Holmes. (1994). Authorship attribution. *Computers and the Humanities, 28*(2), 87-106.

John Houvardas and Efstathios Stamatatos. (2006). N-gram feature selection for authorship identification. Paper presented at the *Proceedings of the 12th International Conference on Artificial Intelligence: Methodology, Systems, Applications*, (pp. 77-86).

Patrick Juola. (2006). Authorship Attribution. *Foundations and Trends in Information Retrieval, 1*(3), 233-334.

Vlada Kešelj, Fuchun Peng, Nick Cercone, and Calvin Thomas. (2003). N-gram-based author profiles for authorship attribution. Paper presented at the *Proceedings of the Conference Pacific Association for Computational Linguistics, PACLING*, (pp. 255-264).

Robert Layton, Paul Watters, and Richard Dazeley. (2012). Recentred local profiles for authorship attribution. *Natural Language Engineering, 18*(3), 293-312.

Kim Luyckx and Walter Daelemans. (2010). The effect of author set size and data size in authorship attribution. *Literary and Linguistic Computing, 26*(1), 35-55.

Thomas C. Mendenhall. (1887). The characteristic curves of composition. *Science, 9*(214), 237-249.

Frederick Mosteller and David L. Wallace. (1964). *Inference and Disputed Authorship: The Federalist.* Reading, MA: Addison-Wesley.

John Olsson and June Luchjenbroers. (2013). *Forensic linguistics*. London: A&C Black.

Fuchun Peng, Dale Schuurmans, Vlado Kešelj, and Shaojun Wang. (2003). Language independent authorship attribution using character level language models. Paper presented at the *Proceedings of the tenth Conference of European Chapter of the Association for Computational Linguistics*, (pp. 267-274).

Beni Ruef and Simone Ueberwasser. (2013). The Taming of a Dialect: Interlinear Glossing of Swiss German Text Messages. In C. M. Bongartz and C. M. Riehl (Eds.), *Non-standard Data Sources in Corpus-Base Research*, (pp. 61-68). Aachen: Shaker.

Yves Scherrer and Owen Rambow. (2010). Natural language processing for the Swiss German dialect area. Paper presented at the *Proceedings of KONVENS 2010*, (pp. 93-102).

Efstathios Stamatatos. (2009). A survey of modern authorship attribution methods. *Journal of the American Society for Information Science and Technology, 60*(3), 538-556.

R. N. Totty, R. A. Hardcastle, and J. Pearson. (1987). Forensic linguistics: The determination of authorship from habits of style. *Journal of the Forensic Science Society, 27*(1), 13-28.

Simone Ueberwasser. (2013). Non-standard Data in Swiss Text Messages with a Special Focus on Dialectal Forms. In C. M. Bongartz and C. M. Riehl (Eds.), *Non-standard Data Sources in Corpus-Based Research*, (pp. 7-24). Aachen: Shaker.

Smoothing Syntax-Based Semantic Spaces: Let The Winner Take It All

Sebastian Padó* **Jan Šnajder†** **Jason Utt*** **Britta D. Zeller***
* Institute for Natural Language Processing, University of Stuttgart
{sebastian.pado,jason.utt,britta.zeller}@ims.uni-stuttgart.de
† Faculty of Electrical Engineering and Computing, University of Zagreb
jan.snajder@fer.hr

Abstract

Syntax-based semantic spaces are more flexible and can potentially better model semantic relatedness than bag-of-words spaces. Their application is however limited by sparsity and restricted coverage. We address these problems by smoothing syntax-based with word-based spaces and investigate when to choose which prediction. We obtain the best results by picking the maximal predicted similarity for each word pair, taking advantage of the tendency of unreliable models to underestimate similarity. We show that smoothing can substantially improve coverage while maintaining prediction quality on two German benchmark tasks.

1 Introduction

Distributional semantics (Turney and Pantel, 2010) assumes that the semantic similarity between words is correlated with usage in the same linguistic contexts. Words can be represented by vectors of their co-occurrence frequencies with context elements.

Two major types of models used today are (a) *bag-of-words (BOW)* models, which use words within a surface window around the target word as contexts, and (b) *syntax-based* models, whose contexts include dependency information. These two types can be found among count models such as those studied here as well as newer predictive models (Mikolov et al., 2013; Levy and Goldberg, 2014).

There is an inherent trade-off between BOW models and syntax-based models: Syntax-based models build on a rich, structured notion of context and can capture fine-grained semantic phenomena such as predicate-argument plausibility (Baroni and Lenci, 2010) and can be considered as allowing more representative semantic similarity predictions. At the same time, syntax-based spaces are more prone to *sparsity problems*: Syntactic co-occurrences are less frequent, and the spaces are very high-dimensional. Vectors for rare words can be so sparse that there is no overlap with any other word, and the words effectively fall out of coverage, resulting in less reliable performance. In contrast, BOW models have almost perfect coverage, but provide a more coarse-grained semantic similarity.

This situation raises the question of how different models of differing levels of granularity can be combined in a globally beneficial manner. There is a research tradition that has developed strategies to unify different input vector spaces into a joint output representation. Andrews et al. (2009) combine feature norms with distributional information. Bruni et al. (2011) experiment with textual and visual distributional features. Fyshe et al. (2013) use word-based and dependency-based features as sources of topical and relational information. All of these studies assume that the information provided by the "input" spaces is of comparable quality, but contains different types of information, and can therefore be combined on equal footing – by dimensionality reduction, feature collation, or even simple addition.

Our work assumes a different point of view, namely that there is an *accuracy-coverage trade-off* among our input spaces, as described above. This resembles the situation in n-gram language modeling where models are typically combined by *smoothing*. We also frame the combination of distributional models a as smoothing problem, combining models not at the level of co-occurrence information, but at the level of *predictions*. To our knowledge, few studies have taken this perspective, with the exception of Utt and Padó (2014) who combine cross-lingual and monolingual syntax-based models, and Padó et al. (2013) who use morphological information for smoothing.

We experiment with two smoothing strategies:

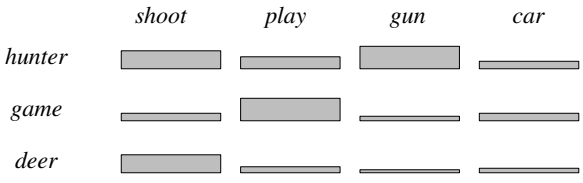

Table 1: Example of a bag-of-words space.

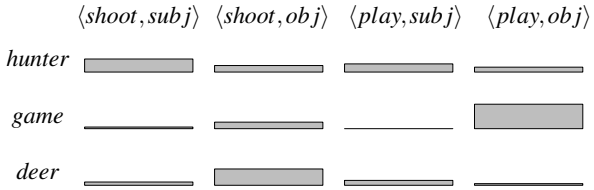

Table 2: Example of a syntax-based space.

Backoff and a *score maximization* strategy, which chooses the highest predicted score. Its intuition is that unreliable distributional models tend to underestimate semantic similarity. Experiments on two German benchmark tasks (semantic similarity prediction and synonym choice) show that score maximization can combine the high precision of syntax-based spaces with the high coverage of BOW spaces.

2 Smoothing Vector Spaces

2.1 Types of Vector Spaces

We concentrate on the two major types: bag-of-words (BOW) and syntax-based models.

BOW models represent target words in terms of context words co-occurring within a surface window. These models are simple, robust, and can be built from any tokenized corpus. They typically have a very high coverage (close to 100%). Different tasks require different context window sizes (Peirsman et al., 2008). Applying dimensionality reduction methods like Singular Value Decomposition (SVD) generally improves space quality.

Syntax-based models are based on word-link-word triples, typically dependency links. This versatile context makes them applicable to languages with free word order and allows them to capture structure-dependent semantic phenomena (Baroni and Lenci, 2010). At the same time, they are much sparser than BOW models, with a lower coverage overall (often 50–70%), which in particular makes the modeling of rare targets problematic. Also, their construction requires a large, well-parsed corpus, which has limited large-scale construction of syntax-based models to few languages (Baroni and Lenci, 2010; Padó and Utt, 2012; Šnajder et al., 2013). Utt and Padó (2014) proposed a cross-lingual method to induce syntax-based models without a parsed corpus, essentially "translating" existing English models. The filter effect created by the use of bilingual lexicon information amplifies the properties of syntax-based models: an even higher quality at the cost of a lower coverage.

2.2 Combining Vector Spaces

As stated above, we assume that there is an *accuracy-coverage* tradeoff between types of vector spaces. Thus, we do not want to unify the individual spaces, but combine their predictions in a sensible way.

Backoff. Backoff and interpolation are two methods that are standardly applied for smoothing in language modeling (Chen and Goodman, 1998). Given our assumptions, Backoff is a straightforward baseline method for combining semantic spaces. It simply defines a linear order on the models and predicts the first model in this order that makes a prediction. This approach was also followed by Utt and Padó (2014).

Score maximization. We propose a second smoothing strategy, *score maximization* or MAX, which chooses the maximum score from the predictions of individual models for each word pair. This strategy is motivated by the hypothesis described in Section 4.

3 Experimental Setup

Tasks. We evaluate on two German lexical-semantic benchmark tasks. The first one is semantic similarity prediction on the Gur350 wordsim dataset (Zesch et al., 2007).[1] It consists of 350 German word pairs with human relatedness ratings on a five-point scale.

The second task is synonym choice: For a target word, its synonym has to be picked from a list of four candidates. We use the German Reader's Digest Word Power dataset (Wallace and Wallace, 2005)[2] with 984 items. It is comparable to the English TOEFL dataset (Landauer and Dumais, 1997), but includes some short phrases as candidates.

[1] Available from: `http://goo.gl/3Dflf1`
[2] Available from: `http://goo.gl/PN42E`

Models. We experiment with three state-of-the-art count models. (1), the BOW space was built from the 800M-token German web corpus SDEWAC (Faaß et al., 2010) using a symmetric context window of size two. A space was extracted with 10k nouns, verbs and adjectives as dimensions, and reduced to 500 dimensions using SVD. (2), the monolingual syntax-based space, "DM", is the German version of Distributional Memory (Baroni and Lenci, 2010), DM.de (Padó and Utt, 2012), induced from a dependency-parsed version of the same corpus. (3), the cross-lingual DM, "tDM", was obtained via translation of the English DM (Utt and Padó, 2014) using the `dict.cc` EN-DE translation lexicon.

We apply both Backoff and score maximization. Model predictions are standardized before smoothing. For Backoff, we assume the linear order (3)>(2)>(1), since (3) has the highest quality, (1) the largest coverage, and (2) assumes an intermediate position. MAX is order-invariant.

Points of Comparison. We consider *random* (for synonym choice) and *frequency* baselines. For word similarity, the frequency baseline predicts the smaller of the two words' frequencies, $\min(f(w_1), f(w_2))$. For synonym choice, it predicts the candidate with the highest frequency. We also compare against current results from the literature, namely UP14 (Utt and Padó, 2014) and PSZ13 (Padó et al., 2013).

Prediction and Evaluation. We compute semantic similarity as cosine similarity. In the case of phrases, we compute the maximum pairwise word similarity. We make a prediction if both words are represented in the model and their vectors have a non-zero cosine. For synonym choice, we make a prediction for an item if we can make a prediction for at least one target–candidate pair.

On both tasks, we compute model coverage, defined as the percentage of items for which a prediction is made. On the similarity task, we measure quality as the Pearson correlation between human rating and model prediction. On the synonym choice task, we compute the accuracy of the covered items with partial credit for ties, following Mohammad et al. (2007). We report performance on all items as well as on the respective subset of covered items. We perform significance testing with bootstrap resampling (Efron and Tibshirani, 1993) on all items.

4 Underestimation Hypothesis

Informally, we believe that noise (e.g., from preprocessing) and sparsity (a perennial issue in distributional semantics) are quite unlikely to increase similarity by chance. To the best of our knowledge, this hypothesis has not been considered yet in the literature:

> **Underestimation hypothesis (UEH).** Unreliable distributional models are more likely to *underestimate* rather than overestimate semantic similarity.

We first develop a geometrical intuition and then corroborate our intuitions with an empirical study.

Geometrical argument . We assume that unreliable distributional models essentially mismeasure co-occurrence frequencies: They do not yield the *ideal vector* $\mathbf{v}$ for a given word, but an *empirical vector* $\hat{\mathbf{v}} = \mathbf{v} + \varepsilon$ that includes a noise vector ε.

We are interested in knowing when cosine similarity decreases due to noise ($cos(\mathbf{v}, \mathbf{w}) > cos(\hat{\mathbf{v}}, \mathbf{w})$). This can be determined by assuming (without loss of generality) that $\mathbf{v}$, $\hat{\mathbf{v}}$, and $\mathbf{w}$ are normalized. This makes them points on the unity hypersphere. Then the cosine decreases if and only if the "empirical" angle $\hat{\alpha}$ between $\hat{\mathbf{v}}$ and $\mathbf{w}$ is larger than the "ideal" angle α between $\mathbf{v}$ and $\mathbf{w}$. As Figure 1 shows, this is the case outside a hypersphere segment of width 2α centered on $\mathbf{w}$. If this segment is maximally wide (180°) if $\alpha = 90°$, it is equally likely that the cosine decreases or increases (in the absence of assumptions on ε). For all smaller angles α, the segment shrinks, and it becomes ever more likely that the cosine decreases, until it necessarily decreases for $\alpha = 0°$ (cf. Fig. 1).

Experimental support for UEH. In order to substantiate the claim of UEH, we designed the following experiment. Ideal vectors are simulated using the entire SDeWaC corpus, giving *'full sims'* for our word pairs. We also construct two halved subspaces by randomly assigning sentences to each half. Word similarities obtained from these two subspaces are termed *'half sims'*. If UEH is true, we would expect the half sims to be, more often than not, lower than the corresponding full sim. A t-test on Gur350 word pairs between full sims and half sims[3] shows a highly significant underes-

[3] As we have two half-sized subspaces, we double the number of wordpairs, pairing each full sim once with half sims

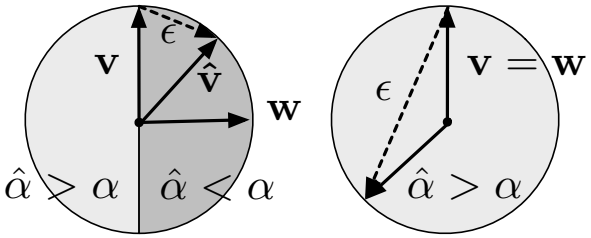

Figure 1: Underestimation hypothesis: ideal and empirical vectors ($\mathbf{v}, \hat{\mathbf{v}}$), point of comparison ($\mathbf{w}$), noise vector ($\varepsilon$). Segments of the hypersphere where angle decreases (dark grey) and increases (light grey). Left: $\alpha = 90°$ (lower $\hat{\alpha}$), Right: $\alpha = 0°$ (higher $\hat{\alpha}$).

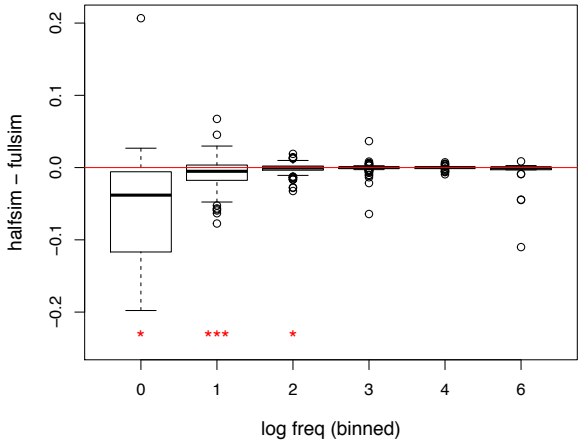

Figure 2: Differences between full and half sims by log frequency of word pairs. (Significance levels are shown for paired t-tests within each bin.)

timation ($t = -4.3647, df = 675, p = 1.473e-05$, mean difference: -0.003277829).

In a second analysis, we test further whether we can further isolate lower frequency word pairs as more reliably showing underestimation. This would correspond to the subnotion with UEH that less evidence for – or more noise in – the representations will intensify the underestimation.

Upon binning word pairs by minimum log frequency, we see that (cf. Figure 2) indeed lower frequency word pairs suffer more from underestimation.

We conclude that, if any of the models predicts a higher similarity, this is a more reliable signal and should be used at the exclusion of others.

from the first subspace, as well as the second. Uncovered items are excluded, in total $df + 1 = 676$ similarity pairs are tested.

	Word similarity			Synonym choice		
Model	r	r_{cov}	cov	acc	acc_{cov}	cov
Random	–	–	–	.25	.25	1
Frequency	.13	.13	1	.31	.31	1
BOW	.34	.34	.97	.52	.53	.95
DM	.38	.43	.60	.48	.53	.84
tDM	.33	.49	.49	.46	**.61**	.58
Smoothed models (sequence tDM>DM>BOW)						
Backoff	.40	.41	**.98**	.56	.57	**.97**
MAX	**.49**	**.50**	**.98**	**.57**	.59	**.97**
Results from the literature						
[UP14]	.42	.47	.69	.55	.59	.89
[PSZ13]	.47	NA	.89	.51	NA	.87

Table 3: Results for baselines and individual models (top), smoothed models (middle) and literature (bottom). Best results per column shown in boldface.

5 Results and Discussion

Table 3 shows the results. All individual models clearly outperform the baselines. Their individual performance matches our accuracy-coverage tradeoff assumptions from above. For example, on the word similarity task, coverage ranges between 97% (BOW) and 49% (tDM). On the covered items, the quality of the tDM predictions outperforms DM, which in turn outperforms BOW (r=.49/.43/.34). The patterns for synonym choice are parallel but less extreme.

The smoothing combination of the three models (tDM>DM>BOW) improves substantially over individual models.[4] In terms of the combination strategy, MAX yields higher results than Backoff.[5] For both tasks, MAX improves highly significantly on all items over the best individual model (word similarity: +.11 r vs. DM; synonym choice: +.05 accuracy vs. BOW; both significant at p<0.01). MAX also outperforms smoothing studies from the literature.

We see different results for the two tasks. On word similarity, smoothing has a larger impact, and the benefit of MAX over Backoff is significant only here (p<0.01). This can be explained by their properties. For word similarity, a regression task, each

[4] In preliminary experiments with the alternative approach of model unification (cf. Section 1), we did not find a comparable benefit for vector concatenation and PCA. This further bolsters our argument from Section 1.

[5] Other combination functions such as arithmetic, geometric and harmonic mean were also tested which however did not provide improvements, in line with UEH.

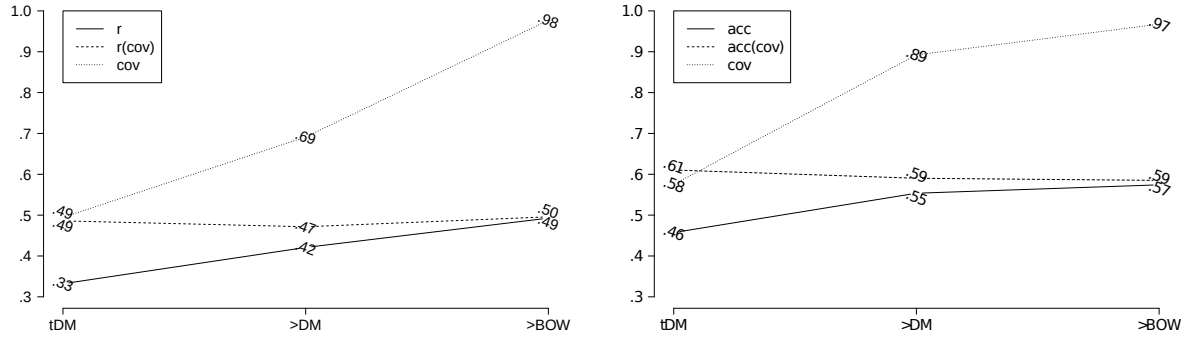

Figure 3: Performance of incremental smoothing (tDM, tDM>DM, tDM>DM>BOW) using score maximization (MAX) for the word similarity (left) and synonym choice (right) tasks

changed prediction influences the evaluation. In synonym choice, a classification task, it only matters which candidate has the highest similarity to the target – the similarities and margins are irrelevant. Consequently, classification is less sensitive to vector changes. This can be observed in practice: Backoff and MAX predictions differ on 155 of 350 word similarity pairs, while the predicted synonym changes only for 52 of 984 targets, i.e., the predictions are almost identical.

Figure 3 shows a more detailed analysis of smoothing. It plots the performance and coverage of MAX for incremental smoothing steps starting from tDM through tDM>DM to tDM>DM>BOW. The plots notably show that the quality on all items increases when adding more models while the quality on the covered items stays almost constant. This shows the robustness of MAX smoothing: The resultant models combine the almost perfect coverage of BOW models with the quality of syntax-based models.

6 Conclusions

This paper investigates the combination of accurate but sparse syntax-based semantic spaces with high-coverage BOW spaces, framing this problem as a smoothing task. We have shown how to reliably smooth by choosing the maximal prediction made by any model. This approach, a "winner-take-all" strategy, exploits the tendency of unreliable distributional models to underestimate semantic similarity making it possible to combine the benefits of different model types, improving both accuracy and coverage across two different semantic tasks and outperforming previous smoothing results. Due to the general nature of the factors giving rise to the underestimation – noise and sparsity in vector representations – we believe that our insights are applicable beyond the models considered in this paper, e.g., to syntax-based continuous vector spaces (Levy and Goldberg, 2014) and document-level models (Landauer and Dumais, 1997).

Acknowledgements

This research is partially funded by the German Research Foundation (SFB 732 at Stuttgart, projects B9 and D10). The second author has been supported by the Croatian Science Foundation under the project UIP-2014-09-7312. We thank the reviewers of this and two other conferences for valuable feedback.

References

Mark Andrews, Gabriella Vigliocco, and David Vinson. 2009. Integrating experiential and distributional data to learn semantic representations. *Psychological Review*, 116(3):463–498.

Marco Baroni and Alessandro Lenci. 2010. Distributional Memory: A general framework for corpus-based semantics. *Computational Linguistics*, 36(4):673–721.

Elia Bruni, Giang Binh Tran, and Marco Baroni. 2011. Distributional semantics from text and images. In *Proceedings of the GEMS 2011 Workshop on GEometrical Models of Natural Language Semantics*, pages 22–32, Edinburgh, UK.

Stanley F. Chen and Joshua Goodman. 1998. An empirical study of smoothing techniques for language modeling. Technical Report TR-10-98, Computer Science Group, Harvard University.

Bradley Efron and Robert J. Tibshirani. 1993. *An Introduction to the Bootstrap*. Chapman and Hall, New York.

Gertrud Faaß, Ulrich Heid, and Helmut Schmid. 2010. Design and application of a gold standard for morphological analysis: SMOR in validation. In *Proceedings of LREC*, pages 803–810.

Alona Fyshe, Brian Murphy, Partha Talukdar, and Tom Mitchell. 2013. Documents and dependencies: an exploration of vector space models for semantic composition. In *Proceedings of CoNLL*, pages 84–93, Sofia, Bulgaria.

Thomas K. Landauer and Susan T. Dumais. 1997. A Solution to Plato's Problem: The Latent Semantic Analysis Theory of Acquisition, Induction, and Representation of Knowledge. *Psychological Review*, 104(2):211–240.

Omer Levy and Yoav Goldberg. 2014. Dependency-based word embeddings. In *Proceedings of ACL*, pages 302–308, Baltimore, MD.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Proceedings of NIPS*, pages 3111–3119, Lake Tahoe, NV.

Saif Mohammad, Iryna Gurevych, Graeme Hirst, and Torsten Zesch. 2007. Cross-lingual distributional profiles of concepts for measuring semantic distance. In *Proceedings of the 2007 Joint Conference on EMNLP and CoNLL*, pages 571–580, Prague, Czech Republic.

Sebastian Padó and Jason Utt. 2012. A Distributional Memory for German. In *Proceedings of KONVENS 2012 Workshop on Lexical-semantic Resources and Applications*, pages 462–470, Vienna, Austria.

Sebastian Padó, Jan Šnajder, and Britta Zeller. 2013. Derivational smoothing for syntactic distributional semantics. In *Proceedings of ACL*, pages 731–735, Sofia, Bulgaria.

Yves Peirsman, Kris Heylen, and Dirk Geeraerts. 2008. Size matters: Tight and loose context definitions in English word space models. In *Proceedings of the ESSLLI Workshop on Distributional Lexical Semantics – Bridging the Gap Between Semantic Theory and Computational Simulations*, pages 34–41, Hamburg, Germany.

Jan Šnajder, Sebastian Padó, and Željko Agić. 2013. Building and evaluating a Distributional Memory for Croatian. In *Proceedings of ACL*, pages 784–789, Sofia, Bulgaria.

Peter D. Turney and Patrick Pantel. 2010. From Frequency to Meaning: Vector Space Models of Semantics. *Journal of Artificial Intelligence Research*, 37(1):141–188.

Jason Utt and Sebastian Padó. 2014. Crosslingual and multilingual construction of syntax-based vector space models. *Transactions of the Association of Computational Linguistics*, 2(Oct):245–258.

DeWitt Wallace and Lila Acheson Wallace. 2005. *Reader's Digest, das Beste für Deutschland*. Verlag Das Beste, Stuttgart.

Torsten Zesch, Iryna Gurevych, and Max Mühlhäuser. 2007. Comparing Wikipedia and German Wordnet by evaluating semantic relatedness on multiple datasets. In *Proceedings of NAACL/HLT*, pages 205–208, Rochester, NY.

Noun Sense Induction and Disambiguation using Graph-Based Distributional Semantics

Alexander Panchenko, Johannes Simon, Martin Riedl and Chris Biemann
Technische Universität Darmstadt, Computer Science Department, LT Group
Hochschulstr. 10, Darmstadt, Germany
{panchenko,simon,riedl,biem}@lt.informatik.tu-darmstadt.de

Abstract

We introduce an approach to unsupervised word sense induction and disambiguation: sense representations for ambiguous words are learned from distributional evidence and subsequently used to disambiguate word instances in context. These sense representations obtained by clustering dependency-parse-based second-order similarity networks as a pivot. We then add features for disambiguation from heterogeneous sources such as window-based and sentence-wide co-occurrences, and explore various schemes to combine these complementary context clues. Our method reaches a performance comparable to the state-of-the-art unsupervised word sense disambiguation systems including top participants of the SemEval 2013 word sense induction task and a more recent state-of-the-art neural word sense induction system.

1 Introduction

A word sense disambiguation (WSD) system takes as input a word and its context and outputs a sense of this word (Navigli, 2009). While the goal of all such methods is the same, there are substantial differences in their implementation. Some systems use knowledge-based approaches that rely on hand-crafted sense inventories, such as WordNet (Miller, 1995), while others use supervised approaches that learn from hand-labeled training data, such as SemCor (Miller et al., 1993). However, hand-crafted lexical resources and training data are expensive to create, often inconsistent and domain-dependent. Furthermore, these methods assume a fixed sense inventory for each word. This is problematic as (1) senses emerge and disappear over time; (2) different applications require different granularities of a sense inventory.

An alternative route explored in this paper is based on unsupervised knowledge-free approach. Our method learns an interpretable sense inventory by clustering semantically similar words. To learn sense inventories, we rely on the JoBimText framework and distributional semantics (Biemann and Riedl, 2013), adding a word sense disambiguation functionality on the top of it.

The key contribution of this paper is a framework that relies on such induced inventories as a pivot for learning contextual feature representations and uses them for disambiguation. The advantage of our method, compared to prior art, is that it can incorporate several types of context features in an unsupervised way. We demonstrate our approach, which combines four heterogeneous types of context features and achieves state of the art results in unsupervised WSD.

2 Related Work

Approaches to WSD vary according to the level of supervision and according to the amount of external knowledge they use (Agirre and Edmonds, 2007; Navigli, 2009).

Supervised approaches use an explicitly sense-labeled training corpus to construct a model, usually building one model per target word. Successful machine learning setups include SVMs (Lee and Ng, 2002) and classifier ensembles (Klein et al., 2002). Wee (2010) shows that decision trees using bag-of-word features are unable to outperform the most frequent sense baseline. Supervised approaches achieve the top performance in shared tasks on WSD such as SemEval, but require considerable amounts of sense-labeled examples.

A WSD method that uses predefined dictionaries, lexical resources or semantic ontologies can be considered *knowledge-based*. Knowledge-based systems rely on a lexical resource and vary from the classical Lesk (1986) algorithm that use word definitions to the *BabelFy* (Moro et al., 2014) system

that harnesses a multilingual semi-automatically constructed lexical semantic network. Knowledge-based approaches to WSD do not learn a model per target, but rather utilize information from a lexical resource that provides the sense inventory as well. Examples include (Lesk, 1986; Banerjee and Pedersen, 2002; Pedersen et al., 2005).

In this paper we deal with *unsupervised* and *knowledge-free* WSD approaches. They use neither handcrafted lexical resources nor hand-annotated sense-labeled corpora. Instead, they induce word sense inventories automatically from corpora. According to Navigli (2009), unsupervised WSD methods fall into two categories: context clustering (Pedersen and Bruce, 1997; Schütze, 1998) and word (ego-network) clustering (Lin, 1998; Pantel and Lin, 2002; Widdows and Dorow, 2002; Biemann, 2006; Hope and Keller, 2013a).

Context clustering methods, e.g. (Schütze, 1998), usually represent an instance by a vector that characterizes its context, where the definition of context can vary greatly. These vectors of each instance are then clustered. Multi-prototype extensions of the popular skip-gram model (Mikolov et al., 2013) also belong to the same group. They learn one embedding word vector per word sense and are commonly fitted with a disambiguation mechanism (Huang et al., 2012; Tian et al., 2014; Neelakantan et al., 2014; Bartunov et al., 2016; Li and Jurafsky, 2015).

The *AI-KU* system (Baskaya et al., 2013) is also based on context clustering. First, for each instance the system identifies the 100 most probable lexical substitutes using an *n*-gram model (Yuret, 2012). Each instance is thus represented by a bag of substitutes. These vectors are clustered using *k*-means. The *Unimelb* system by Lau et al. (2013) implements context clustering using the Hierarchical Dirichlet Process (HDP) (Teh et al., 2006). Latent topics discovered in the training instances, specific to every word, are interpreted as word senses.

Another class of word sense induction systems cluster *word ego-networks*, rather than single instances of words. An ego network consists of a single node (ego) together with the nodes they are connected to (alters) and all the edges among those alters, cf. Figure 1. Nodes of an ego-network can be (1) words semantically similar to the target word, as in our approach, or (2) context words relevant to the target, as in the *UoS* system (Hope and Keller, 2013a). Edges usually represent semantic similarities resp. association strength between nodes. The sense induction process using word graphs was previously explored by (Widdows and Dorow, 2002; Biemann, 2010; Hope and Keller, 2013a). Disambiguation of instances is performed by assigning the sense with the highest overlap between the instance's context words and the words of the sense cluster, similar to the simplified Lesk algorithm.

The *UoS* system by Hope and Keller (2013a) builds a word ego-network with nodes being the 300 highest-ranked words in a dependency relation with the target word and clusters the graph to obtain senses weighted by word similarities. The graph is clustered with the MaxMax algorithm. Similar clusters are merged. Disambiguation of instances is performed by assigning the sense with the highest overlap between the instance's context words and the words of the sense cluster.

While arguably the *UoS* system is the most similar to ours, there are crucial differences. First, nodes in their ego network are (first-order) context features, not (second-order) similar words. Second, edge weights in our network represent the number of shared features, not the significance of co-occurrences. Finally, their disambiguation component relies on overlap between context and a sense's cluster words.

Our system combines several of the above ideas, such as word sense induction based on clustering word similarities (Pantel and Lin, 2002), but in contrast to other unsupervised knowledge-free systems, we are able to combine and systematically evaluate the evidence from several features that model context differently.

3 Data-Driven Noun Sense Modelling

Our method consists of the three steps: computation of a distributional thesaurus, word sense induction, and building a disambiguation model of the induced senses.

3.1 Distributional Thesaurus of Nouns

The goal of this step is to build a graph of word similarities, such as "(tablet, notebook, 0.781)".[1] To compute the graph, we used the *JoBimText* framework (Biemann and Riedl, 2013). While multiple alternatives exist for the computation of semantic similarity e.g. (Mikolov et al., 2013), this framework is convenient in our case due to efficient

[1] We use the terms "semantic similarity/relatedness" to denote scores derived with a distributional semantics approach.

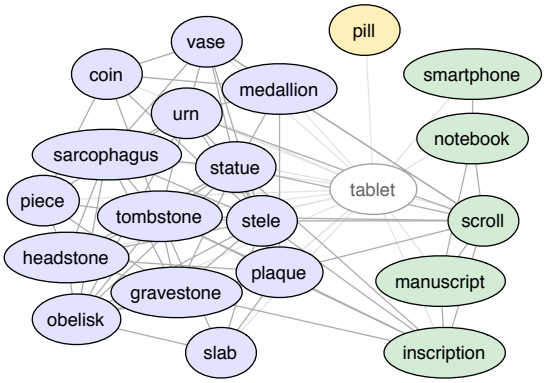

Figure 1: Visualization of the ego-network of the word "tablet" with three color-coded senses: "stone", "device", and "pill". Note that the ego word "tablet" is excluded from clustering.

computation of nearest neighbours for all words in the corpus while providing comparable precision (Riedl, 2016). For each noun in the corpus we retain the 200 most similar nouns.

3.2 Noun Sense Induction

Similar to (Pantel and Lin, 2002) and (Biemann, 2006), we induce a sense inventory which represents senses with word clusters. For instance, the the sense "tablet (device)" can be represented by the cluster "smartphone, notebook, scroll, manuscript, inscription", see Figure 1. To compute the clustering, first we construct an ego-network G of a word t and then perform graph clustering of this network. An ego-network (Everett and Borgatti, 2005) contains all nodes connected to the target node, called "ego". The identified clusters are interpreted as senses. Figure 1 depicts an ego-network of "tablet". Panchenko et al. (2013) proposed a system for dynamic visualization of word ego-networks similar to those used in our method.[2] The key property of word ego-networks is that the words with similar senses tend to be connected among each other, while having fewer connections to words from other senses, therefore forming clusters.

The sense induction processes one word t of the distributional thesaurus T per iteration. First, we retrieve nodes of the ego-network G being the N most similar words V of t according to T. Note that the target word t itself is not part of the ego-network. Second, we connect the nodes in G to their n most similar words from T. Finally, the ego-network is clustered with the Chinese Whispers algorithm (Biemann, 2006).

The sense induction algorithm has two meta-parameters: the *ego-network size* (N) of a target ego word t; and the *ego-network connectivity* (n) one of these neighbors v is allowed to have within the network. The parameter n regulates the granularity of the inventory. In our experiments we set N and n to 200 to obtain a coarse-grained inventory. In preliminary experiments, we found inventories based on dependency features superior to other inventories, which is why we use only dependency-based similarities in our WSI experiments.

3.3 Disambiguation of Induced Noun Senses

The goal of this step is to construct a disambiguation model $P(s_i|C)$ for each of the induced senses $s_i \in S$, where C is a feature representation of the target word w in a context. We approximate the conditional probability of the sense s_i in the context $C = \{c_1, ..., c_m\}$ with the Naïve Bayes model:

$$P(s_i|C) = \frac{P(s_i)\prod_{j=1}^{|C|} P(c_j|s_i)}{P(c_1, ..., c_m)}, \qquad (1)$$

where the best sense given C is chosen as following: $s_i^* = \arg\max_{s_i} P(s_i)\prod_{j=1}^{|C|} P(c_j|s_i)$.

To learn this model we use the assumption that words from a sense cluster S are, to some extent, semantically substitutable. For example, consider the sense cluster that represents the "fish" sense of the word "bass": {trout, catfish, eel, perch} and the following sentence: "*Most fish such as • live in freshwater lakes and rivers*". As can be observed in this example, similar words usually occur in similar contexts and thus often have similar context features. As it will be clear from our experiments, in spite of inherent noise in such training data one can use these data for training a disambiguation model.

Based on this assumption, it is possible to extract sense representations by aggregation of features from all words of the cluster s_i: we simply count in the training corpus the number of co-occurrences $f(w_k, c_j)$ and the cluster word w_{ik} with the context feature c_j across all words belonging to the sense cluster s_i: $\{w_1, ..., w_n\}$.

We cannot directly count any sense frequencies $f(s_i)$ or joint sense-feature frequencies $f(s_i, c_j)$ from an unlabeled text corpus. To estimate these frequencies we utilize an implication of our hypothesis: since two similar words are assumed to

[2] http://www.serelex.org

be substitutable, we assume any occurrence of the i-th word from the k-th cluster, denoted as w_k, to be interchangeable with an occurrence of sense s_i. The frequency of s_i is then given by $f(s_i) = \sum_i^{|s_i|} f(w_k)$, where $|s_i|$ is the number of words in the sense cluster s_i. The same principle can be applied to determine a joint frequency $f(s_i, c_j)$. To estimate the probability of a sense feature given a cluster word, we normalize the joint frequency by word frequency. This solves the problem of dominating high frequency cluster words:

$$P(c_j|w_k) = \frac{f(w_k, c_j)}{f(w_k)}. \quad (2)$$

A sense cluster usually contains a large number of similar words (up to $N = 200$ in our case). Often there is a high discrepancy among the similarities of the cluster words to the target word. Thus, some words better represent the sense than the others. To account for this effect, we introduce an additional weighting coefficient λ_k that is equal to the similarity between k-th cluster word w_k and the target word w being disambiguated.

While cluster words may be ambiguous, this issue is compensated by the fact that most cluster words have common features, while the noisy features of ambiguous words are specific to these words: they are not confirmed by noisy features of other ambiguous words. In some cases this assumption does not hold. For instance, the word "Chelsea" is similar to other words such as "Milan" or "Barcelona" that can represent both either a club or a city.

To normalize the score we divide it by the sum of all the weights $\Lambda_i = \sum_k^{|s_i|} \lambda_k$:

$$P(c_j|s_i) = \frac{1-\alpha}{\Lambda_i} \sum_k^{|s_i|} \lambda_k \frac{f(w_k, c_j)}{f(w_k)} + \alpha, \quad (3)$$

where α is a small number, e.g. 10^{-5}, added for smoothing.

The prior probability of each sense is computed based on the largest cluster heuristic:

$$P(s_i) = \frac{|s_i|}{\sum_{s_i \in S} |s_i|}. \quad (4)$$

We also explored estimation of the prior by a weighted average of cluster word counts, but this method provided lower results:

$$P(s_i) = \frac{1}{\Lambda_i} \sum_k^{|s_i|} \lambda_k \, f(w_k). \quad (5)$$

Note that to calculate the sense models we only need (1) the distributional thesaurus T; (2) sense clusters; and (3) word-feature frequencies: $f(w_k) = f_{n*}$, and $f(w_k, c_j) = f_{nm}$, where n is the index of the word w_k and m is the index of the feature c_j in a word-feature matrix. Finally, sense features are pruned: in our experiments, each sense s_i is represented with most significant 20,000 context features in terms of $P(c_j|s_i)$.

3.4 Feature Extraction and Combination

Our method learns separate models $P(s_i|C)$ for each type of context features. During classification, we either use these single-featured models directly or combine them at the feature- or meta-levels as described below.

Single features. We use four groups of word-feature counts $f(w_k, c_j)$ listed below to estimate probability of the feature given a sense $\hat{P}(c_j|s_i)$. A single-sense model is then trained for each of these feature types. Note that our framework allow using of any other context features if one can estimate $f(w_k, c_j)$ for it.

- **Cluster features** directly use words from the induced sense clusters i.e., the $\hat{P}(c_j|s_i)$ equals to the similarity score λ_{kj} between the target word w_k and the context word c_j.
- **Dependency features** of a target word w_k are all syntactic dependencies attached to it. For instance, the word "tablet" has features such as "subj(•,type)" or "amod(digital,•)", where "•" represents position of the target word. During disambiguation, we use this kind of features in two modes: the first one, denoted as *Deptarget*, represents the context C as a set of all dependencies attached to the target word being disambiguated; the second mode, denoted as *Depall* represents the context C with dependencies of all words in the sentence, not just the target word. This is an expansion of feature representation aiming to compensate the sparsity of the dependency representation.
- **Dependency word features**, denoted as *Depword*, are extracted from all syntactic dependencies attached to a target word w_k. Namely, we reduce dependency features to dependent words. For instance, the feature "subj(•,write)" would result in the feature "write". We also experimented with word co-occurrences, but they provided lower results.
- **Trigram features** are pairs of left and right

words around the target word w_k. For instance, the word "tablet" has features such as "typing_•_or" and "digital_•_.". Similarly to the dependency features, during disambiguation we use two modes to build the context C: the *Trigramtarget* represents the target word with one trigram extracted from its context; the *Trigramall* represents the target word with trigrams extracted from all words in the sentence.

Feature-level combination of features. This method builds the set of context features C uniting different context features under combination, such as dependencies and trigrams. Next, we use the Naïve Bayes model based on this extended context representation to estimate $\hat{P}(s_i|C)$, using conditional probabilities $\hat{P}(c_j|s_i)$ depending on the type of the corresponding feature $c_j \in C$.

Meta-level combination of features. This method starts by performing independent sense classifications with the combined models. Next, these predictions are aggregated using one of the three following strategies:

- **Majority** selects the sense s_i selected by the largest number of single models.
- **Ranks**. First, results of single model classification are ranked by their confidence $\hat{P}(s_i|C)$: the most suitable sense to the context obtains rank one and so on. Next, we assigns the sense with the least sum of ranks.
- **Sum**. This strategy assigns the sense with the largest sum of classification confidences i.e., $\sum_i \hat{P}(s_i|C_k^i)$, where i is the number of the single model.

4 Results

We evaluate our method on three complementary datasets: (1) a small-scale collection of homonyms used for convenient interpretation of results; (2) a large-scale collection of homonyms and polysemous senses used for development of meta-parameters; and (3) a mid-scale SemEval dataset used for comparison with other systems.

In the experiments described below, we trained models on two corpora commonly used for training distributional models: ukWaC (Ferraresi et al., 2008) and Wikipedia[3]. Table 1 presents statistics about these two text collections.

[3]We used a dump of Wikipedia of October 2015: http://panchenko.me/data/joint/corpora/en59g/wikipedia.txt.gz

	# Tokens	Size	Text Type
Wikipedia	$1.863 \cdot 10^9$	11.79 Gb	encyclopaedic
ukWaC	$1.980 \cdot 10^9$	12.05 Gb	Web pages

Table 1: Corpora used for training our models.

4.1 Evaluation on PRJ

The goal of this evaluation is to make sure the method performs as expected in simple settings i.e., in case of homonyms. We chosen a small scale dataset to be able to track each misclassified context.

Dataset. This dataset consists of 60 contexts of words "python", "ruby" and "jaguar", hence the name of the dataset (PRJ). Each word has two homonymous senses, respectively "snake" or "programming language", "gem" or "programming language", and "animal" or "car", respectively. Contexts were randomly sampled from the first three paragraphs of the corresponding Wikipedia articles. Each sense is represented with 10 contexts. We manually assigned senses from the induced inventory derived from the ukWaC corpus. In this experiment, we used the model trained on the ukWaC corpus.

Evaluation metrics. Since the contexts are labeled with the induced senses, we directly use precision and recall without mapping of inventories.

Discussion of results. Agirre and Soroa (2007) suggest that the WSD of homonyms is almost solved problem for supervised systems, reaching F-scores above 0.90. Our results summarized in Table 2 confirm this for the unsupervised approach. Our method reaches precision up to 0.953 and F-score of 0.950.

The three misclassified samples by the system reached F-score of 0.950 are the following. The first one is from the article about "ruby (gem)" which describes possible colors of ruby gems. It was wrongly labeled with the "ruby (color)" sense. The second misclassified example from the "jaguar (animal)" article contains multiple named entities, such as "USA" that strongly related to economic activities such car production. Finally, the reason of misclassification of the third context from the "python (snake)" article is that the "molurus" feature received high score in the "language" sense. We attribute this learning error due to unbalanced

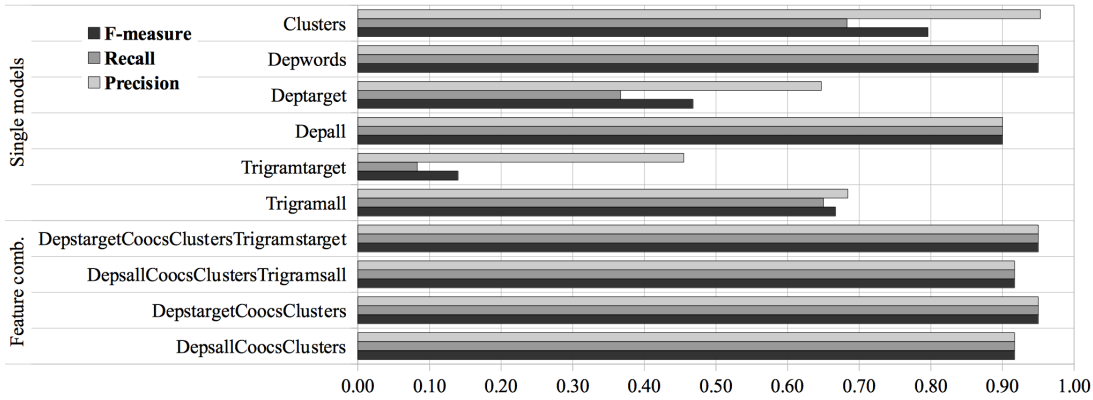

Figure 2: Performance of our method on the PRJ dataset. The models based on meta-combinations are not shown for brevity as they did not improve performance of the presented here models in terms of F-score.

nature of the ukWaC, as in the model trained on Wikipedia this feature has a higher score for the "snake" sense. Thus, we conclude that our approach performs as expected in simple cases, yielding almost no errors.

Combinations of the single predictors neither provide extra improvement in these simple settings: none of the combined models improve the overall results, nor they introduce any extra errors (see Figure 2). Finally, the meta-combination based on sum of ranks yielded the highest precision at the cost of a recall drop (not shown in Figure 2 for brevity).

4.2 Evaluation on TWSI

The goal of this evaluation is to test performance of our method on a large scale dataset that contains both homonyms and polysemous senses.

Dataset. This test collection is based on a large-scale crowdsourced resource (Biemann, 2012) that comprises 1,012 frequent nouns with average polysemy of 2.33 senses per word. For these nouns, 145,140 annotated sentences are provided. Besides, a sense inventory is explicitly provided, where each sense is represented with a list of words that can substitute target noun in a given sentence. The sense distribution across sentences in the dataset is highly skewed resulting in 79% of contexts assigned to the most frequent senses.

Evaluation metrics. To compute performance we create an explicit mapping between the system-provided sense inventory and the TWSI senses: senses are represented as bag of words vectors, which are compared using cosine similarity. Every induced sense gets assigned at most one TWSI sense. Once the mapping is completed, we can calculate Precision and recall of the sense labeling with respect to the original TWSI labeling.

Note that performance of a disambiguation model depends on quality of the sense mapping. Therefore, we use five baselines that facilitate interpretation of the results:

1. **MFS of the TWSI inventory** assigns the most frequent sense in the TWSI dataset.
2. **Random sense of the TWSI inventory**.
3. **MFS of the induced inventory** assigns the identifier of the largest sense cluster.
4. **Upper bound of the induced vocabulary** selects the correct sense for the context, but only if the mapping exist for this sense.
5. **Random sense of the induced inventory**.

Discussion of results. Table 2 presents evaluation of our method trained on the Wikipedia corpus (comparison of these results with the ukWaC corpus is provided in Figure 3). First, one can observe that, similarly to the PRJ dataset, the *Cluster* features yield a precise results up to $P = 0.719$. Yet, recall of these feature is inherently limited by the size of these clusters (15 to 200 words as compared to up to 20,000 for other types of features). Besides, *Trigramtarget* features yield even higher precision of 0.729, but their recall of 0.193 is even less than that of clusters. The single model based on the *Deptarget* features balances precision and recall, reaching F-measure of 0.571 at $P = 0.709$.

Several models based on feature- and meta-level combinations clearly outperform single-feature models. The best scores in terms of F-score (0.696-0.698) are obtained by a combination of four fea-

ture types (*Deptarget, Depword, Cluster, Trigramtarget*) at the feature level or using the sum meta-combination. Similar results (F-score of 0.694-0.695) can be obtained via combination of the same features without the *Trigramtarget*. In terms of precision, the best results are delivered by a meta-combination of the above-mentioned features, combined by summing their ranks. In these settings, the combined models yield precision of 0.713-0.720.

Figure 3 compares the performance of our models trained on the Wikipedia corpus and the ukWaC corpus. The Wikipedia-based models consistently outperform their counterparts trained on the ukWaC. This can be attributed to the fact that the TWSI contexts were originally sampled from the Wikipedia. Besides, Wikipedia is a more balanced and "clean" corpus than ukWaC.

All our models outperform the random sense baselines and the most frequent sense (MFS) baseline of the induced inventory in terms of precision and most of them outperforms these baselines in terms of F-score. These results show that the features used in our technique indeed provide a strong signal for word sense disambiguation. However, none of our models was able to outperform the most frequent sense of the TWSI.

We assumed that this is due to the highly skewed nature of the dataset where 79% of contexts are associated with the most frequent sense. To validate the hypothesis that our system yields state-of-the-art performance in spite of this result we compared its performance to a recent unsupervised WSD system based on sense embeddings, called AdaGram (Bartunov et al., 2016). This is a multi-prototype extension of the Skip-gram model (Mikolov et al., 2013), which relies on Bayesian inference to perform sense disambiguation. We chosen this method as it yields state-of-the-art results, outperforming other approaches based on sense embeddings, such as (Neelakantan et al., 2014). We tried several models varying the α parameter that controls granularity of the induced sense inventory. The best AdaGram configuration with the $\alpha =$ equals 0.05 yields F-score on of 0.656, which is below the most frequent sense of the TWSI, similarly to our top model *Deptarget-DepwordClusterTrigramtarget* that reaches F-score of 0.698.

4.3 Evaluation on SemEval-2013 Task 13

The goal of this evaluation is to compare performance of our method to the state-of-the-art unsupervised WSD systems.

Dataset. The SemEval-2013 task 13 "Word Sense Induction for Graded and Non-Graded Senses" (Jurgens and Klapaftis, 2013) provides 20 nouns, 20 verbs and 10 adjectives in WordNet-sense-tagged contexts. It contains 20-100 contexts per word, and 4,664 contexts in total, which were drawn from the Open American National Corpus. In our experiments, we use the 1,848 noun-based contexts. Participants were asked to cluster these 4,664 instances into groups, with each group corresponding to a distinct word sense. We report result on the 20 nouns as our approach is designed for nouns.

Evaluation metrics. Performance is measured with three measures that require a mapping of sense inventories (Jaccard Index, Tau and WNDCG) and two cluster comparison measures (Fuzzy NMI and Fuzzy B-Cubed).[4] During evaluation the test data is divided into five segments: four of which are used to build the mapping, and one for evaluation.

Discussion of results. Participating teams in this task were *AI-KU* (Baskaya et al., 2013), *Unimelb* (Lau et al., 2013), *UoS* (Hope and Keller, 2013b) and *La Sapienza*. The latter relies on WordNet as sense inventory and uses a knowledge-rich approach to disambiguation. Only the *UoS* used an induced sense inventory, similarly to us, while all other participating teams performed sense clustering directly on the disambiguation instances, thus not being able to classify additional instances without re-clustering the whole dataset.

Table 3 compares the performance of our method to other approaches. As one may observe, most of the combined models only sightly improve over the single-feature models according to Jaccard Index and Fuzzy NMI. However, one class of combined models that achieves a consistent improvement over the single-feature systems is the meta-combination based on the sum of ranks. Similarly to the TWSI experiment, the two best combined models are based either on four (*Deptarget, Depword, Cluster, Trigramtarget*) or three (*Deptarget, Depword, Cluster*) features. These two models

[4] Detailed interpretation of the five performance metrics: https://www.cs.york.ac.uk/semeval-2013/task13/index.php%3Fid=results.html

Model		#Senses	Precision	Recall	F-score
TWSI baselines	MFS of the TWSI inventory	2.31	0.787	0.787	0.787
	Random sense of the TWSI inventory	2.31	0.535	0.535	0.535
Induced baselines	Upper bound of the induced inventory	1.64	1.000	0.746	0.855
	MFS of the induced inventory	1.64	0.642	0.642	0.642
	Random Sense of the induced inventory	1.64	0.559	0.558	0.558
Sense embeddings	AdaGram, $\alpha = 0.05$, upper bound of induced inv.	4.33	1.000	0.865	0.928
	AdaGram, $\alpha = 0.05$	4.33	0.656	0.656	0.656
Single models	Cluster	1.64	**0.719**	0.405	0.518
	Depword	1.64	0.684	0.684	0.684
	Deptarget	1.64	0.709	0.571	0.633
	Depall	1.64	0.689	0.689	0.689
	Trigramtarget	1.64	**0.729**	0.193	0.305
	Trigramall	1.64	0.670	0.561	0.611
Feature comb.	DeptargetDepwordClusterTrigramtarget	1.64	0.698	**0.698**	**0.698**
	DepallDepwordClusterTrigramall	1.64	0.697	**0.697**	**0.697**
	DeptargetDepword Cluster	1.64	0.694	**0.694**	**0.694**
	DepallDepwordCluster	1.64	0.691	**0.691**	0.691
Meta comb.	Cluster+Deptarget+Depword+Trigramtarget: majority	1.64	**0.718**	0.605	0.656
	Cluster+Deptarget+Depword+Trigramtarget: ranks	1.64	0.687	0.360	0.472
	Cluster+Deptarget+Depword+Trigramtarget: sum	1.64	0.696	**0.696**	**0.696**
	Cluster+Depall+Depword+Trigramall: majority	1.64	0.692	0.685	0.688
	Cluster+Depall+Depword+Trigramall: ranks	1.64	**0.715**	0.420	0.529
	Cluster+Depall+Depword+Trigramall: sum	1.64	0.693	0.693	0.693
	Cluster+Deptarget+Depword: majority	1.64	0.704	0.630	0.665
	Cluster+Deptarget+Depword: ranks	1.64	0.713	0.410	0.521
	Cluster+Deptarget+Depword: sum	1.64	0.695	0.695	**0.695**
	Cluster+Depall+Depword: majority	1.64	0.689	0.688	0.688
	Cluster+Depall+Depword: ranks	1.64	**0.720**	0.406	0.519
	Cluster+Depall+Depword: sum	1.64	0.693	0.693	0.693

Table 2: Performance of our method on the TWSI dataset trained on the Wikipedia corpus. Top 5 scores of our approach per section are set in boldface; the best scores are underlined.

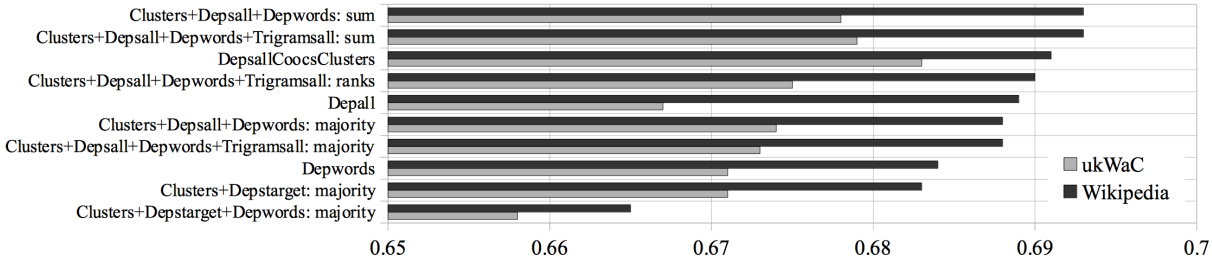

Figure 3: Effect of the corpus choice on the WSD performance: 10 best models according to the F-score on the TWSI dataset trained on Wikipedia and ukWaC corpora.

perform comparably to the best participants of the SemEval challenge or outperform them, depending on the metric. On one hand, the top SemEval system (AI-KU remove5-add1000) reaches Jaccard Index of 0.229 while our approach obtains scores up to 0.219. The second best SemEval system according to this metric (UoS top-3) has a score of 0.220. On the other hand, according to the Tau and Fuzzy B-Cubed scores, our best systems outperform the SemEval participants. Therefore, we conclude that performance of our approach is comparable to the other unsupervised state-of-the-art word sense disambiguation approaches.

Finally, note that none of the unsupervised WSD methods discussed in this paper, including the top-ranked SemEval submissions and the AdaGram, were able to beat the most frequent sense baselines of the respective datasets. Similar results are observed for other recently proposed unsupervised word sense disambiguation methods (Nieto Piña and Johansson, 2016).

5 Conclusions

Performance of the state-of-the-art knowledge-based and supervised WSD systems reached satisfactory levels, but they inherently suffer from inevitable out of vocabulary terms in any "non-standard" domain or language. We presented a new unsupervised knowledge-free approach to word sense induction and disambiguation that addresses these problems as it can be trained on a domain-

Model		Jacc. Ind.	Tau	WNDCG	Fuzzy NMI	Fuzzy B-Cubed
Baselines	One sense for all	0.171	0.627	0.302	0.000	0.631
	One sense per instance	0.000	0.953	0.000	0.072	0.000
	Most Frequent Sense (MFS)	0.579	0.583	0.431	–	–
SemEval systems	AI-KU (add1000)	0.176	0.609	0.205	0.033	0.317
	AI-KU	0.176	0.619	0.393	**0.066**	0.382
	AI-KU (remove5-add1000)	**0.228**	**0.654**	0.330	0.040	0.463
	Unimelb (5p)	0.198	0.623	0.374	0.056	0.475
	Unimelb (50k)	0.198	0.633	0.384	0.060	**0.494**
	UoS (#WN senses)	0.171	0.600	0.298	0.046	0.186
	UoS (top-3)	0.220	0.637	0.370	0.044	0.451
	La Sapienza (1)	0.131	0.544	0.332	–	–
	La Sapienza (2)	0.131	0.535	**0.394**	–	–
Sense embeddings	AdaGram, $\alpha = 0.05$, 100 dim. vectors	**0.274**	0.644	0.318	0.058	0.470
Single models	Cluster	0.196	0.652	0.319	0.032	0.610
	Depword	0.196	0.652	0.319	0.032	0.610
	Deptarget	0.189	0.655	0.314	0.025	0.610
	Depall	0.188	0.650	0.313	0.029	0.608
	Trigramtarget	0.179	0.632	0.303	0.009	**0.612**
	Trigramall	0.182	0.650	0.302	0.015	0.594
Feature comb.	DeptargetDepwordClusterTrigramtarget	0.188	0.654	0.317	0.032	**0.611**
	DepallDepwordClusterTrigramall	0.197	0.652	0.317	0.034	**0.611**
	DeptargetDepwordCluster	0.189	0.655	0.318	0.033	**0.611**
	DepallDepwordCluster	0.197	0.651	0.317	0.034	**0.611**
Meta comb.	Cluster+Deptarget+Depword+Trigramtarget: majority	0.197	0.645	0.317	0.037	0.600
	Cluster+Deptarget+Depword+Trigramtarget: ranks	**0.219**	**0.657**	0.309	0.034	0.487
	Cluster+Deptarget+Depword+Trigramtarget: sum	0.204	0.646	0.320	0.040	0.607
	Cluster+Depall+Depword+Trigramall: majority	0.196	0.646	0.315	0.035	0.601
	Cluster+Depall+Depword+Trigramall: ranks	0.216	0.654	0.316	0.042	0.526
	Cluster+Depall+Depword+Trigramall: sum	0.193	0.651	0.317	0.034	0.605
	Cluster+Deptarget+Depword: majority	0.200	0.647	0.317	0.039	0.601
	Cluster+Deptarget+Depword: ranks	**0.217**	**0.659**	**0.324**	**0.048**	0.533
	Cluster+Deptarget+Depword: sum	0.204	0.647	0.319	0.040	0.607
	Cluster+Depall+Depword: majority	0.200	0.647	0.317	0.039	0.601
	Cluster+Depall+Depword: ranks	0.200	0.646	0.317	0.039	0.601
	Cluster+Depall+Depword: sum	0.197	0.655	0.318	0.038	0.607

Table 3: Performance of our method on the nouns contexts from the SemEval 2013 Task 13 dataset. The models were trained on the ukWaC corpus. Top scores of the state-of-the-art systems (SemEval participants and the AdaGram) and of our systems are set in boldface; the best scores overall are underlined.

specific texts. The method takes as input a text corpus and learns an interpretable coarse-grained sense inventory, where each sense has a rich feature representation used for disambiguation.

The novel element of our approach is the use of an induced sense inventory as a pivot for aggregation and combination of heterogeneous context clues. This framework let us easily incorporate various context features in a single model. In our experiments we demonstrated combinations of four classes of features, but the framework can easily accommodate other types of features.

While other systems already used some features employed in our approach (e.g., the UoS system relies on dependency features), according to our knowledge, before there was no general methodology for incorporation of heterogenous features in an unsupervised WSD model.

The single-feature model based on dependency words proved to be most robust across tested datasets. As to the combination variants, we found it advantageous to combine all four types of features considered in our experiments. Combining models on the feature level yields highest F-scores in comparison to the meta-combinations. However, the meta-combination based on sum of confidences yields the most robust results across the datasets. Besides, the meta-combination based on sum of ranks provides higher precision at the cost of recall.

Experiments on a SemEval dataset, show that our approach performs comparably to the state-of-the-art unsupervised systems. Besides, the method perform almost no errors in the case of coarse-grained homonymous senses.

Implementation of our approach with several pre-trained models is available online.[5]

Acknowledgments

We acknowledge the support of the Deutsche Forschungsgemeinschaft (DFG) foundation under the project "JOIN-T: Joining Ontologies and Semantics Induced from Text".

[5] https://github.com/tudarmstadt-lt/JoSimText

References

Eneko Agirre and Philip Edmonds. 2007. *Word sense disambiguation: Algorithms and applications*, volume 33. Springer Science & Business Media.

Satanjeev Banerjee and Ted Pedersen. 2002. An adapted Lesk algorithm for word sense disambiguation using WordNet. In *Proceedings of the Third International Conference on Intelligent Text Processing and Computational Linguistics*, pages 136–145, Mexico City, Mexico.

Sergey Bartunov, Dmitry Kondrashkin, Anton Osokin, and Dmitry Vetrov. 2016. Breaking sticks and ambiguities with adaptive skip-gram. In *Proceedings of the AISTATS Conference*, Granada, Spain.

Osman Baskaya, Enis Sert, Volkan Cirik, and Deniz Yuret. 2013. AI-KU: Using Substitute Vectors and Co-Occurrence Modeling for Word Sense Induction and Disambiguation. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, volume 2, pages 300–306, Atlanta, Georgia, USA.

Chris Biemann and Martin Riedl. 2013. Text: Now in 2D! a framework for lexical expansion with contextual similarity. *Journal of Language Modelling*, 1(1):55–95.

Chris Biemann. 2006. Chinese Whispers: An Efficient Graph Clustering Algorithm and Its Application to Natural Language Processing Problems. In *Proceedings of the First Workshop on Graph Based Methods for Natural Language Processing*, pages 73–80, New York City, USA.

Chris Biemann. 2010. Co-Occurrence Cluster Features for Lexical Substitutions in Context. In *Proceedings of the 5th Workshop on TextGraphs in conjunction with ACL*, pages 55–59, Uppsala, Sweden.

Chris Biemann. 2012. Turk Bootstrap Word Sense Inventory 2.0: A Large-Scale Resource for Lexical Substitution. In *Proceedings of the 8th International Conference on Language Resources and Evaluation*, pages 4038–4042, Istanbul, Turkey.

Martin Everett and Stephen P Borgatti. 2005. Ego network betweenness. *Social networks*, 27(1):31–38.

Adriano Ferraresi, Eros Zanchetta, Marco Baroni, and Silvia Bernardini. 2008. Introducing and evaluating ukWaC, a very large web-derived corpus of English. In *Proceedings of the 4th Web as Corpus Workshop (WAC-4): Can we beat Google*, pages 47–54.

David Hope and Bill Keller. 2013a. MaxMax: A Graph-based Soft Clustering Algorithm Applied to Word Sense Induction. In *Proceedings of the 14th International Conference on Computational Linguistics and Intelligent Text Processing - Volume Part I*, pages 368–381, Samos, Greece. Springer-Verlag.

David Hope and Bill Keller. 2013b. UoS: A Graph-Based System for Graded Word Sense Induction. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, number 1, pages 689–694, Atlanta, GA, USA.

Eric H. Huang, Richard Socher, Christopher D. Manning, and Andrew Y. Ng. 2012. Improving word representations via global context and multiple word prototypes. In *Proceedings of the ACL*, pages 873–882, Jeju Island, Korea.

David Jurgens and Ioannis Klapaftis. 2013. Semeval-2013 task 13: Word sense induction for graded and non-graded senses. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, volume 2, pages 290–299, Atlanta, Georgia, USA.

Dan Klein, Kristina Toutanova, H. Tolga Ilhan, Sepandar D. Kamvar, and Christopher D. Manning. 2002. Combining Heterogeneous Classifiers for Word-Sense Disambiguation. In *Proceedings of the ACL-02 Workshop on Word Sense Disambiguation: Recent Successes and Future Directions*, volume 8, pages 74–80, Philadelphia, PA, USA.

Jey Han Lau, Paul Cook, and Timothy Baldwin. 2013. unimelb: Topic Modelling-based Word Sense Induction. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, volume 2, pages 307–311, Atlanta, Georgia, USA.

Yoong Keok Lee and Hwee Tou Ng. 2002. An empirical evaluation of knowledge sources and learning algorithms for word sense disambiguation. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing - EMNLP '02*, volume 10, pages 41–48, Philadelphia, PA.

Michael Lesk. 1986. Automatic Sense Disambiguation Using Machine Readable Dictionaries: How to Tell a Pine Cone from an Ice Cream Cone. In *Proceedings of the 5th annual international conference on Systems documentation*, pages 24–26, Toronto, ON, Canada. ACM.

Jiwei Li and Dan Jurafsky. 2015. Do multi-sense embeddings improve natural language understanding? In *Conference on Empirical Methods in Natural Language Processing, EMNLP'2015*, pages 1722–1732, Lisboa, Portugal.

Dekang Lin. 1998. An information-theoretic definition of similarity. In *Proceedings of ICML*, volume 98, pages 296–304, Madison, WI, USA.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. In *Workshop at International Conference on Learning Representations (ICLR)*, pages 1310–1318, Scottsdale, AZ, USA.

George A. Miller, Claudia Leacock, Randee Tengi, and Ross T. Bunker. 1993. A Semantic Concordance. In *Proceedings of the Workshop on Human Language Technology - HLT '93*, pages 303–308, NJ, USA.

George A Miller. 1995. WordNet: a lexical database for English. *Communications of the ACM*, 38(11):39–41.

Andrea Moro, Alessandro Raganato, and Roberto Navigli. 2014. Entity linking meets word sense disambiguation: a unified approach. *Transactions of the Association for Computational Linguistics*, 2:231–244.

Roberto Navigli. 2009. Word sense disambiguation: A survey. *ACM Computing Surveys (CSUR)*, 41(2):10.

Arvind Neelakantan, Jeevan Shankar, Alexandre Passos, and Andrew McCallum. 2014. Efficient non-parametric estimation of multiple embeddings per word in vector space. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1059–1069, Doha, Qatar.

Luis Nieto Piña and Richard Johansson. 2016. Embedding senses for efficient graph-based word sense disambiguation. In *Proceedings of TextGraphs-10, Proceedings of the Human Language Technology Conference of the NAACL*, pages 1–5, San Diego, USA.

Alexander Panchenko, Pavel Romanov, Olga Morozova, Hubert Naets, Andrey Philippovich, Alexey Romanov, and Cédrick Fairon. 2013. Serelex: Search and visualization of semantically related words. In *European Conference on Information Retrieval*, pages 837–840. Springer.

Patrick Pantel and Dekang Lin. 2002. Discovering word senses from text. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 613–619. ACM.

Ted. Pedersen and Rebecca Bruce. 1997. Distinguishing word senses in untagged text. In *Proceedings of the Second Conference on Empirical Methods in Natural Language Processing*, pages 197–207, Providence, RI, USA.

Ted Pedersen, Satanjeev Banerjee, and Siddharth Patwardhan. 2005. Maximizing semantic relatedness to perform word sense disambiguation. *University of Minnesota supercomputing institute research report UMSI*, 25:2005.

Martin Riedl. 2016. *Unsupervised Methods for Learning Semantics of Natural Language*. Ph.D. thesis, Technische Universität Darmstadt, Darmstadt, Germany.

Hinrich Schütze. 1998. Automatic Word Sense Discrimination. *Computational Linguistics*, 24(1):97–123.

Yee Whye Teh, Michael I Jordan, Matthew J Beal, and David M Blei. 2006. Hierarchical Dirichlet Processes. *Journal of the American Statistical Association*, 101(476):1566–1581.

Fei Tian, Hanjun Dai, Jiang Bian, Bin Gao, Rui Zhang, Enhong Chen, and Tie-Yan Liu. 2014. A probabilistic model for learning multi-prototype word embeddings. In *COLING*, pages 151–160, Dublin, Ireland.

Heng Low Wee. 2010. Word Sense Prediction Using Decision Trees. Technical report, Department of Computer Science, National University of Singapore.

Dominic Widdows and Beate Dorow. 2002. A graph model for unsupervised lexical acquisition. In *Proceedings of the 19th international conference on Computational linguistics*, pages 1–7, Taipei, Taiwan.

Deniz Yuret. 2012. FASTSUBS: An efficient and exact procedure for finding the most likely lexical substitutes based on an n-gram language model. *IEEE Signal Processing Letters*, 19(11):725–728.

Developing a Toolkit for Distributional Analysis of Abnormal Collocations in Russian

Polina Panicheva
St. Petersburg State University
St. Petersburg, Russia
ppolin86@gmail.com

Olga Mitrofanova
St. Petersburg State University
St. Petersburg, Russia
oa-mitrofanova@yandex.ru

Abstract

We propose a distributional approach to automatic correction of abnormal collocations in a Russian text corpus containing different types of erroneous word combinations, in particular, construction blending. We develop a toolkit which uses syntactic bigrams from RNC Sketches as training data and Word2Vec semantic model. A corpus of Russian Student Texts with annotation of erroneous word combinations, parsed morpho-syntactically with TreeTagger and MaltParser, was used in experiments. The annotated construction blending errors have been analyzed in terms of error correction by automatically proposing substitution candidates. The correction algorithm involves a set of association metrics based on context selectional preferences and semantic modeling, allowing to rank substitution candidates by their acceptability. Experimental results with nouns annotated as construction blending errors demonstrate the effectiveness of our toolkit. The results show that co-occurrence and Word2Vec semantic models perform ranking of the candidates in terms of different principles: purely constructional and semantic. As a result, the use of Word2Vec semantic filtering improves the quality of error correction.

1 Introduction

The goal of the paper is to model abnormal collocations and correct them automatically. Theoretically abnormal collocation is understood in terms of violation of a syntagmatic relation in a text (i.e., 'You have to try the national ham – jamon'). The abnormal collocation correction model is based on the assumption that a keyword presenting collocation abnormality can be substituted by a word fitting the current context better, while being semantically similar to the initial keyword. In practice, we present an algorithm for automatic correction of abnormal collocations by substituting the keyword with the most frequent word in the given context.

The abmormal collocations are provided by the Corpus of Russian Student Texts (CoRST), (Zevakhina and Dzhakupova, 2015), which consists of educational essays on various topics written by native speakers of Russian. The corpus is annotated, among others, with lexical errors caused by construction blending (Puzhaeva et al., 2015), which involves merging of structural features of different constructions (e.g. '*играть роль*' (*to play the role*) + '*занимать место*' (*to take a seat, to replace*) = *'*играть место*' (**to play a seat*)). Blended constructions present a case of abnormal collocations, as they contain at least one word which is untypical in the current context and can be replaced by a semantically similar word to form a proper construction. Moreover, blended constructions present a subtle case of abnormal collocations, as the former are produced by fluent native speakers, and the overall utterance stays meaningful in spite of the blending.

In order to provide a model of abnormal collocation correction we address the following issues:

1. A set of annotated errors by native speakers caused by construction blending is extracted from CoRST;
2. A syntactic-based co-occurrence model is applied to identify and rank substitutes in the blended constructions; a word-embeddings semantic model is added to measure semantic similarity;
3. The construction blending errors are automatically corrected by the proposed model.

2 Related work

Distributional semantic approaches have been applied to identification of a broader scope of lexical anomalies, i.e., metaphor (Shutova,

2010), semantic deviance (Vecchi et al., 2015) and learner errors (Kochmar and Briscoe, 2013). We follow (Shutova, 2010) in applying the Context-Based Paraphrasing weighting algorithm to identifying and ranking possible substitution candidates. However, the difference of our work is that CBP is based on collocation counts with individual words for error identification, while Shutova (2010) analyzes word clusters for a more abstract metaphoric usage. Kochmar and Briscoe (2013) apply features derived from word-embeddings semantic models to identify learner errors, whereas in our case word-embeddings have to be combined with a fine-grained syntax-based CBP model to handle subtle errors.

A common feature of the mentioned works is that training and test data are constructed from relatively frequent keywords of English, and context words are added according to a restricted list of syntactic relations (attribute, verb subject or/and object). The crucial difference of the current work is that both training and test data consist of unrestricted corpora containing all possible syntactic relations, thus rendering the task closer to a real-life problem. This causes obvious difficulties, such as word and collocation sparsity and imbalance issues. There are also important restrictions imposed by Russian NLP resources, with the syntactic bigram statistics available only in terms of SynTagRus syntactic relations (Boguslavsky et al., 2002), restricting the relevant morpho-syntactic algorithms to TreeTagger and MaltParser (Sharoff et al., 2008; Sharov and Nivre, 2011). It is also noteworthy that current training and test corpora belong to different genres, rendering the task genre-independent.

Our work is the first attempt to automatically approach an unrestricted (by frequency or syntactic properties) corpus of real-world lexical errors. To our knowledge, it is the first approach to lexical anomalies in Russian texts by native speakers. The datasets of syntactic and distributional variety have been applied for model training. The novelty of the method involves combining syntactic count-based and word-embeddings distributional models.

3 Toolkit design

The aim of the toolkit is to analyze, correct and, to an extent, identify word collocations, with anomalies in contextual restrictions caused by creative language processing in metaphor, violation of fine-grained selectional restrictions in the texts of language learners and native speakers, pronounced mistakes caused by speech impairment. The basic assumption is that a coherent text complies with the requirements concerning semantic and selectional restrictions on syntagmatic relations between words. Technically it is rendered by the idea that a word basically occurs in contexts in which it has already occurred frequently, or in some sense similar ones. The system thus learns co-occurrence regularities from text corpora and processes a keyword and its context, measuring their mutual association and proposing substitutes for the keyword where possible.

The toolkit is expected to work in two settings. First, it should provide analysis and substitutes for words annotated as abnormal in a text. This setting is applied in the current work. Second, it should be able to automatically identify some abnormal words in collocations. The latter goal is a subject of future work.

3.1 Input

Fine-grained selectional restrictions analysis requires either very large datasets or syntactic processing. In Russian, morphological analysis is required in both settings. Bag-of-Words models, as Word2Vec, do not require any further parsing, but offer paradigmatic-oriented insight which is difficult to interpret and tune in a syntagmatic collocational setting. While syntactically parsed corpora are difficult to obtain, they provide fine-grained information which is indispensable when identifying the nature of syntagmatic violation.

As a training corpus we use the RNC Sketches syntactic bigram statistics[1]. It provides statistics on syntactic relations based on a sample of the Russian National Corpus (RNC) of 200M words, where every keyword is associated with a list of its relations and their frequencies. A syntactic relation is a pair *(relation, word)*, where the relations inventory is that of the SynTagRus corpus (Boguslavsky et al., 2002), and the word is the dependent word, e.g. '*попробовать (try) -> 1st completive -> себя (oneself) : 126*', '*попробовать (try) -> 1st completive -> блюдо (dish) : 7*', '*национальный (national) -> attrib -> идея (idea) : 390*', '*национальный (national) -> attrib -> блюдо (dish) : 55*'. An additional step of reverting the syntactic relations is required to obtain source words for every dependent keyword. In order to unify the format of the training data and the data used for error analysis,

[1] http://ling.go.mail.ru/synt/

we apply MaltParser and TreeTagger used to create RNC Sketches (Sharoff et al., 2008; Sharov and Nivre, 2011) to the testing data. Individual word frequencies were obtained from the Russian Frequency Dictionary (Lyashevskaya and Sharov, 2009). We also supply our algorithm with a Word2Vec semantic model based on RNC (Kutuzov and Andreev, 2015).

The data used for automatic error analysis is provided by the Corpus of Russian Student Texts (CoRST). It contains educational texts by native speakers of Russian (500K words) annotated with a broad range of errors (10K annotated errors). The errors caused by construction blending (Puzhaeva et al., 2015) are especially relevant to our task, as they present subtle violations of selectional restrictions.

3.2 Statistical models

We use the RNC Sketches syntactic bigrams as a syntactic model and apply automatic ranking of the erroneous keywords based on their context. The list of possible substitutes for a particular keyword is generated as the list of words occurring in the bigram corpus in the same syntactic context as keyword. Namely, it is the intersection of the words occurring with every syntactic relation in the keyword context. The substitutes are commonly ranked using the following association measure scores: Mutual Information scoring (Khokhlova, 2008), context-based paraphrasing (CBP) (Shutova, 2010), Resnik's selectional association based on Kullback-Leibler distance (Resnik, 1993), and Word2Vec-based semantic scoring (Kutuzov and Andreev 2015). The likelihood *L* of a particular paraphrase *i* of the word *w* is estimated as the likelihood of the joint events: the substitute *i* co-occurring with all the other lexical items from its context $w_1, \ldots w_N$ in syntactic relations $r_1, \ldots r_N$.

Context-Based Paraphrasing: The context-based paraphrasing likelihood estimation is based on syntactic co-occurrence:

$$L_i(CBP) = \frac{\prod_{n=1}^{N} f(w_n, r_n, i)}{(f(i))^{N-1}}.$$

Word2vec Semantic Scoring: In order to account for purely semantic word properties, i.e. restrict the list of substitutes to words semantically similar to the keyword, we apply the Word2Vec model trained with RNC data. Semantic similarity between a keyword *kw* and it's substitute *i* is calculated as the cosine distance between the corresponding vectors in the Word2Vec semantic space:

$$Sim(kw, i) = \cos(kw, i).$$

4 Experimental setup

We perform a proof-of-concept experiment by automatically correcting the errors caused by construction blending in CoRST with context-based paraphrasing and additional Word2Vec semantic scoring. The errors are made by native speakers and represent violations of selectional restrictions. There are 130 lexical errors in the corpus caused by construction blending. We have extracted 29 sentences from the corpus, containing a noun annotated as a lexical error caused by construction blending. We set out to automatically suggest a list of substitutes for the erroneous nouns and score them according to the Context-Based Paraphrasing procedure. We also perform Word2Vec semantic filtering to improve the results.

№	Example sentence	Syntactic context		Weighted substitutes		Evaluation result	
		Relation	Word	Candidate	Likelihood	Strict	Loose
1	Между нравами и законами трудно провести четкое **различие**. - It's hard to draw a strict **difference** between customs and laws.	1st completive	провести - draw	**линия - line** **грань - border** **разграничение - distinction** **граница - boundary**	82.5 60.4 49.1 42.9	Corr	Corr
		attrib	четкий - strict				
2	Обязательно попробуйте национальный **окорок** – хамон, … - You have to try the national **ham** – jamon, …	1st completive	попробовать - try	сила - power **блюдо - dish** напиток - drink **продукт - product**	21.0 12.3 9.5 2.7	Inc	Corr
		attrib	национальный - national				
3	приходится платить за каждый аттракцион и из-за их дорогой **стоимости**… - one has to pay for every attraction, and because of their high **price** …	prepos	из-за - because of	черта - feature отношение - relation страх - fear лес - forest	0.02 0.0009 0.0008 0.0004	Inc	Inc
		quasi-agent	они - they				
		attrib	дорогой- high				

Table 1. Examples of context-based paraphrasing results.

We calculate the accuracy of the results by applying manual evaluation. A substitute candidate is marked correct if it fits the context better than the erroneous keyword and leaves the meaning of the sentence unchanged. The resulting lists of candidates contain up to 50 ranked words. The assumption is that the highest ranked words represent the best substitution candidates in the provided contexts. It is examined by manually analyzing a short-list of top candidates. Evaluation is performed in two settings:

1. The **strict mode** implies that the substitutes provided by the algorithm are correct if the candidate with the highest rank is correct.
2. The **loose mode** renders the substitutes list correct if there is a correct candidate among the four highest ranked candidates.

We do not perform further evaluation procedures at this stage, because the initial proof-of-concept experiment is aimed at providing an overall insight on the task, its restrictions and improvement possibilities.

5 Results and discussion

5.1 Context-based paraphrasing

Out of 29 sentences, 4 contained morphological and syntactic annotation errors in the morpho-syntactic analysis of the erroneous nouns, which made the list of the candidates provided by CBP empty. The rest of the examples, 25 sentences, were processed with CBP substitute ranking.

Out of 25 examples, the algorithm provided **15 (60%)** correct substitutes in the loose mode and **10 (40%)** in the strict mode. The results of the substitution experiment are exemplified in Table 1. Analysis shows that among the 10 loose-mode incorrect results, 5 are defined by the syntactic context which doesn't allow retrieving any meaningful candidates: there is a very limited number of candidates co-occurring with all the context features in the corpus, and their meaning is either too broad or too distant from that of the original keywords (for example, in '*это было обусловлено православной религией'* (*it was preconditioned by orthodox religion*) substitutes for '*религия*' (*religion*) only include '*образование*' (*education*), '*организация*' (*organization*)). However, strict mode-specific mistakes include correct substitutes, which are downgraded in their rank by the words fitting the syntactic context very well but bearing a meaning unrelated to the keywords (see ex. 2 in Table 1). These cases could be improved by adding purely semantic information to the model.

5.2 Semantic filtering

Shutova (2010) performs semantic filtering based on WordNet by limiting the paraphrasing candidates to those in hypernym or co-hyponym relations with the keyword restricted to three-level distance. In order to avoid sparsity of data covered by hand-coded resources, we apply RNC-based Word2Vec model as a semantic filter to eliminate substitution candidates unrelated to the keyword. The semantic similarity threshold

№	Example sentence	Weighted substitutes	
		No filtering	Word2Vec semantic filtering
1	… Между нравами и законами трудно провести четкое **различие**. - It's hard to draw a strict **difference** between customs and laws.	**линия - line** **грань - border** **разграничение - distinction** **граница - boundary**	**грань - border** **разграничение - distinction** **граница - boundary** параллель - parallel
2	Если рассматривать этот вопрос с религиозной **стороны** то тут тоже тяжело найти оправдание. – Looking at the issue from the religious **side**, …	**точка - point** **позиция - position** начало - start язык - language	**точка - point** **позиция - position** конец - end
3	Поэтому отдых на Байкале … помогает человеку снова набраться жизненной **силой**. – Holiday at Baikal … helps one to collect life **power**.	опыт - experience впечатление - impression **энергия - energy** дух - spirit	опыт - experience **энергия - energy** дух - spirit мудрость - wisdom
4	… круглые сироты, не имеющие в целом **свете** ни единого родственника? – … total orphans, having no relatives in the whole **world(1)**?	ряд - row **мир – world(2)** арсенал - arsenal район - region	**мир – world(2)** **жизнь - life** страна - country город - city
5	Обязательно попробуйте национальный **окорок** – хамон, … - You have to try the national **ham** – jamon, …	сила - power **блюдо - dish** напиток - drink **продукт - product**	**блюдо - dish** напиток - drink **продукт - product** **лакомство - delicacy**
6	… люди, ставящие перед собой высокие **рамки** – … people who set high **limits**	**цель - goal** оценка - mark честь - honour точка - point	**цель - goal** **планка - bar** барьер - barrier положение - position

Table 2. Differences between the results with and without semantic filtering.

value is experimentally set to **0.1**.

As expected, filtering results in slight improvement in loose mode evaluation, correctly analyzing **18** examples (**72%**). However, it gives considerably higher results in the strict mode, eliminating semantically unrelated candidates and ranking correct substitutes higher: accuracy evaluated in the strict mode is **14** examples **(56%)**. Table 2 illustrates the meaningful differences between the substitution results with and without semantic filtering, with the keywords in the example sentences and the correct substitutes highlighted in bold and the results crucial for the strict mode performance also underlined. It is important to notice that semantic filtering also improves the performance beyond the first-rank candidate by making qualitative modifications to the candidate list: it reduces the number of low-likelihood substitutes (ex. 2), increases the rank of correct substitutes and their proportion in the four highest-ranked candidates (ex. 3, 5, 7).

6 Conclusions and future work

We have introduced a toolkit for abnormal collocation analysis and automatic correction. The toolkit applies collocation-based association measures aimed at analyzing various types of context restriction violations. We have performed a proof-of-concept experiment with construction blending errors by native speakers of Russian, which confirms the applicability of the statistical association measures to this task. Close analysis of algorithm errors has revealed the need for semantic restrictions, which cannot be accounted for by purely context-based methods. Adding Word2Vec-based semantic filtering has improved the results qualitatively and in terms of accuracy, making the incorporation of various language models a promising approach in analyzing abnormal associations. Another crucial point in this task is accurate and consistent morpho-syntactic analysis of training and test corpora.

Our future work includes adding more data to the analysis (other parts of speech annotated in CoRST) and processing anomalies of a different nature: learner errors, intentional semantic deviance in figurative language, errors caused by language impairment.

7 Future considerations

An important finding of the current experiment is the need for combination of fine-grained syntactic and distributional semantic models. The combination is expected to play a crucial role in future analysis of different error types. As shown in current research, native speaker errors present subtle co-occurrence violations while basically maintaining the meaning of the keyword comparing to its correct substitute. However, we expect a different trade-off between syntactic co-occurrence and semantics in other types of errors. It appears that the higher the level of a language learner, the more the erroneous combinations maintain their basic meaning; whereas the lack of immediate experience with fluent text is reflected in co-occurrence violations, regardless of language proficiency level.

Figurative text has been shown to contain semantic violations of a specific type, as in metaphor, where the meaning of a source domain is projected onto a different target domain (Shutova, 2010). Metaphor presents errors violating the basic semantic restrictions, but requiring a more abstract semantic analysis based on word clusters and domains. On the contrary, speech impairment is expected to produce semantic violations with no underlying abstract pattern or with a pattern fundamentally different from that identified in figurative language.

Acknowledgments

The reported study is supported by RFBR grant 16-06-00529 "Development of a linguistic toolkit for semantic analysis of Russian text corpora by statistical techniques".

References

Igor Boguslavsky, Ivan Chardin, Svetlana Grigorjeva, Nikolai Grigoriev, Leonid Iomdin, Leonid Kreydlin, and Nadezhda Frid. 2002. Development of a dependency treebank for Russian and its possible applications in NLP. In *Proceedings of the Third International Conference on Language Resources and Evaluation (LREC-2002)*, vol. III, pages 852–856

Maria Khokhlova. 2008. Extracting collocations in Russian: Statistics vs. dictionary. In *JADT 2008: 9es Journ´ees internationales dAnalyse statistique des Donn´ees*, pages 613–624.

Ekaterina Kochmar and Ted Briscoe. 2013. Capturing anomalies in the choice of content words in compositional distributional semantic space. In *RANLP*, pages 365–372.

Andrey Kutuzov and Igor Andreev. 2015. Texts in, meaning out: neural language models in

semantic similarity task for Russian. *arXiv preprint arXiv:1504.08183*.

George Lakoff and Mark Johnson. 2008. *Metaphors we live by*. University of Chicago press.

Omer Levy and Yoav Goldberg. 2014. Dependency-based word embeddings. In *ACL (2)*, pages 302–308.

Olga Lyashevskaya and Sergey Sharov. 2009. Chastotnyy slovar'sovremennogo russkogo yazyka (na materialakh natsional'nogo korpusa russkogo yazyka) [The frequency dictionary of modern Russian (on the materials of the Russian National Corpus)]. *Moscow: Azbukovnik Publ*.

Svetlana Puzhaeva, Natalia Zevakhina, and Svetlana Dzhakupova. 2015. Construction blending in non-standard variants of Russian in the Corpus of Russian Student Texts. In *Proceedings of the 6th International Conference "Corpus Linguistics-2015"*, 390-397. Saint-Petersburg. (in Russian)

Philip Stuart Resnik. 1993. Selection and information: a class-based approach to lexical relationships. *IRCS Technical Reports Series*, page 200.

Serge Sharoff, Mikhail Kopotev, Tomaz Erjavec, Anna Feldman, and Dagmar Divjak. 2008. Designing and evaluating a Russian tagset. In *LREC*.

Sergey Sharov and Joakim Nivre. 2011. The proper place of men and machines in language technology. Processing Russian without any linguistic knowledge. In *Proceedings of the Annual International Conference Dialogue, Computational Linguistics and Intellectual Technologies*, number 10, page 657.

Ekaterina Shutova. 2010. Automatic metaphor interpretation as a paraphrasing task. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 1029–1037. Association for Computational Linguistics.

Eva M Vecchi, Marco Marelli, Roberto Zamparelli, and Marco Baroni. 2015. Spicy adjectives and nominal donkeys: Capturing semantic deviance using compositionality in distributional spaces.

Natalia Zevakhina and Svetlana Dzhakupova. 2015. Corpus of Russian student texts: design and prospects. In *Proceedings of the 21st International Conference on Computational Linguistics "Dialog"*. Moscow, 2015.

Verb lemmatization and semantic verb classes in a Middle English corpus

Michael Percillier
Universität Mannheim
Anglistische Linguistik/Diachronie
L13, 9, 68131 Mannheim, Germany
percillier@uni-mannheim.de

Abstract

The paper describes the creation of new resources and associated tools in the framework of the research project *Borrowing of Argument Structure in Contact Situations* (BASICS), which investigates the borrowing of argument structures of verbs from Old French (OF) to Middle English (ME). The first resource is a database of ME form-lemma correspondences, on which a lemmatization process is based. This process also identifies French-based verbs and thus enables a first diachronic analysis of their prevalence in ME. The second item discussed is a newly developed method for querying ME verbs according to their semantic classes. The created resources and methods are crucial in the continuation of the research project, and can be applied to annotate further ME corpora and train other tools for the treatment of ME data.

1 Introduction

This paper is part of a research project[1] that investigates grammatical change in the language contact situation between Middle English (ME) and Old French (OF) that set in after the Norman Conquest (1066) and lasted until ca 1500. More specifically, the project focuses on the connection between the lexical borrowing of verbs and the transfer of their argument structures (AS) from the source language OF to the recipient language ME.

One of the objectives of the project is to trace the spread of AS from borrowed verbs, i.e. verbs originally from OF, to native verbs, i.e. verbs already part of the English lexicon prior to the language contact situation. To this end, corpus queries of syntactic structures need to be complemented by searches for specific verbs and semantic verb classes. The focus on specific verbs allows for a detailed comparison of native and borrowed verbs, while the focus on semantic verb classes will make it possible to follow the spread of syntactic structures from borrowed verbs to other verbs sharing similar meanings.

The currently available resources, described in Section 2, are geared towards queries of syntactic structures, but not specific verbs, let alone semantic verb classes. In order to fulfill the needs of the project, the existing resources have to be enhanced in two ways: (1) the extension of existing annotation with lemma information for verbs, and (2) a method for determining semantic classes of ME verbs. The implementation of both enhancements is described in this paper, as well as their possible application on a recent study of the French borrowing *please* (Trips and Stein, accepted).

2 Currently available resources

The *Oxford English Dictionary* (Proffitt, editor, 2015), abbreviated as OED, serves as a point of reference for the project, not only because it is an authoritative resource on the English lexicon, but also because it contains a wealth of etymological information. Owing to a cooperation in the project with the OED's principal etymologist Philip Durkin, we were able to obtain a list of 2,026 English verbs borrowed from French between 1066 and 1500 based on an explicit query. The verbs in said list constitute the starting point of the project, as they are the loan words whose AS is thus introduced to English and can thereafter extend to other verbs.

The ways in which these loan verbs were used should be verified empirically in a corpus. For ME, the *Penn-Helsinki-Parsed-Corpus of Middle English* (Kroch and Taylor, 2000), henceforth PPCME2, presents the advantage of being syntac-

[1] *Borrowing of Argument Structure in Contact Situations: The Case of Medieval English under French influence* (BASICS).

tically annotated. The corpus consists of 55 texts, totaling ca 1.2 million words, and is divided into four periods: M1 (1150–1250), M2 (1250–1350), M3 (1350–1420), and M4 (1420–1500).[2] The annotation format used is *Penn-Treebank*, which can be queried using the specialized software tool *CorpusSearch* (Randall, 2010). The format uses sets of parentheses to represent the clause hierarchy, as illustrated for Modern English in the example below.[3]

```
( (IP-MAT (ADVP-TMP (ADV Then))
          (NP-SBJ (D the)
                  (N child))
          (VBD became)
          (ADJP (ADJR happier)
                  (CONJ and)
                  (ADJR happier))
          (E_S .)) )
```

At the lowest level of the tree hierarchy, each form is assigned a part-of-speech (POS) tag. Consequently, the annotation format, in combination with *CorpusSearch*, makes it possible to search for specific grammatical properties, such as past tense verbs using the *VBD* tag, or specific forms such as *became*. However, due to frequent spelling variation in ME data and the existence of irregular verb paradigms, queries for all forms of a verb, such as *become*, are not readily available by searching for verb stems in ME corpora. To remedy this, all lexical verb forms in the PPCME2 are to be lemmatized, a process described in Section 3.

For the definition of semantic verb classes, the model proposed by Levin (1993), which groups lexical verbs on a semantic basis, can be used as a point of reference. The advantage over other semantic resources such as *WordNet* (Princeton University, 2010) lies in the listing of possible syntactic alternations for each verb class. However, the model applies to Present Day English (PDE) and cannot be directly applied to ME for a number of reasons: (1) semantic changes occurred from ME to PDE, so that the classification proposed by Levin (1993) may be inaccurate for certain ME verbs, (2) ME verbs that no longer exist in PDE are not included in Levin's classification, so that a direct application of the model to ME would result in only partial coverage, (3) a number of PDE verbs did not yet exist in ME and are therefore irrelevant in the definition of ME verb classes, and (4) the potential of syntactic alternations cannot be postulated on the basis of intuition for earlier periods.

In addition to the OED, the *Middle English Dictionary* (McSparran et al., 2001), henceforth MED, constitutes a further dictionary resource that is relevant for the lemmatization of a ME corpus and the definition of ME semantic verb classes. The MED uses unique numerical identifiers (henceforth MED-IDs) for each entry that can serve to disambiguate homonyms. Furthermore, entries in the MED and the OED are linked, so that using both resources in tandem makes it possible to distinguish between native and borrowed ME verbs by checking them against the list of verbs borrowed from French provided by the OED.

3 Lemmatization of a ME corpus

As previously stated, the lemmatization of a ME corpus, in particular of its verbs, is a crucial step for any study in which queries of specific verbs or semantic verb classes are to be undertaken. Given the absence of lemmatized ME corpora or any gold standard for the lemmatization of ME data, the lemmatization process relies on the semi-manual assignment of graphemic verb forms to their respective lemmas. The process is divided into two major steps: (1) the creation of an inventory of form-lemma correspondences linking forms in the PPCME2 to lemmas in the MED, and (2) the insertion of this lemma information into the corpus.

3.1 Assignment of form-lemma correspondences

Verb forms were extracted from the PPCME2, and each verb form was paired with a lemma and the corresponding ID extracted from the MED. This assignment of verb forms to lemmas was undertaken manually by four trained research assistants and the author using a spreadsheet application. They also had the option of specifying multiple lemmas or marking their choices as doubtful. In total, 19,320 graphemic verb forms were assigned to 2,979 lemmas as primary matches, alongside 4,973 lemmas specified as additional possible matches. The resulting form-lemma links were exported to the YAML (Evans, 2009) format, which was chosen so as to allow the data to be easily imported as

[2] Information from http://www.ling.upenn.edu/hist-corpora/PPCME2-RELEASE-4/description.html.

[3] Example adapted from http://corpussearch.sourceforge.net/format.html.

a hash/dictionary in any programming language.[4]

3.2 Insertion of lemma information into the corpus

Using the inventory of form-lemma correspondences just mentioned, the insertion of lemma information is performed. For every verb marked with a POS tag beginning with *V* in the corpus,[5] the following instructions are carried out:

The main approach is a lexical lookup in the inventory of form-lemma correspondences. Should this not return any results, two fallbacks are used: (1) Spelling variants are generated and queried for corresponding lemmas. The following grapheme substitution rules are used: $i \rightarrow e/y$, $e \rightarrow i$, $y \rightarrow i/g/+g$, $u \rightarrow v/ou$, $v \rightarrow u$, $th \rightarrow +t/+d$, $+t \rightarrow th$, $+d \rightarrow th$, $g \rightarrow +g/y$, $+g \rightarrow g/y$, $ou \rightarrow u$, $ll \rightarrow l$, $nn \rightarrow n$, and $pp \rightarrow p$.[6] Further, forms containing hyphens or tildes are assigned spelling variants without these characters. (2) The form is stemmed and checked against all stemmed forms in the form-lemma inventory. Stemming is achieved by removing the following ME inflectional suffixes: $+d$, $+d+d$, $+t$, $+t+t$, *an*, *ande?*, *dd?*, *den?*, *e*, $e+d$, $e+t$, *ede?*, *enn?*, *e?st*, *et*, *in?d?e?*, *ingg?e?*, *ode*, *odest*, *oden*, *ten?*, *th*, *tt?*, *yde?*, *ynde?*, *ynn?*, *yngg?e?*, and *yst*.[7]

The lemma information is appended directly to the form in the corpus, so as to still comply with the Penn-Treebank format and related software such as *CorpusSearch*. Each piece of inserted information is demarcated by @ characters and specified by an attribute. Verb lemmas are specified by the attribute *l*, and MED-IDs by the attribute *m* (see Example (1)). For verbs occurring in the list of French-based verbs, an additional attribute *e* (for *etymology*) is defined as *french* (see Example (2)). The attribute *w* (for *warning*) indicates that the lemma was matched using either the spelling substitution or the stemming method (see Examples (2)/(5) and (3) respectively), or that the manual form-lemma match was deemed doubtful (see Example (4)). For verbs spelt as multiple words, the information is appended to the final element (see Example (5)). Should no form-lemma correspondence have been found even after the stemming method, the lemma and MED-ID are marked as *NA* (see Example (6)). The lemma insertion process is summarized in Figure 1.

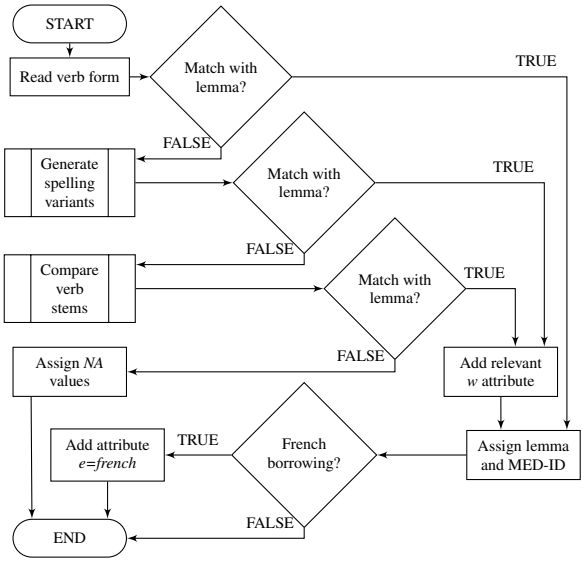

Figure 1: Lemma insertion process.

(1) `(VAG settyng@l=setten@m =39654@)`

(2) `(VAG consyderyng@l= consideren@m=9387@e=french@w =substitution@)`

(3) `(VB tellyn@l=tellen@m=44693 @w=stemming@)`

(4) `(VBI wilne@l=wilnen@m=52815 @w=doubt@)`

(5) `(VBP21 vnder)(VBP22 stont@l =understonden@m=48362@w= substitution@)`

(6) `(VAN iii@l=NA@m=NA@)`

With this additional annotation, the PPCME2 can be queried for syntactic structures as before, but also for specific verbs. Using *CorpusSearch*, this is achieved by specifying the lemma with the *exists* function, e.g. `(*l=setten@* exists)`. To distinguish between homonyms, the MED-ID can also be used for unambiguous queries, e.g. `(* m=39654@* exists)`.

3.3 Evaluation

The lemmatization of verbs in the PPCME2 treated 130,282 verbs in total. 110,116 verbs (84.52%) were directly assigned matching lemmas. Additionally, 5,868 verbs (4.5%) were assigned a lemma

[4] For example with the *PyYAML* (Simonov, 2014) module in *Python* (Python Software Foundation, 2015).

[5] Lexical *be*, *do*, and *have* need not be lemmatized as their tags (*B**, *D**, and *H** respectively) already reveal their lemma.

[6] In the PCCME2, the character sequences $+d$, $+g$, and $+t$ represent the graphemes <ð, ȝ, þ> respectively.

[7] Question marks refer to the regular expression quantifier specifying that the preceding character may or may not occur.

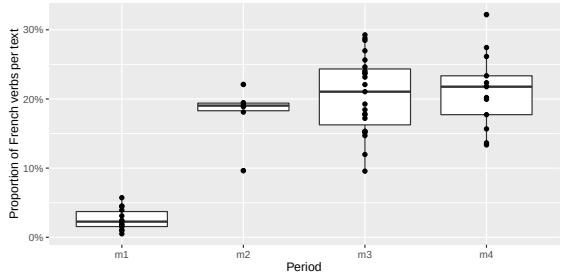

Figure 2: Proportion of French-based verbs in sub-periods of ME.

	M1	M2	M3
M2	1.0e-07	-	-
M3	8.2e-15	1	-
M4	2.6e-13	0.97	1

Table 1: Pairwise t-tests of French-based verbs per ME sub-period, using Bonferroni correction.

using spelling substitution, and 10,421 verbs (8%) using stem comparison. The total of lemmatized verbs is thus 126,405 (97.02%), whereas 3,877 verbs (2.98%) could not be assigned any lemma. Based on controls of random samples of 100 tokens, the spelling substitution and stem comparison fallbacks were estimated to be accurate to 86% and 90% respectively.

The estimation of French-based verbs and the division of ME into the sub-periods M1–M4 make it possible to investigate the diachronic spread of French-based verbs in ME (see Figure 2).[8] The analysis suggests a strong increase in the usage of French-based verbs between M1 and M2, with only little fluctuation thereafter. This is confirmed through pairwise t-tests (see Table 1) with Bonferroni correction (Baayen, 2008, 105–106).

4 Determining ME semantic verb classes

In order to identify ME semantic verb classes, the classification proposed by Levin (1993) can serve as a point of reference, but cannot be applied directly to ME, as already mentioned in Section 2. The estimation of ME equivalents to the semantic verb classes proposed by Levin (1993) is undertaken in three steps: (1) the creation of a database of semantic classes and the verbs therein from which verb lists can be extracted, (2) a method for finding ME verbs synonymous to the PDE verbs extracted in the previous step, and (3) a method for querying the corpus for multiple verbs simultaneously.

4.1 Creating an inventory of semantic verb classes

An electronic index of Levin (1993) exists as a HTML file,[9] but it only lists which verbs occur in which numbered section of the monograph, thereby omitting the names and descriptions of the classes entirely. An updated index was therefore generated that not only numbers but also names classes. This index can be queried with a script that parses the HTML tree[10] and allows for two types of searches: (1) by verb class, to determine which verbs occur in a given class, or (2) by verb, to determine to which verb classes a particular verb belongs.

4.2 Matching ME and PDE verb meaning

Determining ME equivalents to PDE verb classes proposed by Levin (1993) entails finding ME verbs synonymous to verbs listed in PDE classes. The MED allows a "reverse lookup" of ME verbs via its search engine[11] when specifying a PDE verb as a query within entry definitions, which returns a list of MED entries in which the query term occurs anywhere within the definition. For example, a reverse search for the PDE verb *acknowledge* returns ME verbs such as *agraunten* ('to acknowledge, grant'), *aknouen* ('to recognize (sth.) as a fact, acknowledge, know'), or *kithen* ('to acknowledge (sb.) as (sth.)'). A script automates the process by querying a given list of PDE verbs, then excluding any results that are not verbs.

This list of verbs requires manual verification for two reasons: (1) the presence of a verb in the list merely indicates that the query item was found within the definition, but not necessarily that the PDE and ME verbs are synonyms, and (2) the PDE verb used in a query may be polysemous, so that the PDE and ME lemmas may be correctly matched, but their specific meanings may differ. An example of the first point is a query of the PDE verb *crown* that returned the ME verb *cacchen* ('to catch') because the MED entry contains "**cacchen of**: take off (one's crown, etc.) quickly". In this case, the matched string referred to the noun *crown* as used in an example sentence, and therefore does not

[8] Figure generated in R (R Core Team, 2016) with *ggplot2* (Wickham, 2009) and *scales* (Wickham, 2016).

[9] `http://www-personal.umich.edu/~jlawler/levin.html`

[10] Using the module *BeautifulSoup* (Richardson, 2015).

[11] `http://quod.lib.umich.edu/m/med/structure.html`

constitute a valid match. The second point can be illustrated by the PDE verb *consider*, which is listed as a "verb with predicative-complements" by Levin (1993, 181), more specifically in the subclass "*appoint* verbs". However, this classification only applies to a specific meaning of *consider*, i.e. "to regard in a certain light or aspect", but not to other meanings such as "to think/contemplate". The separation of different meanings of polysemic verbs is crucial, given that they result in distinct AS (Löbner, 2002, 114–116). For this reason, valid matches for PDE verbs have to be checked for congruence with the specific meaning used in a given semantic class. ME verbs that fulfill these conditions can be considered as semantic equivalents to the PDE class defined as input for the query.

4.3 Querying multiple verbs

Simultaneous queries of multiple verbs can be specified in a `*.q` file that serves as input to *CorpusSearch*. The query language allows the logical operator `|` (OR), so that multiple MED-IDs can be searched, e.g. `(*m=9348@*|*m=9356@* exists)`. As the lists of ME verbs to be queried can be long, the creation of such query files is automated via a script that reads the list of MED-IDs from a column named "MED-ID" in a CSV table, then generates a corresponding `*.q` query file.

4.4 Application of the method

The proposed method of identifying ME semantic verb classes and querying the verbs in a simultaneous manner has direct applications for recent and ongoing studies.

For instance, Trips and Stein (accepted) empirically verified the assumptions proposed by Allen (1995) on the transfer of prepositional 'datives' from the French-based verb *plesen* ('to please') to the native verbs *liken* ('to like') and *quemen* ('to please'). They conclude that ME, having lost most of its formal case distinctions, adopted the 'dative' arguments of the donor language OF. The semantic properties of the borrowed verb *plesen* allowed the transfer of its AS, specifically the use of prepositional objects, to native verbs belonging to the same semantic class of verbs of psychological state, so-called *psych* verbs (Levin, 1993, 188–193). This transfer led to a rise in the use of prepositional objects with native psych verbs, with *quemen* ultimately replaced by *plesen*. The new structure eventually spread to native verbs belonging to other semantic classes, e.g. *yeven* ('to give').

The important findings presented by Trips and Stein (accepted) regarding the ME verbs *liken*, *quemen*, and *plesen* can be systematically verified for other ME psych verbs by using the proposed method of identifying ME semantic verb classes. Furthermore, the spread of new structures to verbs of other semantic classes, as in the case of *yeven*, can also be investigated by examining whether certain semantic classes adopted the new structures more frequently or more quickly than others.

5 Conclusions

The present paper discussed two enhancements to a parsed corpus of ME that are necessary for a project investigating AS borrowing from OF to ME. The first enhancement is the lemmatization of lexical verbs, so that queries for specific verbs can be performed in addition to searches for syntactic structures. The second enhancement builds upon the first in that it allows for the search for multiple verb lemmas at once, more specifically those belonging to a given semantic class. By identifying French-based verbs, the lemmatization process also enabled a diachronic analysis of the proportion of French-based verbs per ME sub-period.

6 Outlook

The two processes discussed are vital for the project at hand, as they clear a methodological "bottleneck", thus allowing searches for specific ME verbs and semantic classes to proceed. Furthermore, the analysis of the proportion of French-based verbs raises an interesting research question pertaining to the delay between the wider adoption of French-based verbs in M2 and the expansion of their AS to native verbs.

Although tailored to a specific project, the resources and methods have further applications. The form-lemma links and their lemmatizer script can be applied to other ME corpora, and a general lemmatizer for ME (i.e. not limited to verbs) can benefit from this inventory as a training resource. The index of PDE semantic verb classes and the method to adapt it to ME can be used to perform semantic verb class searches in both ME and PDE corpora.

7 Resources

The created resources and their associated tools are available via the *BASICS Toolkit* web application.[12]

[12] `http://terrano.philosophie.uni-stuttgart.de/BASICStoolkit`

Acknowledgements

Funding from the *Deutsche Forschungsgemeinschaft* (grant TR555/6-1) is gratefully acknowledged. I thank Carola Trips, Achim Stein, Yela Schauwecker, and Richard Ingham for their cooperation in the BASICS project. For their work on the form-lemma correspondences, I thank Lena Kaltenbach, Natascha Schultheiß, Lisa Seidel, and Jonas Stork. I would also like to thank the three anonymous reviewers for their useful suggestions.

References

Cynthia L. Allen. 1995. *Case Marking and Reanalysis: Grammatical Relations from Old to Early Modern English*. Oxford University Press, Oxford.

Rolf Harald Baayen. 2008. *Analyzing Linguistic Data: A Pratical Introduction to Statistics Using R*. Cambridge University Press, Cambridge.

Clark C. Evans. 2009. YAML. `http://yaml.org`.

Anthony Kroch and Ann Taylor. 2000. The Penn-Helsinki Parsed Corpus of Middle English, Second Edition (PPCME2), Release 3. `http://www.ling.upenn.edu/hist-corpora/`.

Beth Levin. 1993. *English Verb Classes and Alternations: A Preliminary Investigation*. University of Chicago Press, Chicago.

Sebastian Löbner. 2002. *Understanding Semantics*. Routledge, London.

Frances McSparran, Paul Schaffner, John Latta, Alan Pagliere, Christina Powell, and Matt Stoeffler. 2001. Middle English Dictionary. `http://quod.lib.umich.edu/m/med/`.

Princeton University. 2010. WordNet. `http://wordnet.princeton.edu`.

Michael Proffitt, editor. 2015. Oxford English Dictionary. `http://www.oed.com`.

Python Software Foundation. 2015. Python 2.7.10. `https://www.python.org`.

R Core Team. 2016. R: A language and environment for statistical computing. `https://www.R-project.org/`.

Beth Randall. 2010. Corpussearch 2.003.00. `http://corpussearch.sourceforge.net`.

Leonard Richardson. 2015. Beautiful Soup Version 4.41. `http://www.crummy.com/software/BeautifulSoup/`.

Kirill Simonov. 2014. PyYAML. `http://pyyaml.org/wiki/PyYAML`.

Carola Trips and Achim Stein. accepted. Contact-induced changes in the argument structure of Middle English verbs on the model of Old French. *Journal of Language Contact*.

Hadley Wickham. 2009. *ggplot2: Elegant Graphics for Data Analysis*. Springer, New York.

Hadley Wickham. 2016. scales: Scale functions for visualization. `https://CRAN.R-project.org/package=scales`. R package version 0.4.0.

Running into Brick Walls Attempting to Improve a Simple Unsupervised Parser

Martin Riedl, Tim Feuerbach and **Chris Biemann**
Language Technology Group, CS Department, TU Darmstadt, Germany
riedl@cs.tu-darmstadt.de, uni@spell.work, biem@cs.tu-darmstadt.de

Abstract

In this article, we present a re-implementation of a simple unsupervised parser introduced by Søgaard (2012). This parser is able to parse sentences without any training. Furthermore, we propose various extensions to this parser. We evaluate the impact of several extensions on six languages. While we observe some improvements, different extensions impact different languages differently and we cannot give language-independent recommendations.

1 Introduction

Syntactic dependency parsing is a major preprocessing step needed for most applications and tasks in natural language processing like question answering (Hirschman and Gaizauskas, 2001), machine translation or similarity computations, e.g. (Levy and Goldberg, 2014; Weeds et al., 2004; Curran and Moens, 2002). However, most available dependency parsers are based on supervised machine learning algorithms, which need to be trained on manually created data. In addition, the creation of such training data is time-consuming and larger treebanks are not available for many languages.

In Riedl et al. (2014) several unsupervised dependency parsers have been extrinsically evaluated by using them as context representations for computing distributional similarities. In this work, the unsupervised parser by Søgaard (2012) yielded the second best results while being the fastest parser. In contrast to the other unsupervised dependency parsers, it does not require any training on raw text and is able to perform the parsing sentence-wise as opposed to whole-corpus parsing.

Whereas some unsupervised dependency parsers, e.g. Klein and Manning (2002), have been optimized and extended, e.g. Gillenwater et al. (2010), no further extensions have been proposed to many other unsupervised dependency parsers.

As the parser introduced by Søgaard (2012) is very basic in its heuristics, we will investigate whether integrating further features can improve its parsing performance. For this, we consider using semantics and Multiword Expressions (MWEs). Additionally, we re-run the parsing and train a supervised parser based on the output of the unsupervised parser.

2 Related Work

One of the first unsupervised syntactic dependency parsers that outperformed a random baseline was introduced by van Zaanen (2001) and uses an alignment-based learning approach. This algorithm is based on comparisons of sentences and uses sequence regularities in the corpus as constituents. A more sophisticated algorithm was presented by Klein and Manning (2002) that is based on an EM approach, which uses the linguistic phenomenon that long constituents often have shorter representations of the same grammatical function when they occur within a similar context. A combination of the work of Klein and Manning (2002) with a dependency model was presented by Klein and Manning (2004), which is called Dependency Model with Valence (DMV). This approach was the first one that outperformed the right branching baseline. Due to these results, this model has been extended by using lexical information (Headden III et al., 2009) and adding posterior regularizations in the training process (Gillenwater et al., 2010). These approaches require training, based on raw text or POS-tagged text. In contrast the method introduced by Søgaard (2012) does not require any training and can be applied with and without POS information.

Information about Multi-word Expressions (MWEs) has been shown to be beneficial for supervised dependency parsers. Le Roux et al. (2014) showed that for French, the detection of MWEs improves the parsing performance. Similarly, Eryiğit

et al. (2011) demonstrated that predicting Multiword Expressions (MWEs) and using such information for training a parser increases the performance.

3 Søgaard's Parser

In this paper we extend the unsupervised parser introduced in Søgaard (2012). It operates on single sentences and has three stages. First, tokens are ranked according to their valency. This is achieved by creating a multigraph with the sentence's tokens as its nodes. Edges are added following these heuristics:

- add pairwise edges to any neighbor in 1-step vicinity
- add pairwise edges to any neighbor in 2-step vicinity
- add an edge to a function word (determined by a word list) from any 1-step neighbor. The function word list is generated in advance using a simplification of TextRank (Mihalcea and Tarau, 2004) without stopword removal. The method is applied to the training data and we extract the top 50 words.
- add an edge to the verb from every other token in the sentence
- add pairwise edges between any tokens for which the 3-letter-prefix does not match
- add pairwise edges between any tokens for which the 3-letter-suffix does not match

Then, PageRank (Brin and Page, 1998) is applied in order to rank the nodes. The tokens are sorted in descending order to their rank and stored in a list called *dependents*. Additionally, a list called *head nodes* is created and a *ROOT* node is added. At the final stage, the dependency tree is created according to the following algorithm:

- while *dependents* is not empty
  1. remove first token
  2. assign a head from *head nodes*:
     - if universal dependency rules (Naseem et al., 2010) are used: assign the closest head (in terms of distance in the sentence) for which a rule fires
     - else, or if no rule applies: assign the closest head candidate
     - if ties: assign the head with the highest PageRank score
  3. add token to *head nodes*

4 Extensions

In this section, we describe all the extensions we will apply in order to achieve improvements for the parsing.

4.1 Re-running the Parsing

We expect that dependencies produced by the unsupervised parser might be helpful also for the parsing. Thus, we first apply Søgaard's parser to a new sentence. Then, we add the detected syntactic dependencies as weights to the normal heuristics, apply the ranking and build the dependency tree again.

4.2 Learning Regularities

One main advantage of Søgaard's parser is that it does not require any training since it is applies a collection of heuristics. However, previous decisions provide valuable information about the relationship of various POS. In order to utilize this information, we apply Søgaard's parser on raw text and use the dependency labels as training data for the MaltParser (Nivre, 2008). Using this model, we parse the test data and perform the evaluation on these dependencies.

4.3 Integrating Semantics

Words that have a similar meaning are usually on a similar level of salience. Therefore, we experimented removing edges between neighboring tokens that have a distributionally similar meaning. We use similarities computed with the approach by Biemann and Riedl (2013). As context representation we use the so-called trigram context extraction method, which uses the left and right neighboring word as context. In addition, we show results for German and English when using similarities computed using syntactic dependencies from a supervised method as context.

4.4 Integrating Multiword Expressions

Recognizing MWEs is beneficial for parsing, cf. Le Roux et al. (2014). Thus, we add edges between words that are recognized as MWEs according to a generated list of MWEs. This resource is generated using the unsupervised word sequence ranking measure called DRUID (Riedl and Biemann, 2015).

The measure does not require any POS filtering and can be applied to corpora without any linguistic pre-processing. We computed DRUID on a larger background corpus and used only word sequences of a maximum length of 4 and a score above 0.5. If a token was part of the same MWE as a head candidate, we preferred that candidate in the same vein as if it would match a universal rule.

5 Experimental Setting

We evaluate on German, Danish, Dutch, Portuguese, and Swedish test data from the 2006 CoNLL shared task on multi-lingual dependency parsing[1]. For English, we evaluated on Section 23 of the Wall Street Journal part of Penn Treebank III (PTB-III). As development set we use Section 11 of PTB-III. The treebank was converted to dependencies using the LTH Constituent-to-Dependency converter[2]. We train the MaltParser based on the parser's output on the train data of Danish, Dutch, German, Portuguese and Swedish. For English, we used the entire Wall Street Journal section of PTB-III. Unlabeled attachment scores were obtained using the official CoNLL-07 scorer.

For computing the similarities and the MWE resource for English we use 105M sentences of newspaper extracted from the Leipzig Corpora Collection (LCC) (Richter et al., 2006) and Gigaword (Parker et al., 2011). The computations for German are performed on 70M sentences from the LCC; for Swedish 60M sentences of newspaper data from Spraakbanken[3] are used. For Dutch, we compute similarities and MWEs based on 259 million sentences from the Dutch web corpus (Schäfer and Bildhauer, 2013).[4] The Portuguese is computed based on the Brazilian web corpus (Boos et al., 2014).

The dependency-based similarities are computed using the Stanford Parser (de Marneffe et al., 2006) for English and the MaltParser (Nivre, 2008) for German.

6 Results

In this section, we show the result of our re-implementation and additionally show the performances on different languages when incorporating the different modifications.

6.1 Performance on several languages

The results with our implementation[5] are presented for the six languages in Table 1, next to the results from Søgaard (2012).

	no UR		UR			
	We	Søgaard	We	Søgaard	Baseline	Oracle
Danish	55.70	50.8	54.38	51.4	43.77	71.49
Dutch	40.85	39.7	40.45	38.3	36.21	65.38
English	43.29	52.6	52.00	59.9	26.38	76.13
German	44.73	48.7	55.15	57.6	25.61	69.85
Portuguese	39.07	47.0	48.75	54.6	34.22	70.45
Swedish	47.68	52.3	56.86	60.5	30.60	71.87

Table 1: Basic unlabeled attachment scores on sentences with at most 10 tokens without punctuation. UR: Universal dependency rules enabled.

For unknown reasons, we cannot replicate results reported in (Søgaard, 2012)[6]. Whereas for Danish and Dutch, we observe higher scores than the ones in the paper, most results are below the performance of Søgaard (2012). This finding is consistent for both using universal dependency rules (URs) and without using URs. In accordance with the original implementation, our re-implementation outperforms the right-branching baseline. Like Søgaard (2012), we considered as upper bound an oracle function that ranks tokens in a top-to-bottom, left-to-right fashion according to their gold dependency trees.

6.2 Performance of Extensions

In this section, we describe the performance of the various extensions for adding edges into the graph-based method. First, we show results in Table 2 when re-running the algorithm, using dependency links from the first pass as additional edges. The number of additional edges (6) was determined using the English development data.

We observe that this extension reduces the performance both for Danish and Dutch tremendously. However, for English we observe significant improvements both for using/not using universal dependency rules. For German and Portuguese we only observe improvements when using universal

[1] http://ilk.uvt.nl/conll/post_task_data.html

[2] http://nlp.cs.lth.se/software/treebank-converter

[3] http://spraakbanken.gu.se

[4] available at: http://webcorpora.org/.

[5] The implementation is available under the Apache 2.0 license: http://jobimtext.org/jobimtext//components/unsupervised-parser

[6] Although we also tested the original implementation, we could not achieve the results from the paper. This might be attributed due to different keyword lists and different corpus transformations.

	no UR		UR	
	Basic	Re-running	Basic	Re-running
Danish	55.70	53.58	54.38	50.66
Dutch	40.85	36.21	40.45	35.81
English	43.29	43.62†	52.0	53.15†
German	44.73	44.36	55.15	58.33†
Portuguese	39.07	38.90	48.75	50.25
Swedish	47.68	47.29	56.86	56.17

Table 2: Results for re-running the algorithm on the same sentence. Scores with a † are significant over the basic score (paired bootstrap resampling test (Koehn, 2004) with $p = 0.05$, $n = 1000$).

dependency rules. Thus, no general trend can be obtained for re-using unsupervised dependency information.

Next, we show results in Table 3 when using the links obtained with Søgaard's dependency parser in order to train the supervised MaltParser as described in Section 4.2. Except for Danish, this

	no UR		UR	
	Basic	+MaltParser	Basic	+MaltParser
Danish	55.70	54.91	54.38	54.51
Dutch	40.85	41.25	40.45	43.77†
English	43.29	44.51†	52.0	50.19
German	44.73	45.47	55.15	54.53
Portuguese	39.07	39.40	48.75	46.08
Swedish	47.68	48.86	56.86	55.48

Table 3: Results for using the unsupervised dependency parses for training MaltParser and using MaltParser to parse the test data.

approach consistently yields improvements. This changes when universal rules are used; here, the performance on Dutch and Danish increases. For English we significantly outperform the basic results. However this comes at the cost of losing the runtime benefit of Søgaard's parser.

Next, we present the impact when integrating semantic information and MWE information into the unsupervised parser. As can be obtained from Table 4, using semantic information that is computed

	no UR			UR		
	Basic	MWEs	Semantics	Basic	MWE	Semantics
Dutch	40.85	40.98	40.72	40.45	40.58	40.05
English	43.29	43.33	43.03	52.0	51.96	52.15
German	44.73	44.98	44.61	55.15	54.90	55.64
Portuguese	39.07	39.23	39.40	48.75	48.41	49.42†
Swedish	47.68	47.78	47.09	56.86	56.47	56.37

Table 4: Results for using semantic information and preferring heads from the same MWE.

on neighboring words decreases the performance for all languages but Portuguese. Applying these rules, we observe declines for Dutch and Swedish, but gain improvements for the remaining languages. Additionally, we tested similarities for English and German that are computed using syntactic dependencies as context representation for testing purposes, as it defies the goal of inducing a parser for languages without treebank resources. Without using universal rules, we observe a decrease in terms of performance for English (43.25) and obtain slight increases for German (45.22).

Integrating information from the MWE resource and not applying the universal rules results in consistent yet small improvements among all tested languages (see Table 4). Similar to the results using semantic information, scores increase for all languages except for Dutch when using universal rules.

In the next experiment, we combined several extensions. As can be observed from Table 5 integrating semantic and MWE information improves the performance in all cases except for Swedish. In addition we also present results when adding the re-running to the algorithm. For Dutch and Swedish we notice a performance decline. When using universal rules, we observe an increase in performance for English, German, and Portuguese.

	no UR			UR		
	Basic	MWEs +Sem	MWEs +Sem +Re-running	Basic	MWEs +Sem	MWEs +Sem +Re-running
Dutch	**40.85**	**40.85**	35.94	40.45	40.45	35.15
English	43.29	43.37	43.37	52.0	**52.11**	**52.11**
German	44.73	45.34	44.73	55.15	55.51	**58.46**†
Portuguese	39.07	39.57†	39.57	48.75	49.08	**50.92**
Swedish	47.68	47.19	46.40	**56.86**	55.97	55.08

Table 5: Results for combining some of the extensions.

7 Conclusion

In this paper we have shown that intuitive and reasonable extensions for Søgaard's dependency parser do not translate into general improvements among all languages. This is in line with the findings described in (Riedl et al., 2014) that most unsupervised dependency parsers are optimized for English rather than the other languages. Whereas some extensions yield minor improvements, we cannot give any language-independent recommendation.

References

Chris Biemann and Martin Riedl. 2013. Text: Now in 2D! A Framework for Lexical Expansion with Contextual Similarity. *Journal of Language Modelling*, 1(1):55–95.

Rodrigo Boos, Kassius Prestes, Aline Villavicencio, and Muntsa Padró. 2014. In Jorge Baptista, Nuno Mamede, Sara Candeias, Ivandré Paraboni, Thiago A. S. Pardo, and Maria das Graças Volpe Nunes, editors, *Proceedings of the 11th International Conference on Computational Processing of the Portuguese Language*, PROPOR 2014, pages 201–206, São Carlos/SP, Brazil.

Sergey Brin and Lawrence Page. 1998. The Anatomy of a Large-Scale Hypertextual Web Search Engine. In *Seventh International World-Wide Web Conference*, WWW 1998, pages 107–117, Brisbane, Australia.

James R. Curran and Marc Moens. 2002. Improvements in automatic thesaurus extraction. In *Proceedings of the ACL-02 Workshop on Unsupervised Lexical Acquisition - Volume 9*, ULA 2002, pages 59–66, Philadelphia, PA, USA.

Marie-Catherine de Marneffe, Bill Maccartney, and Christopher D. Manning. 2006. Generating typed dependency parses from phrase structure parses. In *Proceedings of the International Conference on Language Resources and Evaluation*, LREC 2006, pages 449–454, Genova, Italy.

Gülşen Eryiğit, Tugay İlbay, and Ozan Arkan Can. 2011. Multiword expressions in statistical dependency parsing. In *Proceedings of the Second Workshop on Statistical Parsing of Morphologically Rich Languages*, SPMRL 2011, pages 45–55, Dublin, Ireland.

Jennifer Gillenwater, Kuzman Ganchev, João Graça, Fernando Pereira, and Ben Taskar. 2010. Sparsity in dependency grammar induction. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics - Short Papers*, ACL 2010, pages 194–199, Uppsala, Sweden.

William P. Headden III, Mark Johnson, and David McClosky. 2009. Improving unsupervised dependency parsing with richer contexts and smoothing. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, NAACL-HLT 2009, pages 101–109, Boulder, CO, USA.

Lynette Hirschman and Rob Gaizauskas. 2001. Natural Language Question Answering: The View from Here. *Journal of Natural Language Engineering (NLE)*, 7(4):275–300.

Dan Klein and Christopher D. Manning. 2002. A generative constituent-context model for improved grammar induction. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL 2002, pages 128–135, Philadelphia, PA, USA.

Dan Klein and Christopher D. Manning. 2004. Corpus-based induction of syntactic structure: Models of dependency and constituency. In *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, ACL 2004, pages 478–485, Barcelona, Spain.

Philipp Koehn. 2004. Statistical significance tests for machine translation evaluation. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, EMNLP 2004, pages 388–395, Barcelona, Spain.

Joseph Le Roux, Antoine Rozenknop, and Matthieu Constant. 2014. Syntactic parsing and compound recognition via dual decomposition: Application to french. In *Proceedings of the 25th International Conference on Computational Linguistics: Technical Papers*, COLING 2014, pages 1875–1885, Dublin, Ireland.

Omer Levy and Yoav Goldberg. 2014. Dependency-Based Word Embeddings. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, ACL 2014, pages 302–308, Baltimore, MD, USA.

Rada Mihalcea and Paul Tarau. 2004. Textrank: Bringing order into texts. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, EMNLP 2004, pages 404–411, Barcelona, Spain.

Tahira Naseem, Harr Chen, Regina Barzilay, and Mark Johnson. 2010. Using universal linguistic knowledge to guide grammar induction. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, EMNLP 2010, pages 1234–1244, Cambridge, MA, USA.

Joakim Nivre. 2008. Algorithms for deterministic incremental dependency parsing. *Computational Linguistic*, 34(4):513–553.

Robert Parker, David Graff, Junbo Kong, Ke Chen, and Kazuaki Maeda. 2011. *English Gigaword Fifth Edition*. Linguistic Data Consortium, Philadelphia, PA, USA.

Matthias Richter, Uwe Quasthoff, Erla Hallsteinsdóttir, and Chris Biemann. 2006. Exploiting the Leipzig Corpora Collection. In *Proceedings of the Fifth Slovenian and First International Language Technologies Conference*, IS-LTC 2006, pages 68–73, Ljubljana, Slovenia.

Martin Riedl and Chris Biemann. 2015. A Single Word is not Enough: Ranking Multiword Expressions Using Distributional Semantics. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP 2015, pages 2430–2440, Lisboa, Portugal.

Martin Riedl, Irina Alles, and Chris Biemann. 2014. Combining supervised and unsupervised parsing for distributional similarity. In *Proceedings of the 25th International Conference on Computational Linguistics: Technical Papers*, COLING 2014, pages 1435–1446, Dublin, Ireland.

Roland Schäfer and Felix Bildhauer. 2013. *Web Corpus Construction*. Synthesis Lectures on Human Language Technologies. Morgan and Claypool.

Anders Søgaard. 2012. Unsupervised dependency parsing without training. *Natural Language Engineering*, 18(02):187–203.

Menno van Zaanen. 2001. Building treebanks using a grammar induction system. Technical report, University of Leeds, UK, School of Computer Studies.

Julie Weeds, David Weir, and Diana McCarthy. 2004. Characterising measures of lexical distributional similarity. In *Proceedings of the 20th international conference on Computational Linguistics*, COLING 2004, pages 1015–1021, Geneva, Switzerland.

Isolation and Mapping of Place-Name Forms in Toponymic Data

Tobias Roth
Schweizerdeutsches Wörterbuch
Auf der Mauer 5
8001 Zürich
Switzerland
tobias.roth@idiotikon.ch

Abstract

We apply a customised approximate matching method to toponymic text data in order to isolate single place-name forms. Current place-names are matched to current and historical variants in standard and non-standard spelling. Such one-to-one mappings are preferred to text snippets with context, e. g. in the case of geo-referencing historical documents. The presented method yields an error rate of about 2%, which can be reduced manually and with reasonable effort to approximately 1%.

1 Introduction

An important task in the digitisation of historical documents is the tagging of place-names and the geo-referencing of these place-names found. This geo-referencing task can be completed much more efficiently if the tagging tool has access to a mapping from historical place-name forms to geographical coordinates. A promising data source for such mappings are toponymic projects (books of place-names) where one can often find both geographical coordinates and place-name forms in the historical-evidence sections.

Many of the toponymic projects also have their data in digital form. Yet, the records used as evidence are normally given as a line of plain text with minimal context, but without explicitly indicating the place-name form itself. For the usage of such data in geo-referencing the actual place-name form has to be isolated first. The present paper will explore methods to detect place-name forms in toponymic records. The problem does not look very difficult at first sight. Typically, there is a reference form (normally the current name) and a line of text that contains a form of the same place-name (in standard or non-standard spelling). An example of this historical-evidence part is shown below. It is an abbreviated entry for the name *Waldrüti* from Reber (2014):[1]

Waldrüti
Sources:
[...]
1534: ein Stuck matten vf der wald Rütj am menweg (Zins und Zehnten F1, 90r)
1548: wider an die walld Rüttÿ, biß vff ... kalberweÿde (Gösg Urb 1548, unpag.)
[...]
1704: die waldrüthj sampt der Ziegermatt (Ber 159, 198r)
1826: Jn der Wald-Rüti, Matten & Holzland (Haue Gb 1826, 463)
1872: Waldrüti (HaIf ÜbPlan 1872, Übersichtsplan)
[...]

For toponymic data in German speaking Switzerland which is considered first here, digitally readily available data amounts to roughly 450 000 toponyms with an estimated number of about 1.2 million source records.[2] With this amount of data in mind the focus lies on automated matching, not manual annotation. The present matching problem is situated somewhere between the normalisation problem of historical spellings and named entity recognition.

The paper is organised as follows: it starts with a short description of the data in question. We then look at different matching methods and their results and try to optimise the matching method to our case. We conclude with an error analysis and a general summary.

2 Forms of place-names present in the data

The data we would like to cover is toponymic data in German speaking Switzerland. There are or have been several regionally organised toponymic projects. This alone accounts for a certain heterogeneity in the data although the projects on the

[1] Cf. also https://search.ortsnamen.ch/record/106016746 (22.07.2016) for the complete entry.

[2] Cf. https://www.ortsnamen.ch (22.07.2016).

whole are quite compatible. Many inter-project diffences can also be found within projects. Such differences that are particularly relevant with respect to the present form-isolation task are presence or absence of a dialectal reference form, length of the context given in evidence records, additional information coming with the reference form, etc.

2.1 Present-day forms

Present-day i. e. 20th and 21st century forms in the records are easiest to match: They often coincide with the reference form and comply with standard spelling. They frequently come from maps, so they have little or no context with them. One difficulty can be dialectal forms as they show non-standard spelling. There even are phonetically transcribed strings (following different transcription systems).

2.2 Historical forms

Historical forms tend to differ considerably from the present-day reference form. Different patterns of deviation can be observed.

2.2.1 Non-standard orthography

Older forms show more variety in spelling as there was no standard orthography established yet. Some characters were used differently and there were also characters that are not in use anymore. Tokenisation differs sometimes: a compound word written as one word today is often written in two or more words in historical documents.

2.2.2 Discontinuous forms

In some cases forms are discontinuous, with two patterns that can be found frequently. One is the coordination pattern with ellipsis as in *X or Y street* with the target form being *X street*.

The other rather frequent case is the swapping of elements as in *X street* vs. *street to X*.

2.2.3 Name change

Names can change over time. It is not the loss or gradual change of phonetic material that is addressed here. If places are completely renamed this is, of course, a nearly unsurmountable barrier for a linguistic matching algorithm. But sometimes it is just parts of names that change: attributes are omitted, added or replaced, etc.

2.2.4 Substitutions with synonyms

A special variant of name change is the substitution of name constituents with synonyms. This is frequent with classifying constituents (e. g. routes can be called *Strasse, Weg* or *Gasse* interchangeably), but can also happen with attributes (e. g. *unter-* vs. *nieder-* for English *lower;* or the historical form *leupriesters garten* for modern *Pfarrgarten,* English *parish garden).*

2.2.5 Translations

Some of the older sources are written in Latin. Place-names mentioned in these documents are often also translated to Latin, at least the readily translatable elements.

Examples are attributes like *inferior* for English *lower* as in the record *in Ernlisbach Inferiori*[3] for modern *Niedererlinsbach*. Another frequently translated element is *bonum* for German *Gut* (English *estate, manor),* e. g. in the record *Bonum Scherерin*[4] for a now extinct toponym *Schärersguet.*

2.2.6 Uncertain naming status

If you look at a record it is not always clear which elements belong to the name and which ones are merely additional attributes that describe the place. There are records where all the elements you can find in the modern name are already present, but the sentence structure suggests that it is not a name yet. Attributive relative clauses are instances of this pattern: for the modern form *Trimbacherstrasse* there are historical records like *an der strasß die gon Trimppach godt*[5] (English *at the road that goes to Trimbach).*

2.3 Inflected forms

Both historical and modern forms can occur with inflectional endings. Inflectional forms are more frequent in older sources as present-day sources are very often maps or geographical information systems.

3 Matching of place-names

The isolation of actual place-name formes in place-name data is a rather specialised approximate-matching task. Classical named entity recognition (NER, cf., e. g., Sekine and Ranchhod (2009)) is not likely to perform well in this case. Although context is very restricted there can occur many more place-names and other named entities in such a text snippet, not only the wanted form.

Algorithms for the normalisation or canonicalization of historical text could be more helpful here.

[3]Record from 1406 (Reber, 2014).
[4]Record from 1423 (Reber, 2014).
[5]Record from 1623 (Reber, 2014).

For a general overview see, e. g., Piotrowski (2012, 69ff.). Different methods of approximate matching have been proposed. Hauser and Schulz (2007) and Bollmann et al. (2011) both use training data to automatically deduce weights for use in the computation of an edit-distance based similarity score; very similar Pilz et al. (2008), but with manual rule derivation in addition. Jurish (2008) converts the text with an adapted letter-to-sound system before comparison.

3.1 Methods and results

3.1.1 Development and test data

For development and test purposes, in a random sample of 6 000 records from Reber (2014) the actual place-names have been tagged manually. About 800 of these records were not used because they concerned family names or because they were phonetic transcriptions in IPA. Half of the remaining records were used as a development set, the other half as the final test set.

Another set of around 30 000 records with their corresponding isolated name forms from Dittli (2007) was used in development only.

3.1.2 Similarity based on edit distance

The following example can help to show what the task in question exactly consists of. It is a record for the name *Waldrüti*:[6]
wider an die walld Rüttÿ, biß vff . . . kalberweÿde
Given the standard form *Waldrüti* the desired result of the matching task for this record is the string *walld Rüttÿ*.

As a kind of baseline, matching was first performed using a similarity ratio based on simple edit distance (Levenshtein, 1966) computed with all strings transformed to lower case (column *ED* in table 1; see column *BL* in table 1 for baseline rates with random selection of words[7]). The ratio was computed as follows – with a cost of 1 for delete and insert, cost 2 for replace operations:

$$sim(x,y) = \frac{length(x) + length(y) - dist(x,y)}{length(x) + length(y)}$$

The error rate in the test set was 3.3% with this method. It got slightly better (3.1%) if all diacritics in the text were removed (column *ASC* in table 1); i. e. the text *wider an die walld rutty, biß vff . . . kalberweyde* was compared to the converted version of the reference word (*waldruti*).

Century	Count	Error rates in %			
		BL	ED	ASC	CST
<15th	68	80.9	13.2	13.2	10.3
15th	144	93.1	9.0	9.0	6.3
16th	524	94.5	6.1	5.7	3.6
17th	195	81.5	1.5	1.5	1.5
18th	226	77.4	4.9	4.4	1.8
19th	781	52.5	1.8	1.7	0.9
20th	312	11.5	1.0	1.0	0.6
21st	391	5.9	0.3	0.3	0.3
total	2641	56.3	3.3	3.1	2.0

BL = baseline; random selection of items
ED = edit distance, lower case
ASC = edit distance, lower case, diacritics removed
CST = edit distance after customised transformation

Table 1: Error rates with different matching methods (by century).

3.1.3 Weighted similarity

As we could use a set of 30 000 records with manually pre-annotated place-name forms (Dittli, 2007) we tried to improve the simple edit-distance based method above by taking these records as training data. We deduced replacement rules from this data set, comparable to methods described in Bollmann et al. (2011) and Hauser and Schulz (2007). The rules operated on one character with one character of context to the left and to the right. The cost of a given replacement depended on the ratio of its application in the training data, with a maximum cost assigned to unseen replacements. The similarity ratio was then computed as above, but with this weighted cost function.

The resulting error rate in the test set was at 3.4% and thus even lower than with the simple edit-distance based approach. A closer look at the training data suggests that there are too many errors in it,[8] so we decided not to further pursue the weighted-similarity branch for lack of adequate data.

3.1.4 Customised transformation and matching

Another possible approach are all the methods that try to simplify the strings before matching (after

[6] Record from 1548 (Reber, 2014).

[7] The low error rates for the 20th and 21st century even with random selection are again a sign of the many one-word records that come from maps or geographical databases.

[8] This data was not used in the printed version, so at some point it presumably was not maintained properly anymore.

the model of phonetic simplification as in methods like *Soundex* or *Kölner Phonetik* (Postel, 1969), see also Piotrowski (2012) and Jurish (2008)).

We adopted such a method with a very small set of manually selected replacement rules (see also Pilz et al. (2008) and Jurish (2008)). We replaced different writing variants of umlaut to e (e. g. *ö* to *oe*), merged *i, y, j* and *ie* to *i*, *th* to *t*, removed certain diacritics such as accents, etc. The rule set comprised less than 30 rules, for the simple reason that, at this point, rules we added (collected from our experiments in 3.1.3) did not further improve the error rate.

Sequences of identical characters were then reduced to one occurrence. Computation of the similarity ratio was done like in 3.1.2. Certain character alternations were not replaced before computation but were assigned reduced cost. An example is *v* that frequently alternates with *u* but also with *f* and *w*. It is easier to assign a reduced cost afterwards than to decide beforehands whether it is used as a vowel or as a consonant.

The inspection of the remaining errors in the development set led us to allow for discontinuous forms and discontinuous forms with swapped order (cf. 2.2.2). We also introduced penalties for forms that started or ended in certain words such as articles or prepositions, and we favoured forms that occurred just after an article or the like.

As a result we could lower the error rate in the test set to 2.0% (see column *CST* in table 1 for detailed results by century).

4 Error analysis and error management

There are some error types though that cannot be handled well with this procedure. Notably the types mentioned in 2.2.3–2.2.5 (name change, synonym substitution, translation) where the difference is not just a matter of spelling or sound change. A much more sophisticated apparatus than the one set up would be needed to account for these error types.

An error analysis with error rates by similarity ratio can show whether a threshold for the similarity ratio might be useful or how efficient manual post-processing might be. Table 2 presents these figures for our test set. The second and third columns give record counts and error rates for every similarity range. The two last columns show cumulated percentages of record counts as well as the proportion of all errors within these records. There are, for example, 75 records with a similarity ratio of 0.6–0.7, the error rate within these 75 records is at 21.3%; the records with a similarity ratio of up to 0.7 constitute 4.2% of all records, and they contain 50% of all errors.

Sim. ratio	**Count**	**Err. %**	**Cumulated**	
			% records	% err.
0.0–0.4	4	50.0	0.2	3.8
0.4–0.5	7	42.9	0.4	9.6
0.5–0.6	26	19.2	1.4	19.2
0.6–0.7	75	21.3	4.2	50.0
0.7–0.8	175	6.3	10.9	71.2
0.8–0.9	537	1.3	31.2	84.6
0.9–1.0	1817	0.4	100.0	100.0

Table 2: Error analysis by similarity ratio.

As 50% of all errors are in a well-defined set of around 4% of the records it could be considered to correct these errors manually. You could thus – with a reasonable effort – reduce the overall error rate to about 1%.[9]

Depending on the application one could also introduce a threshold for the similarity ratio of e. g. 0.7, but then you would lose the correctly classified forms of this range and these are likely to be the most interesting ones.

5 Conclusion and outlook

This paper has shown that place-name forms can reliably be detected in toponymic data using an automated matching method with a few manually set replacement rules and an edit-distance based similarity score (2% errors). The algorithm performs rather well in discerning doubtful cases: half of all the errors are in those 4% of the records with the lowest similarity ratio. Manual correction of these 4% can further reduce the error rate to about 1%.

For even further improvement such manual corrections could be taken as additional reference forms. Or, if set up as a web service for georeferencing historical documents, freshly annotated forms could and should be fed back into the original system.

References

Marcel Bollmann, Florian Petran, and Stefanie Dipper. 2011. Applying rule-based normalization to different types of historical texts – an evaluation. In *Pro-*

[9] For the situation in Switzerland depicted in the introduction, it would mean that about 50 000 records would have to be checked manually.

ceedings of the 5th Language & Technology Conference: Human Language Technologies as a Challenge for Computer Science and Linguistics (LTC 2011). Poznan, Poland.

Beat Dittli. 2007. *Zuger Ortsnamen. Lexikon der Siedlungs-, Flur- und Gewässernamen im Kanton Zug. Lokalisierung, Deutung, Geschichte. 5 Bände und Kartenset.* Balmer Verlag, Zug.

Andreas W. Hauser and Klaus Schulz. 2007. Unsupervised learning of edit distance weights for retrieving historical spelling variations. In Stoyan Mihov and Klaus U. Schulz, editors, *Finite State Techniques and Approximate Search, Proceedings of the First Workshop on Finite-State Techniques and Approximate Search*, pages 1–6, Borovets, Bulgaria.

Bryan Jurish. 2008. Finding canonical forms for historical German text. In Angelika Storrer, Alexander Geyken, Alexander Siebert, and Kay-Michael Würzner, editors, *Text Resources and Lexical Knowledge: Selected Papers from the 9th Conference on Natural Language Processing* (KONVENS 2008), pages 27–37. Mouton de Gruyter, Berlin.

Vladimir Levenshtein. 1966. Binary codes capable of correcting deletions, insertions and reversals. *Cybernetics and Control Theory*, 10:707–710.

Thomas Pilz, Andrea Ernst-Gerlach, Sebastian Kempken, Paul Rayson, and Dawn Archer. 2008. The identification of spelling variants in english and german historical texts: Manual or automatic? *Literary and Linguistic Computing*, 23(1):65–72. `http://llc.oxfordjournals.org/cgi/content/full/fqm044?ijkey=0RtdsFnq2rH7gwL&keytype=ref`.

Michael Piotrowski. 2012. *Natural Language Processing for Historical Texts*. Number 17 in Synthesis Lectures on Human Language Technologies. Morgan & Claypool Publishers LLC.

Hans Joachim Postel. 1969. Die Kölner Phonetik. Ein Verfahren zur Identifizierung von Personennamen auf der Grundlage der Gestaltanalyse. *IBM-Nachrichten*, 19. Jahrgang:925–931.

Jacqueline Reber, editor. 2014. *Die Flur- und Siedlungsnamen der Amtei Olten-Gösgen*, volume 3 of *Solothurner Namenbuch*. Schwabe, Basel.

Satoshi Sekine and Elisabete Ranchhod, editors. 2009. *Named Entities: Recognition, Classification, and Use*. John Benjamins Publishing, Amsterdam and Philadelphia.

Verifying the robustness of opinion inference

Josef Ruppenhofer
Institute for German Language
Mannheim, Germany
ruppenhofer@ids-mannheim.de

Jasper Brandes
Hildesheim University
Hildesheim, Germany
brandesj@uni-hildesheim.de

Abstract

There is increasing interest in recognizing opinion inferences in addition to expressions of explicit sentiment. While different formalisms for representing inferential mechanisms are being developed and lexical resources are being built alongside, we here address the need for deeper investigation of the robustness of various aspects of opinion inference, performing crowdsourcing experiments with constructed stimuli as well as a corpus study of attested data.

1 Introduction

In recent years, sentiment analysis has seen increasing interest in inferring implicit opinions in addition to capturing explicit expressions of opinion. Work by Reschke and Anand as well as Wiebe and her collaborators (Anand and Reschke, 2010; Reschke and Anand, 2011; Deng et al., 2013; Wiebe and Deng, 2014) has pointed up the great potential of opinion inference: speakers and authors leave many implicit opinions for hearers to infer. In (1), we can infer, for instance, that the speaker felt negatively about having the flu, if we assume that she values herself and has a negative attitude towards the flu. Further, we can infer that she has a negative attitude towards the flu shot that she deems causally responsible for getting the illness.

(1) The last time I got a flu shot, it GAVE me the flu.

However, corpus annotation studies and subsequent efforts to acquire lexical acquisition for opinion inference have left certain questions about the robustness of the inferences unaddressed. The one that we take up here is the limited range of potential inference types that have been empirically evaluated so far. Existing studies have focused on predicates related to (1) creation/destruction, (2) possession/lack and (3) affectedness and they have typically tested inference about event evaluation given knowledge about participant evaluation. However, as argued by Ruppenhofer and Brandes (2016), additional classes of predicates give systematic rise to opinion inferences, for instance, predicates related to similarity and location. In our crowd-sourcing experiments, we include these new classes of predicates. Further, we look at inferences in the 'opposite' direction, going from event evaluation toward participant evaluation. We also explore inferences in several kinds of less prototypical constellations. For instance, we look at concessive situations, in which a good or bad situation fails to be prevented. Similarly, we explore whether inferences only arise when an event producing a resultant state is explicitly mentioned, or also when pure states are presented, as implied by the work of Reschke and Anand (2011).

2 Related work

There exist several related but distinct approaches to sentiment inference. Two key ones are the work of Klenner and colleagues on verb polarity frames (Klenner et al., 2014; Klenner, 2015; Klenner and Amsler, 2016) and the work by Wiebe and colleagues on effect-based inference (Deng et al., 2013; Choi and Wiebe, 2014; Deng and Wiebe, 2014). The work of Klenner and colleagues is focused on effects on participants, whereas we are interested in the evaluation of an event by external viewers. The work of Wiebe and colleagues shares our perspective but due to its specific approach has a limited coverage compared to the approach that we adopt, functor-based inference.

2.1 Opinion inference based on functors

Reschke and Anand (2011) explored the relationship between the lexical semantics of predicates and the attitudes that speakers are inferred to have towards the events referred to by those predicates. They treat predicates and their arguments as func-

tors that map tuples of argument and verb properties to evaluations. An example is given in Table 1. The first row of the table applies to the situation where there is a possessor (x) who is valued positively by some nested source and a possession (y) that is also valued positively. If the relation between them is *have*, that relation is valued positively (left grey cell). If the relation is *lack*, that relation is valued negatively (right grey cell). The table shows that the reasoning for *lack* also applies to events of withholding and depriving which result in lack. Note that the possessor x of *withhold* and *deprive* is the grammatical object of these verbs in active-form sentences rather than the subject as in the case of *have* and *lack*. However, this difference is unimportant to the logic that applies.

x	y	have	lack	withhold	deprive
+	+	+	-	-	-
+	-	-	+	+	undef.
-	+	-	+	+	+
-	-	+	-	-	undef

Table 1: Functors for verbs embedding a state of possession

Two considerations are important to keep in mind. First, the goal of the inference procedure is to assess the attitude of an **external viewer** on the event. For instance, while in (2) the external viewer Sue may feel negatively towards a situation where a person she dislikes, *x*, got something desirable, *y*, the relevant possessor, Peter, will most likely feel positively about the award he got.

(2) Sue is **disappointed** Peter WON the award.

Second, the inference procedure must be **context-dependent** and be capable of producing different results, at least under some circumstances. In other words, there cannot be an inference that always goes through and yields the same polarity. That would not be a contextual inference but simply part of inherent lexical meaning.

Ruppenhofer and Brandes (2016) adopt the functor idea, proposing new functors for additional classes of verbs, among them predicates of location, similarity and sentiment.

Location This functor covers predicates entailing a state of location, e.g. *in/out of*; *at/away from* and *enter/exit*.

Figure	Ground	*in*	*out of*
+	+	+	-
+	-	-	+
-	+	-	+
-	-	+	-

Table 2: Functor for predicates expressing location

Sentiment This functor covers predicates expressing sentiment, e.g. *love/hate* and *fall {in/out of} love*.

Experiencer	Stimulus	*love*	*hate*
+	+	+	-
+	-	-	+
-	+	+	-
-	-	-	+

Table 3: Functor for predicates expressing sentiment

Similarity This functor covers predicates expressing similarity, e.g. *similar/different* and *assimilate/deviate*.

Item1	Item2	*similar*	*differ*
+	+	+	-
+	-	-	+
-	+	+	-
-	-	-	+

Table 4: Functor for predicates expressing similarity

We will use the Similarity functor in our crowdsourcing experiments. The functor underlies examples such as 3, which may be used to criticize the addressee for sharing traits with a parent.

(3) You're just `like` {your father/mother}!

2.2 Evaluation of functor-based inference

In the work of Reschke and Anand (2011), the usefulness of the predictions implicit in the proposed functors (existence, affectedness, possession) was tested using constructed sentences in which the participants in the argument slots of each predicate are canonically positive (e.g. *hero, cathedral*), negative (e.g. *villain, torture chamber*), or neutral (e.g. *man, building*). The authors presented annotators with a stimulus such as in (4) and asked them to assess

as *positive, negative* or *neutral* the author's overall evaluation of the event described in the sentence.

(4) The villain `murdered` the child.

Reschke and Anand (2011) report high inter-annotator agreement ($\kappa = 0.92$) for the predictions related to the affectedness and existence functors: "that is, killing was judged more positive when the entity losing existence was an enemy and judged more negative when it was an ally". For the possession functor, results seemed to be less clear-cut ($\kappa = 0.68$) for positively evaluated possessors possessing a positively evaluated possession (e.g. "a hero gaining a valuable watch") and negative possessors showing evaluations similar to neutrally judged possessors.

Ruppenhofer and Brandes (2016) report some results on crowd-sourcing experiments, in which they evaluate for several functors how consistently human ratings match the predictions that the functors make. In their experiments, the parameters that we vary in our experiments are stable: they always use eventive predicates, clearly biased sentence adverbs, canonical positioning of roles, and they always focus on the evaluation of the roles in the *entailed* relation. For instance, in the case of the possession functor, experiments test how well the functor predictions match human ratings for Possessor and Possession, but not for the Donor causing the entailed possession relation. Our experiments test the robustness of functor-based inference in a significantly broader range of constellations.

3 Experimental Design

While Reschke and Anand (2011) tested only the inference of event evaluation, we, like Ruppenhofer and Brandes (2016), run the functor-based inference process in the opposite direction: we fix the speaker's overall event evaluation but leave one or both of the participants of the functor predicates underspecified. Subjects are asked to guess the speaker's evaluation of one of the participants whose description is unbiased. Figure 1 shows a sample item illustrating this design. When guessing the speaker's evaluation of the participant in brackets, study participants could choose between five possible responses: 'positive', 'negative', 'neutral', 'mixed', and 'cannot tell'. In addition, users could leave a comment for each judgment.

Depending on the specific functor at issue, we expect to find either a preference for a specific kind of polarity towards the participant in question, or to see considerable variation of the polarity, if the event evaluation is compatible with both a positive or negative attitude towards the participant in question. For the latter cases, we are interested to find out whether raters choose among the possible polarities with more or less equal likelihood or whether we can find evidence of biases or default preferences. Consider the example given in Figure 1 involving the sentiment functor, shown in Table 3 above. We can derive the expected evaluations(s) of the target role (i.e. the phrase in brackets) that we would expect to see as the responses of our subjects for the stimulus. We have an eventive predicate and the sentiment functor is denied (*have fallen out of love*). The overall event evaluation by the speaker is explicitly positive as conveyed by the adverb (*fortunately*). The relevant argument to be judged by the study participants is in the *Arg1* position (*voters*). The sentiment functor in Table 3 shows that in theory both a positive and a negative evaluation of *Arg1* are compatible with the positive event evaluation (cf. second and fourth line).

The judgments we analyse are collected as part of larger surveys in which we also elicit intensity ratings for words and phrases. Each survey has about 40 utterances which are to be judged in terms of their evaluative stance. There is roughly the same number of intensity ratings in each survey. The two types of questions serve to mutually distract study participants from each other. Our items are randomized and presented singly to the participants. For each item, we collect judgments from 20 individuals.[1] We use a local installation of the LimeSurvey software and distribute the surveys to English native speakers registered in the US via the crowdsourcing website `prolificacademic.co.uk`. On average our surveys took between 10 to 21 minutes to complete. We paid each user between 2.40 Euro and 2.80 Euro, depending on the number of items in a survey. Each user could only participate in one survey in order to avoid learning effects or bias.

A full analysis of the factors impacting event evaluation would have to consider at least the parameters shown in Table 5, which we will briefly present. The `functor` parameter simply refers to which functor is relevant for the predicate at issue. `Functor polarity` refers to the question

[1] Due to a technical error, we sometimes received responses from one or two additional subjects.

Figure 1: Screenshot of a survey item

Element	Values
Functor	Existence, Possession, Sentiment, Similarity, ...
Functor polarity	Affirmed, Denied
Event evaluation	Positive, Negative, Un(der)specified
Sentence polarity	Affirmed, Denied
Relevant role	Arg1, Arg2, Causal, Concessive
Placement of adjunct role	in place, fronted
Biassing of other arguments	Yes, No

Table 5: Element inventory for constructing the survey items

whether the state of affairs referred to by the functor is affirmed or denied. E.g. for the adjective *similar* the affirmed version of the similarity functor is relevant, for the adjective *different* the denied version. The parameter `event evaluation` keeps track of whether the event is explicitly evaluated negatively or positively (cf. (5)), or whether it is under- (cf. 6) or unspecified (cf. (7)).

(5) Unfortunately/Fortunately, John `got` the job.

(6) Surprisingly, John `got` the job.

(7) John `got` the job.

`Sentence polarity` refers to whether the predicate is within the scope of syntactic negation. Accordingly, (8) is a case of the denied possession functor, while the affirmed functor applies to (6).

(8) Surprisingly, John didn't `get` the job.

The parameter `relevant role` tracks for which role we are interested in the author's asssessment. Examples (9)–(12) illustrate the roles we consider here.

(9) Unfortunately, [John] didn't `get` the job. (Arg1)

(10) Unfortunately, John didn't `get` [the job]. (Arg2)

(11) Unfortunately, John didn't `get` the job [because of his uncle]. (Causal)

(12) Unfortunately, John didn't `get` the job [despite his uncle]. (Concessive)

The `placement` parameter lets us distinguish for adjuncts whether they are placed in their default location (in place) or whether they are fronted. Thus, the causal argument is fronted in (13) but in place in (11).

(13) Unfortunately, [because of his uncle] John didn't `get` the job . (Fronted)

The use of the `placement` parameter is motivated by the fact that in combination with negation, the interpretation of certain adjuncts is potentially ambiguous when they are in place. For instance, (11) may mean one of two things:

(14) John got the job but this is not so for reasons to do with his uncle. The speaker evaluates John's getting the job for the wrong reasons negatively. (Cause > Neg)

(15) John didn't get the job and this is so for reasons to do with his uncle. The speaker

evaluates John's missing out on the job negatively. (Neg > Cause)

Finally the `biassing` parameter indicates whether any of the other arguments or adjuncts are specified for polarity. (16) has such a biased argument (*stupid John*), unlike its counterpart with a neutral Arg2 (10).

(16) Unfortunately, stupid John didn't `get` [the job]. (Arg2)

Although there are many parameters that we would ideally all control for, here we will only look at several small, focused contrasts as we were not able collect the full data set needed for a global analysis, given the funds available to us. We construct the utterances for the survey items by varying the stimuli along some (but not all) of the dimensions shown in Table 5.

4 Results

We present the crowdsourcing results separately for each of our component studies. We always contrast *positive* vs. *negative* event evaluation, *affirmed* vs. *denied* functor polarity, and two settings for a third parameter. Often the third parameter concerns the argument roles (e.g. *Arg1* vs. *Arg2*) of the functor to be judged. Thus, for each functor, we investigate 8 sentences that differ in the realizations of these features.

For reasons of comprehensibility, we present two result tables with counts per study, one for *positive* and one for *negative* event evaluation. This allows us to represent the data in a two-dimensional fashion, as shown in Table 6.

event evaluation		Parameter X Value 1	Parameter X Value 2
Functor	Aff	pos/neg/unbiased	pos/neg/unbiased
	Den	pos/neg/unbiased	pos/neg/unbiased

Table 6: General format of a crowdsourcing result table

The cells in this table contain the response frequencies for a *positive, negative*, and *unbiased* evaluation by the author of the utterance towards the specified role. Note that for the *unbiased* category, we conflate the three responses 'neutral', 'mixed', and 'cannot tell'.

4.1 Causal and concessive adjuncts

Previous work on opinion inference has focused on the derivation of an event's evaluation from evaluations of its participants. Ruppenhofer and Brandes (2016) focused on inference about participants, given the event evaluation. However, they focused on the participants in the entailed relation. For example, for predicates with a possession entailment, they looked at evaluations of the possessor and the possession. By contrast, inferences about the donor were not tested.

Here, we look specifically at roles that have to do with the causation of the entailed relation. The simple case are expressions that refer to a causal force bringing about the event and thereby its entailed relation. In cases like (17), one can simply project the event's (positive) evaluation on the causal force that is responsible for bringing about the event.

Concessive expressions are more complicated. These refer to situations or events that took place, and whose taking place would ordinarily lead one to expect that the situation in the main clause does not hold. Nevertheless, the situation expressed by the main clause does hold. In other words, concessive expressions (clauses or prepositional phrases) talk about cases of failed prevention. For instance, in (18), one understands (i) that it is true that the immigrants did not assimilate; (ii) that the group's efforts were aimed at preventing the non-assimilation; and (iii) those efforts failed. The evaluation of the (failed) counter-force that is expressed in the concessive clause, thus, ordinarily should be the opposite of that of the event that took place (i.e. the event that was not prevented). For example (18), one should thus expect a negative judgment about *the group's efforts*, given that they were aimed at preventing an event that the speaker approves of.

(17) Fortunately the immigrants have `assimilated` to the surrounding culture [because of the group's efforts]. (affirmed, causal)

(18) Luckily the immigrants haven't `assimilated` to the surrounding culture [despite the group's efforts]. (denied, concessive)

Tables 7 and 8 show the elicited results. We report both the raw counts and three measures of entropy, in bits, that reflect the consistency of the crowd in a single number. The entropy is zero when one of the outcomes is certain, that is, when all responses agree. We report the overall entropy for the three possible responses (3-way); the entropy

pos.	cau	con
Aff	18/0/2	10/9/1
Den	15/3/2	6/11/3

(a) Counts

	3-way		P v N		U v ¬U	
pos.	cau	con	cau	con	cau	con
Aff	0.47	1.23	0	1.00	0.47	0.28
Den	1.05	1.41	0.65	0.94	0.47	0.61

(b) Entropy

Table 7: Evaluation of causal and concessive roles for *similarity* functor given positive event evaluation

neg.	cau	con
Aff	1/15/6	5/13/3
Den	1/13/7	6/7/8

(a) Counts

	3-way		P v. N		U v. ¬U	
neg.	cau	con	cau	con	cau	con
Aff	1.09	1.32	0.34	0.85	0.85	0.59
Den	1.17	1.58	0.37	1.00	0.92	0.96

(b) Entropy

Table 8: Evaluation of causal and concessive roles for *similarity* functor given negative event evaluation

of the probability distribution for just the positive - negative opposition (P v. N); and the entropy for the distribution of unbiased vs biased (U v. ¬U). For the 3-way entropy, the range of values is (roughly) [0,1.59); for the other two entropy measures, it is [0,1.0].

The responses are much as expected for causal roles: they are mainly rated positively or negatively in line with the specified event evaluation. For concessive roles, the situation is less clear. For instance, in response to stimulus sentence (18) we would have expected to see overwhelmingly negative judgments of the concessive role (i.e. *the group's efforts*). Yet, as the lower right cell in Table 7 shows, we find quite a few (6) positive judgments relative to the expected negative ones (11).

4.2 Stative versus eventive predicates

The results shown in Tables 7 and 8 in Section 4.1 came about in response to stimuli which expressed a change of state. Now, we want to test the assumption that a change of state is not necessary for the functor reasoning to apply. Accordingly, in Tables 9 and 10 we present results that are derived from stimuli that are parallel in all respects to those for which results are reported in Tables 7 and 8, except that they are based on *stative* predicates. In other words, rather than use the predicate *assimilate* and its negation, we use the adjectives *similar* and *different*.

pos.	cau	con
Aff	14/1/5	7/6/7
Den	15/1/4	6/6/8

(a) Counts

	3-way		P v. N		U v. ¬U	
pos.	cau	con	cau	con	cau	con
Aff	1.08	1.58	0.35	0.99	0.81	0.93
Den	0.99	1.57	0.34	1.00	0.72	0.97

(b) Entropy

Table 9: Evaluation of causal and concessive roles for *similarity* functor given positive event evaluation

neg.	cau	con
Aff	1/14/6	1/10/10
Den	2/8/11	4/6/11

(a) Counts

	3-way		P v. N		U v. ¬U	
neg.	cau	con	cau	con	cau	con
Aff	1.12	1.23	0.35	0.44	0.86	1.00
Den	1.34	1.46	0.72	0.97	1.00	1.00

(b) Entropy

Table 10: Evaluation of causal and concessive roles for *similarity* functor given negative event evaluation

We see the same pattern of results for stative predicates that we saw for eventive predicates. But it appears that the eventive predicates yielded somewhat clearer judgments, at least for the causal roles. There are fewer unbiased responses with eventive predicates (Tables 7–8) than with stative predicates (Tables 9–10), which is reflected by lower entropy values for U v. ¬U in the former tables.

4.3 Canonical placement versus fronting

As pointed out above, placement might play a role in how adjuncts are interpreted. Here we specifically consider instances of causal adjuncts, as illustrated above in (11) and (13). The predicates we

use have a possession entailment and we use both affirmed and denied instances in combination with positive or negative event evaluation. The results are shown in Tables 11 and 12.

pos.	can	fro
Aff	17/0/4	19/1/1
Den	11/1/9	13/2/6

(a) Counts

	3-way		P v. N		U v. ¬U	
pos.	can	fro	can	frol	can	fro
Aff	0.70	0.55	0	0.29	0.70	0.28
Den	1.22	1.27	0.41	0.57	0.99	0.86

(b) Entropy

Table 11: Evaluation of causal role for possession functor given positive event evaluation

neg.	can	fro
Aff	1/15/4	3/15/3
Den	1/18/1	1/19/1

(a) Counts

	3-way		P v. N		U v. ¬U	
neg.	can	fro	can	fro	can	fro
Aff	0.99	1.15	0.34	0.65	0.72	0.59
Den	0.57	0.55	0.30	0.29	0.29	0.28

(b) Entropy

Table 12: Evaluation of causal role for possession functor given negative event evaluation

The results are rather heterogeneous, with no clear picture emerging. We do not consistently observe lower entropy values for fronted placement.

4.4 Unbiased event evaluation

The experiments above and those of Ruppenhofer and Brandes (2016) all used *biased* event evaluation via sentence adverbs such as *unfortunately*, *luckily*, etc. Here, we report on a simple control experiment in which we use a sentence adverb that bears no inherent polarity, namely *surprisingly*. We are looking at affirmed and denied instances of the similarity functor, for two different causation-related roles, namely adjuncts expressing a means or a concessive. Two example sentences are given in (19) and (20).

(19) Surprisingly, the immigrants have/haven't `assimilated` to the surrounding culture by [adopting the local customs]. (affirmed/denied, means)

(20) Surprisingly, the immigrants have/haven't `assimilated` to the surrounding culture [despite the party's efforts]. (affirmed/denied, concessive)

Given the unbiased nature of the sentence adverb, we predict responses to be neutral in the main, and to vary randomly between positive and negative among the non-neutral responses.

neu.	means	concessive
Aff	9/0/11	9/6/5
Den	4/3/13	5/4/11

(a) Counts

	3-way		P v. N		U v. ¬U	
neu.	mea	con	mea	con	mea	con
Aff	0.99	1.54	0	0.97	0.99	0.81
Den	1.28	1.44	0.99	0.99	0.93	0.99

(b) Entropy

Table 13: Evaluation of means and concessive roles for similarity functor given unbiased event evaluation

The first prediction that non-biased responses are in the majority is borne out for three of our constellations, as can be seen from Table 13. The exception are affirmed cases, where we ask about concessives. The preference for non-biased responses is, however, not very pronounced as shown by the high entropy values for U v. ¬U.

With regard to the second prediction, that the biased responses would be split rather evenly between positive and negative, this is borne out in most cases. The clear exception are affirmed cases in which we ask about the means role. Here, no negative evaluations of the means of assimilation were produced, resulting in an entropy of 0 for the P/N opposition. Potentially, the problem here lies with our stimulus: our raters might intrinsically all have favored the idea of immigrants assimilating and thus projected that positive attitude onto the means by which the assimilation is accomplished.

5 Corpus study

In sections 3 and 4, we investigated the robustness of opinion inference experimentally. In this section, we want to shed some light on attested instances

Count	*relieved*	*glad*
Existence	34	17
Location	24	27
Possession	16	21
Possibility	7	5
Sentiment	5	16
Affectedness	4	5
n/a	60	59
Total	150	150

Table 14: Functors embedded under *relieved*

of opinion inferences in corpora. To that end, we analyze clauses embedded under the predicates *relieved* and *glad*, which both provide positive event evaluation towards the situations expressed by embedded predicates. Example (21) shows an instance of the predicate *present*, which has an existence entailment that is negated in context, embedded under *glad*; example (22) illustrates a case where a predicate with a negative possession entailment, *conclude*, is embedded under the predicate *relieved*.

(21) I 'm **glad** that didn't present insurmountable problems as , although having suffered over the final volumes of the original " Dune " series I somehow was n't expecting too much , it turned out to be an extremely enjoyable story .

(22) Joan Keane , GMB Regional Organiser , said : " Whilst Mr Williams is **relieved** that the matter is now concluded , he has endured years of bullying and harassment by his colleagues ...

Table 14 shows the distribution of functor types embedded under 150 instances of each of the two predicates *glad* and *relieved*. The instances were randomly sampled from the uKwaC corpus (Ferraresi et al., 2008) and classified by the first author. The table shows that while the Existence and Possession functors proposed by Anand and Reschke (2010) are frequent, the Location functor is, too. Of the other new functors introduced by Ruppenhofer and Brandes (2016), only Sentiment is attested in the sample, but not, for instance, Similarity. [2]

The crowdsourcing experiments suggested that speakers employ certain defaults when reasoning about constellations of event and participants evaluations, where the latter are unspecified. Accordingly, it is interesting to ask if the default interpretations observed in the experiments match those that apply to naturally occurring instances where participant evaluation is unspecified.

We begin by considering the instances of the Possession functor in our samples for *glad* and *relieved*. For both predicates, of the four possible constellations that are compatible with positive event evaluation (cf. gray shaded cells in Table 15), only two occur, with a stark frequency difference among them: the constellation of positive evaluation for both participants and the event itself clearly predominates, which matches the results that Ruppenhofer and Brandes (2016) got when eliciting judgments for parallel, artificially constructed stimuli where the participants were described neutrally and only the event evaluation was explicitly biased.

		relieved		*glad*	
Possessor	P.ion	*have*	*lack*	*have*	*lack*
+	+	+/14	-	+/19	-
+	-	-	+/2	-	+/2
-	+	-	+/0	-	+/0
-	-	+/0	-	+/0	-

Table 15: Possession functor instances embedded under *relieved* and *glad*

We find similar asymmetries for the Location functor as we saw for the Possession functor. As Table 16 shows, the first constellation, where a positively valued Figure is at a positively valued Ground, predominates. However, we seem to find more variety than for Possession.

		relieved		*glad*	
Figure	Ground	*in*	*out of*	*in*	*out of*
+	+	+/13	-	+/22	-
+	-	-	+/8	-	+/3
-	+	-	+/2	-	+/2
-	-	+/1	-	+/0	-

Table 16: Location functor instances embedded under *relieved* and *glad*

For predicates with an Existence entailment, the distribution is as shown in Table 17. For the instances embedded under *glad*, we find a stark asymmetry, as we had before for the other functors. The constellation where existence of a positively valued

[2] The category "n/a" is assigned to instances where the main predicate of the embedded clause cannot be assigned to one of the known functors.

entity is valued positively is much more frequent. There is only instance of the other pattern, where non-existence of a negatively valued entity is evaluated positively by the external viewer, namely (21). By contrast, for *relieved*, the distribution of instances among the positively evaluated constellations is much more even and instances such as (22) are much more common.

	relieved		*glad*	
Entity	*exist*	*not exist*	*exist*	*not exist*
+	+/19	-	+/16	-
-	-	+/15	-	+/1

Table 17: Existence functor instances embedded under *relieved* and *glad*

The difference between *glad* and *relieved* is amenable to explanation. *Relieved* references a situation where an Experiencer feels positively about the fact that something (more) positive rather than something (more) negative happened. In talking about relief, one can highlight either the negative event that *did not* happen or the positive event that *did* happen, but the other viewpoint is always presupposed. Accordingly, we find many more references to instances of the negative functors (not being at a place, not existing) for *relieved* than for *glad*: the latter has no presupposition that a potential negative situation did not come to pass.

Overall, our preliminary corpus study supports the idea that not all constellations covered by a functor are equally frequent and that speakers and hearers may operate with default interpretations in elicitation tasks. The contrasts observed between *glad* and *relieved*, however, suggest that there may not be a global default that applies regardless of the specifics of the embedding predicate that specifies event evaluation. Stimulus construction for experimental tasks thus needs to pay attention to the rich lexical semantics of embedding predicates.

6 Conclusion and Future Work

In this work, we performed several crowdsourcing experiments in which we explicitly evaluated several key aspects of the functor-based framework for opinion inference. First, we established the relevance of the newly introduced similarity functor to opinion inference. Second, we tested opinion inferences that start with given event evaluations and target the evaluation of specific participants/roles. Here, we looked specifically at causal and concessive adjuncts, finding that the latter were less reliably evaluated in the way we had predicted than causal adjuncts. Other results gave evidence that opinion inference does indeed apply both to stative and their related eventive predicates alike, thus confirming Reschke and Anand (2011)'s intuition to that effect. We also performed a control experiment confirming that, given unbiased event evaluation, participant evaluation is either unbiased or varies more or less randomly between positive and negative polarity.

However, much remains to be done to firmly establish how reliable opinion inferences are, or what factors impinge on them. For instance, our experiments on the fronted or regular placement of causal and concessive adjuncts offered no real support for the idea that causal roles in fronted position lead to more consistently biased responses than in canonical (typically, final) position. Likewise, the results of all our experiments show, as does the work of Ruppenhofer and Brandes (2016), that the inferences produced for denied functors (e.g. *different*, *not assimilate*) tend to be less clear than those for affirmed functors (e.g. *similar*, *assimilate*). This is unexpected since, in terms of the logic of functors, the denied cases equally lead to predictable results. Both these last two findings may have resulted from our artificial setting, where context was lacking, even though both the use of fronted placement and negation are very much context-dependent. Testing on naturally occurring instances sampled from corpora might help resolve these and other questions.

To complement our crowdsourcing results, we performed a small corpus study to investigate the question where the default interpretations come from that were observed both in the study of Ruppenhofer and Brandes (2016) and in the present work. Our results suggest that the default values in the elicitation settings may derive from the usage patterns in naturally occurring, contextualized instances of opinion inference. However, the contrasts that we observed between our two event evaluation predicates, *glad* and *relieved*, suggest that there may be slightly different patterns of default reasoning used for different classes of embedding predicates that express event evaluation.

Acknowledgements

The authors were partially supported by the German Research Foundation (DFG) under grants RU 1873/2-1.

References

Pranav Anand and Kevin Reschke. 2010. Verb classes as evaluativity functor classes. *Proceedings of Verb 2010*, pages 98–103.

Yoonjung Choi and Janyce Wiebe. 2014. +/-EffectWordNet: Sense-level Lexicon Acquisition for Opinion Inference. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1181–1191, Doha, Qatar, October. Association for Computational Linguistics.

Lingjia Deng and Janyce Wiebe. 2014. An investigation for implicatures in chinese : Implicatures in chinese and in english are similar ! In *Proceedings of the 5th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 8–17, Baltimore, Maryland, June. Association for Computational Linguistics.

Lingjia Deng, Yoonjung Choi, and Janyce Wiebe. 2013. Benefactive/malefactive event and writer attitude annotation. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 120–125, Sofia, Bulgaria, August. Association for Computational Linguistics.

Adriano Ferraresi, Eros Zanchetta, Marco Baroni, and Silvia Bernardini. 2008. Introducing and evaluating ukwac, a very large web-derived corpus of english. In *Proceedings of the 4th Web as Corpus Workshop (WAC-4) Can we beat Google*, pages 47–54.

Manfred Klenner and Michael Amsler. 2016. Sentiframes: A resource for verb-centered german sentiment inference. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Thierry Declerck, Marko Grobelnik, Bente Maegaard, Joseph Mariani, Asuncion Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France, may. European Language Resources Association (ELRA).

Manfred Klenner, Michael Amsler, and Nora Hollenstein. 2014. Verb polarity frames: a new resource and its application in target-specific polarity classification. In *Proceedings of the 12th Edition of the Konvens Conference*, pages 106–115. Universität Hildesheim.

Manfred Klenner. 2015. Verb-centered sentiment inference with description logics. In *Proceedings of the 6th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 134–139, Lisboa, Portugal, September. Association for Computational Linguistics.

Kevin Reschke and Pranav Anand. 2011. Extracting contextual evaluativity. In *Proceedings of the Ninth International Conference on Computational Semantics*, pages 370–374. Association for Computational Linguistics.

Josef Ruppenhofer and Jasper Brandes. 2016. Effect functors for opinion inference. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Thierry Declerck, Marko Grobelnik, Bente Maegaard, Joseph Mariani, Asuncion Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Portorož, Slovenia, may. European Language Resources Association (ELRA).

Janyce Wiebe and Lingjia Deng. 2014. An account of opinion implicatures. *CoRR*, abs/1404.6491.

Data-Driven Identification of Dialogue Acts in Chat Messages

Dietmar Schabus, **Brigitte Krenn**, and **Friedrich Neubarth**
Austrian Research Institute for Artificial Intelligence (OFAI)
Vienna, Austria
firstname.lastname@ofai.at

Abstract

We present an approach to classify chat messages into dialogue acts, focusing on questions and directives ("to-dos"). Our multi-lingual system uses word lexica, a specialized tokenizer and rule-based shallow syntactic analysis to compute relevant features, and then trains statistical models (support vector machines, random forests, etc.) for dialogue act prediction. The classification scores we achieve are very satisfactory on question detection and promising on to-do detection, on English and German data collections.

1 Introduction

Online chat systems are a form of text-based communication that has been available since the early days of the Internet and that has become widespread in a variety of uses. In recent years, chat systems as a tool for business-internal communication seem to be an especially active market and such systems are sometimes replacing e-mail as the primary means of written communication within organizations. Uthus and Aha (2013) provide a survey over the active research field of automatic chat analysis. In processing chat messages, we face the challenge that traditional Natural Language Processing (NLP) techniques often do not work well, as with other forms of *microtext* (Ellen, 2011). This is due to typical characteristics such as message brevity, incorrect and/or non-standard spelling and grammar, fragment sentences, lack of or non-standard usage of punctuation, as well as influences from spoken and face-to-face communication for expressing sentiment or emphasis (e.g., emoticons, emojis, and character repetitions).

In this paper, we address the problem of classifying chat messages according to *dialogue acts* (Stolcke et al., 2000), i.e., the problem of assigning each chat message to one of a few categories that reflect the role of the message within the (multi-party) dialogue. For our purposes, the term dialogue act is roughly equivalent to the older term *speech act* (Searle, 1969). Typical dialogue acts of interest are various question types and directives, and in particular their realization in chat-based company communication.

Tagging chat messages with dialogue act labels can be useful as an intermediary step for other tasks such as thread disentanglement (Shen et al., 2006; Uthus and Aha, 2013), for facilitating information retrieval and extraction on chat data (e.g., search result filtering based on dialogue acts), and for enabling the chat platform itself to offer "smart" features based on dialogue act tags, for example.

We present an approach on dialogue act tagging of chat messages based on data-driven classification techniques, including random forests and support vector machines, which are trained on a collection of business chat logs. We show that our approach reaches high classification accuracy in an experimental evaluation. The system combines language specific and language independent components. In the current paper we present results for English and German.

The remainder of this paper is organized as follows. Section 2 presents related work, Section 3 describes the data corpus we work with and the dialogue acts we want to identify. In Section 4 we describe our method in detail, and evaluate its performance in Section 5. Finally, Section 6 draws conclusions and points out some ideas for future work.

2 Related Work

Wu et al. (2005) define 15 dialogue acts, including statement, yes-no-question and wh-question, to classify chat messages, using transformation-based learning with expert-provided rule templates. They report to achieve F_1 scores of 0.70 for yes-

no-questions and 0.53 for wh-questions. Using the same 15 categories on a different corpus, Forsyth and Martell (2007) compare a neural network to to a naive Bayes classifier. As features they use several message distances and occurrence counts of specific keywords, as well as presence of certain words as the first token of the message. For the neural network they report F_1 scores of 0.75 for yes-no-questions and 0.74 for wh-questions.

To detect questions in discussion threads on Yahoo! Answers, Wang and Chua (2010) use sequential pattern mining and syntactic shallow pattern mining (parse trees, to which a simplification procedure is applied) as features for a one-class support vector machine. They report an F_1 score of 0.91.

Carpenter and Fujioka (2011) define 43 dialog act categories and use long string matching and several rules (starts with, ends with, contains) to classify IRC chat messages. They report 90% accuracy, but state that this is partly due to the constrained context of the messages in their corpus.

Both Dent and Paul (2011) and Li et al. (2011) attempt to detect questions in Twitter messages, using different rule sets. They achieve F_1 scores of 0.71 and 0.92, respectively. The latter paper also evaluates an approach to detect interrogatives based on support vector machines, however it did not result in an improvement of detection accuracy. Zhang et al. (2011) also work with Twitter messages, but categorize them using five dialogue act categories, including statements and questions. By training a support vector machine on unigram, bigram and trigram features, they achieve an F_1 score of 0.64 both for the question category and as an overall average.

Kim et al. (2010) detect 12 dialogue acts (including open questions, yes-no-questions and requests) in one-on-one chats using conditional random fields on bag-of-words features and additionally exploiting structural and inter-utterance dependencies. They have later expanded their work to 14 dialogue acts on multi-party chats (Kim et al., 2012), where they report F_1 scores of 0.42, 0.75 and 0.87 for requests, wh-questions and yes-no-question, respectively.

O'Shea et al. (2013) attempt to distinguish questions from non-questions, using decision trees trained on 22 part-of-speech-like categories of function words as features, with sentences represented as category vectors. They report classification accuracies of 99% on their "straightforward question vs. non-question without preamble" data set and 79% on their "simulated clauses" data set.

Category	Ab.	Examples
Wh-quest.	wh	@ron so whats the state of dev now
Y-N-quest.	yn	or can I just specify one of them
Echo quest.	ec	a username can contain slashes?
Non-quest.	nq	just read your post :P
Directive	td	nice, please send them to me
Non-dir.	nd	lol no problem. ^^

Table 1: Sample utterances for the categories considered in question and directive classification.

It should be noted that all mentioned contributions deal with English data only, whereas we work on both English and German data and have already generalized many aspects of our system to work with multiple languages. Furthermore, all above papers employ either a rule-based approach or a machine learning approach on very simple features. We extract relevant syntactic features using a small rule set and then employ machine learning techniques. As we show in Section 5, our syntactic features are crucial for the classification results we achieve. In addition to the detection of questions, which many others have also investigated, we also detect directives, which are much less commonly considered.

3 Data Sets

Focusing on the detection of two groups of dialogue acts, *questions* and *directives*, we have assembled collections of sample sentences for classifier training and testing both in English and German. In question detection we attempt to classify any given message as either a *wh-question* (based on an interrogative word), a *yes-no-question* or a *non-question* (e.g. a declarative statement). Additionally, we use the label *echo question* for questions that do not exhibit clear interrogative grammatical structure, such as declarative statements ending in a question mark and fragments with a question mark.

In directive detection we intend to distinguish directives ("to-dos") from non-directive messages. Directives are often phrased as imperatives, but note that it is also possible that a given utterance is both a directive and a question, e.g., "Can you write the report, please?". Table 1 gives examples from our data for all categories under investigation.

Each of the four subcorpora comprises 1500 hand-labeled utterances, which were taken from

Class:	nq	yn	ec	wh	total	td	nd	total
English	618	379	261	242	1500	501	999	1500
German	819	233	204	244	1500	679	821	1500

Table 2: Class frequencies of the question corpora for English and German (left) and of the directive corpora for English and German (right).

various sources, including real English business chat messages, provided to us by our project partner,[1] real German chat messages from the "Dortmunder Chat-Korpus" (Beißwenger, 2013) and sentences/utterances taken from out-of-copyright novels in English and German.[2] By using this mixture we attempt to cover both typical chat message style as well as more grammatically rigid and more elaborate language from novels. Many chat messages in our collection are very short, and even in the longer ones complex sentence structures are very rare. For each question class, we have removed the question marks from 50% of the utterances that originally had one, such that the classifier cannot rely on the presence or absence of question marks alone.

The class frequencies of the four subcorpora are given in Table 2. Note that we have separate disjoint data collections for the question detection task and the directive detection task, i.e., we have 6000 labeled utterances altogether. Ideally, the numbers for the two languages would be more symmetrical, but as we do not focus on a comparison between them, we consider this no serious problem. The agreement between two labelers was higher than 98% for both question subcorpora and higher than 84% for both to-do subcorpora. This difference is due to the fact that the definition of to-dos is by far not as clear-cut as that of questions. Sometimes it can only be decided on a semantic or pragmatic level, assuming a certain context, whether or not a given chat message should be labeled as a to-do or not. We therefore also expect our automatic classifiers to perform better on questions than on to-dos.

4 Method

We have developed a software pipeline for dialogue act detection in chat messages with support for multiple languages. Most parts of the pipeline are language independent, the few language specific ones are currently available for English and German. Both in the training phase and later during detection, messages are first split into utterances and tokens using a custom tokenizer we have developed for chat messages. The tokenizer uses English and German lexica with more than 400,000 and 2,000,000 full form entries, respectively. Looking up a given lexeme in the lexicon yields all possible readings wih the repective part-of-speech (POS) tags and morpho-syntactic features (e.g., "bears" yields a plural noun reading and a third person singular verb reading).

Given that standard NLP tools such as POS-taggers and parsers would not work well on the short and fragmented utterances typically found in chat, especially without adequate training data, we refrain from applying such techniques. Instead we operate on ambiguous morpho-syntactic information as retrieved from the lexicon on which we perform a shallow rule-based analysis: Starting from the beginning of the message, we skip all tokens that are greetings, interjections, conjunctions, adpositions or non-words like URLs, emoticons etc. The first token that is not to be skipped is labeled p_1 (intuitively, the first syntactically relevant word in the message). Starting from p_1, and depending on the (possible) morpho-syntactic features of the token at p_1, a small rule set continues to skip tokens that may belong to the syntactic phrase headed by p_1. After that, the next token is labeled p_2, for example:

haha well ok but $\underbrace{\textit{which}}_{p_1}$ *of these things* $\underbrace{\textit{are}}_{p_2}$ *true?*

When this heuristc procedure works well, p_1 will point to the subject of the clause and p_2 to the finite verb in a declarative statement, and vice-versa in a yes-no-question. In a wh-question, p_1 will point to the interrogative pronoun and p_2 to the finite verb, etc. Position p_2 or even both p_1 and p_2 may be undefined, for example when the message consists only of a single interjection. With this simple procedure for shallow syntactic analysis, we are able to capture the most relevant structural properties for detecting questions and directives, even in short and incomplete sentences.

Given the (ambiguous) POS tags and other morpho-syntactic features for each token, as well as the two positions p_1 and p_2, we define a high-dimensional binary feature vector, which contains, amongst others: The POS tags, lemmata and morpho-syntactic features at p_1 and p_2, all of

[1] http://grape.io
[2] https://www.gutenberg.org

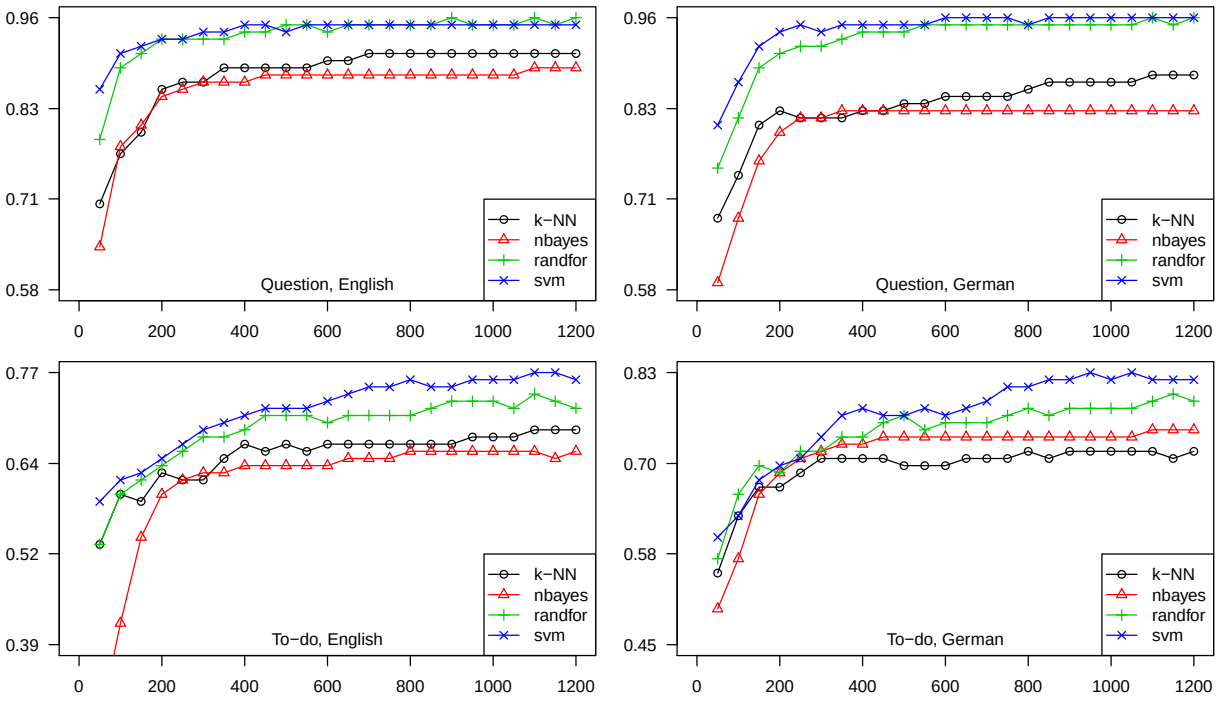

Figure 1: Evaluation results for the four data sets. The horizontal axis shows to the number of training samples. The vertical axis shows the average F_1 score across five cross validation folds. All four plots have identical vertical scaling.

these features appearing anywhere in the utterance, and the presence of some indicative phrases (e.g., "please", "can you", "you should"). It should be noted that POS, lemma and morpho-syntactic features are ambiguous for many tokens, as described above. All features are encoded as binary variables in the feature vector indicating the presence or absence of a certain feature (such as "noun at p_1" and "plural at p_2").

The features and data described above are used to train a classification model such as a support vector machine or random forest. The following section investigates the performance of various methods. After training, new input messages can be classified by first detecting the language of the input, applying utterance splitting, tokenization, rule-based syntactic analysis and feature extraction as described, and finally by using the model to predict the dialogue act.

5 Evaluation

To evaluate the method described in Section 4 on the data described in Section 3, we have carried out a series of experiments. For each of the four data sets of 1500 utterances, a five-fold cross validation setup was employed, yielding 1200 training utterances and 300 test utterances per fold. Within each fold, we initially used only 50 utterances to train a model and gradually increased this number to the full 1200, always evaluating the model on the same 300 test utterances. The whole procedure was repeated using the following four modeling approaches: k-nearest neighbors (k-NN; $k = 5$), naive Bayes (nbayes), random forest (randfor; 100 trees) and support vector machine (svm; linear kernel) from the scikit-learn library (Pedregosa et al., 2011). The results are shown in Figure 1, where each data point is an average F_1 score across the five folds, and in the case of the multi-class problem of question detection also across the four classes (macro averaging).

We observe that overall better results are achieved in question detection than in to-do detection, as expected. Interestingly, the results of the two best methods (svm and randfor) begin to level off already around 500 training utterances for question detection, but they continue to rise for to-do detection, suggesting that in the latter case additional training data could further improve the results.

The results for the best method (svm) using all the 1200 utterances for each fold are shown in Table 3, which lists precision, recall and F_1 score for each of the classes. For question detection, all val-

	English			German		
Cl.	Prec.	Rec.	F_1	Prec.	Rec.	F_1
nq	0.961	0.953	0.957	0.977	0.978	0.977
ec	0.969	0.957	0.963	0.977	0.960	0.968
wh	0.911	0.933	0.922	0.975	0.983	0.979
yn	0.966	0.972	0.969	0.919	0.916	0.917
td	0.783	0.750	0.765	0.804	0.834	0.818

Table 3: Precision, Recall and F_1 score values per category resulting from 5-fold cross validation using a linear kernel support vector machine.

	English			German		
Cl.	Prec.	Rec.	F_1	Prec.	Rec.	F_1
nq	0.855	0.878	0.865	0.942	0.950	0.946
ec	0.827	0.716	0.765	0.805	0.794	0.796
wh	0.764	0.837	0.796	0.838	0.888	0.862
yn	0.709	0.695	0.701	0.709	0.647	0.673
td	0.719	0.599	0.653	0.711	0.699	0.702

Table 4: Precision, Recall and F_1 score values per category resulting from 5-fold cross validation using a linear kernel support vector machine, when the features based on p_1 and p_2 are not used.

ues are above 0.95 except for English wh-questions and for German yes-no-questions, where they are still above 0.91. To-do detection is less reliable, here all values are greater than 0.75.

Typical errors made by the system include: free relative clauses that are mistaken for a wh-question ("What strikes me is that ..."), statements with dropped subject pronoun that are mistaken for a to-do ("love it!", "just read your post"), yes-no-questions with dropped auxilary verb that are not recognized correctly ("you on your way?"), and to-dos that our system misses because they are expressed indirectly ("john, the build system needs an update") or phrased in a way that is too complex for our simple approach ("I would like to note that you still need to finish the presentation").

Interestingly, if we remove the features based on the p_1 and p_2 positions, we observe a substantial drop of the classification results, as shown in Table 4. For example, recall drops from 0.972 to 0.695 for English yes-no-questions and from 0.750 to 0.599 for English to-dos; similar for German. This large difference indicates that our shallow syntactic analysis is crucial for the good classification results we achieve.

6 Conclusion

We have presented a data-driven approach for classifying chat messages into dialogue acts, with a focus on (several types of) questions and directives, in English and German. We use (ambiguous) POS and other morpho-syntactic information in combination with a rule-based shallow syntactic analysis as features for several learning algorithms, with support vector machines achieving the best results in our experiments. Our F_1 scores for question detection seem better than those in related work, although a fair comparison would require a standardized evaluation corpus. For a problem that has not received a lot of attention, our scores in to-do detection are also promising, with some room for improvement. The shallow syntactic analysis plays a key role in our system; in future work we plan to make this component also data-driven rather than rule-based. Furthermore, we would like to additionally consider the conversational context of each message for improving the detection of to-dos.

Acknowledgments

This research was funded in part by the Austrian Research Promotion Agency (FFG) in the "FFG Basisprogramm" under project number 850380. The authors would like to thank UberGrape GmbH (http://grape.io) for the fruitful collaboration and for providing interesting real-world data for us to work with.

References

Michael Beißwenger. 2013. Das Dortmunder Chat-Korpus. *Zeitschrift für germanistische Linguistik*, 41(1):161–164, April.

Tamitha Carpenter and Emi Fujioka. 2011. The role and identification of dialog acts in online chat. In *Proceedings of the Workshop on Analyzing Microtext at the 25th AAAI Conference on Artificial Intelligence*, pages 2–7, San Francisco, CA, USA, August.

Kyle Dent and Sharoda Paul. 2011. Through the Twitter glass: Detecting questions in micro-text. In *Proceedings of the Workshop on Analyzing Microtext at the 25th AAAI Conference on Artificial Intelligence*, pages 8–13, San Francisco, CA, USA, August.

Jeffrey Ellen. 2011. All about microtext – a working definition and a survey of current microtext research within artificial intelligence and natural language processing. In *Proceedings of the 3rd International Conference on Agents and Artificial Intelligence (ICAART)*, pages 329–336, Rome, Italy, January.

Eric N. Forsyth and Craig H. Martell. 2007. Lexical and discourse analysis of online chat dialog. In *International Conference on Semantic Computing (ICSC)*, pages 19–26, Irvine, CA, USA, September.

Su Nam Kim, Lawrence Cavedon, and Timothy Baldwin. 2010. Classifying dialogue acts in one-on-one live chats. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 862–871, Cambridge, MA, USA, October.

Su Nam Kim, Lawrence Cavedon, and Timothy Baldwin. 2012. Classifying dialogue acts in multi-party live chats. In *Proceedings of the 26th Pacific Asia Conference on Language, Information, and Computation (PACLIC)*, pages 463–472, Bali, Indonesia, November.

Baichuan Li, Xiance Si, Michael R. Lyu, Irwin King, and Edward Y. Chang. 2011. Question identification on Twitter. In *Proceedings of the 20th ACM International Conference on Information and Knowledge Management (CIKM)*, pages 2477–2480, Glasgow, UK, October.

James D. O'Shea, Zuhair A. Bandar, and Keeley A. Crockett. 2013. Optimizing features for dialogue act classification. In *IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 474–479, Manchester, UK, October.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

John R. Searle. 1969. *Speech acts: An essay in the philosophy of language*, volume 626. Cambridge university press.

Dou Shen, Qiang Yang, Jian-Tao Sun, and Zheng Chen. 2006. Thread detection in dynamic text message streams. In *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 35–42, Seattle, WA, USA, August.

Andreas Stolcke, Noah Coccaro, Rebecca Bates, Paul Taylor, Carol Van Ess-Dykema, Klaus Ries, Elizabeth Shriberg, Daniel Jurafsky, Rachel Martin, and Marie Meteer. 2000. Dialogue act modeling for automatic tagging and recognition of conversational speech. *Computational linguistics*, 26(3):339–373.

David C. Uthus and David W. Aha. 2013. Multiparticipant chat analysis: A survey. *Artificial Intelligence*, 199–200:106–121.

Kai Wang and Tat-Seng Chua. 2010. Exploiting salient patterns for question detection and question retrieval in community-based question answering. In *Proceedings of the 23rd International Conference on Computational Linguistics (COLING)*, pages 1155–1163, Beijing, China, August.

Tianhao Wu, Faisal Khan, Todd Fisher, Lori Shuler, and William Pottenger, 2005. *Foundations of Data Mining and knowledge Discovery*, volume 6, chapter Posting Act Tagging Using Transformation-Based Learning, pages 319–331. Springer, Berlin/Heidelberg, August.

Renxian Zhang, Dehong Gao, and Wenjie Li. 2011. What are Tweeters doing: Recognizing speech acts in Twitter. In *Proceedings of the Workshop on Analyzing Microtext at the 25th AAAI Conference on Artificial Intelligence*, pages 86–91, San Francisco, CA, USA, August.

Mapping PDTB-style connective annotation to RST-style discourse annotation

Tatjana Scheffler **Manfred Stede**
UFS Cognitive Sciences
University of Potsdam, Germany
{tscheff|stede}@uni-potsdam.de

Abstract

Penn Discourse Treebank and Rhetorical Structure Theory annotation account for different aspects of discourse structure, but to some extent, their analyses also correspond to each other. For a corpus annotated with both types of information, we describe a procedure for mapping systematically from the first layer to the second. In this way, we can observe commonalities and differences in the annotations of discourse structure between the two approaches. The method also allows for a data-driven mapping of coherence relations from one taxonomy to another with a suitable independently-annotated corpus.

1 Introduction

Among the various approaches to discourse structure, Rhetorical Structure Theory (RST, (Mann and Thompson, 1988)) and the Penn Discourse Treebank (PDTB, (Prasad et al., 2008)) have inspired a range of annotation projects, so that a number of corpora are available for both, and can be compared to each other. For English, there is some overlap between the RST-DT (Carlson et al., 2003) and the PDTB texts, but to our knowledge the correspondences between the two layers have not been explored yet. In this paper, we describe our implementation of the mapping in the German *Potsdam Commentary Corpus* (Stede and Neumann, 2014), for which RST and PDTB-style connectives have previously been annotated independently.

Both RST and PDTB attempt to model discourse structure, particularly the coherence relations between abstract entities (propositions, etc.) in the text. However, there are well-known differences between the approaches, such as a global (RST) vs. local (PDTB) view on discourse structure, the grounding of coherence relations in the cognitive effect on the reader (RST) vs. the semantic relation between the relation's arguments (PDTB), etc. Still, there is considerable overlap in the inventory of coherence relations between the formalisms, and insights from one type of annotation can confirm or extend insights from the other. For this purpose, we have developed a procedure that maps corresponding parts of PDTB-style and RST annotations to each other. Besides the practical benefit of checking annotation consistency and quality, we see the mapping as potentially fruitful for further theory development:

- *Structural* decisions may differ: Annotators looking for individual relation–argument configurations in PDTB-style analysis, disregarding any notion of overall text structure, may assign different text spans to a relation than RST annotators do when they are forced to produce a well-formed overall tree. Are such disagreements merely accidental, or do they point to interesting cases of ambiguity? Do they yield evidence that the tree constraint of RST may be too strong?

- *Relation types* or *connective senses* in the two approaches overlap but are not identical. When relations are mapped, they can provide information on the granularity, ambiguity, or vagueness in the inventories of categories and their usage.

In the present paper, we do not address the relation types but focus on describing a procedure for the mapping of structures only.

In the following, Section 2 gives a brief description of the two approaches, and then Section 3 states the assumptions we make for the mapping procedure, which is outlined in Section 4. Then, Section 5 discusses our findings on the relationship between the two accounts of discourse structure in the corpus. Finally, Section 6 addresses related work, and Section 7 gives a summary.

2 Discourse annotation

2.1 Connectives: Penn Discourse TreeBank

In PDTB-style annotation, the primary goal is to identify connectives and to link them to their two arguments: 'Arg2' is the one that is syntactically integrated with the connective, and 'Arg1' is the "external" one. Usually, Arg1 and Arg2 are adjacent (or embedded), but occasionally, Arg1 can be non-adjacent. In addition to proper connectives, annotators are encouraged to also find "alternative lexicalizations" (such as productive phrasal expressions) that serve a connecting function. Furthermore, the PDTB also links adjacent sentences into a relation even when no connective lexicalization is present; these cases are called "implicit connectives". Any instance of a relation (signalled or not) receives a sense label, which is taken from a hierarchy of 43 senses.

A key point is that annotation decisions are made for each relation individually. Connective/argument triples are not being related to one another, so no global text structure is built. This is a deliberate decision of PDTB, which aims at taking "one step beyond sentence syntax" but not the leap toward a discourse representation whose construction would be more difficult to annotate and involve more subjective interpretation.

2.2 Rhetorical trees: RST

In RST, coherence relations are being assigned to adjacent "minimal discourse segments", and recursively to larger spans. The original proposal of (Mann and Thompson, 1988) suggested some 25 relations, but different inventories have been used (notably the one for the aforementioned RST-DT, comprising 78 relations). Connectives can make this decision easier, but they are not the subject of annotation. For most relations, one segment is marked as central for the author's purposes ('nucleus') and the other as merely supportive ('satellite'). A few relations are multinuclear: these may contain two or more nuclei. Importantly, the relation assignment is recursively applied to larger spans as well, so that a tree structure results eventually, which spans the complete text and thereby serves as a model of its coherence. Crossing edges are not allowed according to Mann and Thompson, nor can there be any "gaps" in the analysis: The text is a contiguous sequence of minimal units.

Since many relation definitions involve speaker intentions, an RST analysis amounts to reconstructing the author's "plan", and for non-trivial texts this requires quite a bit of subjective interpretation.

In sum, the PDTB and RST analyses start out from quite different, and to a good extent complementary, goals. At the same time, they obviously have some overlap: Often a connective and its arguments will directly correspond to an RST relation and its segments. In research on RST, the role of signalling devices such as connectives has been discussed prominently (Taboada and Das, 2013). As stated earlier, one goal of our work is to be able to systematically study and quantify this overlap.

3 Constraints on the mapping

In our multi-layer annotation scenario, for the connective-argument layer we use a variant of the PDTB approach. We restrict our discussion in this paper to only explicit connectives (in the sense of (Fraser, 1999) or (Pasch et al., 2003)), excluding free phrasal expressions and non-signalled relations (although the method could be extended to these cases in future work). A connective can consist of multiple tokens, which can be continuous (e.g., *in particular*) or discontinuous (e.g., *either. . . or*), in which case there are exactly two parts. Our annotation does not currently include sense relations on this layer.

The RST layer follows the structural constraints defined by Mann and Thompson, and uses a relation set that is a slight adaptation of the original set. In contrast to the RST-DT, relations with centrally embedded segments are not annotated in our corpus.

Both layers (henceforth: *co* and *rr*) have been manually annotated with dedicated tools that support this process; details are given in the corpus description (Stede and Neumann, 2014). The annotators proceeded independently without consulting the other annotation layer.

In this setting, we make the following assumptions for the mapping from *co* to *rr*:

- In principle: If there is a connective, it corresponds to a relation. I.e., the mapping from *co* to *rr* should be total. The exception results from the non-annotated embedded relations; in a case like "The building, even though it is small, is quite comfortable" the *co*-annotated *even though* could not be mapped to a relation in *rr*.

- A *co* cannot signal more than one *rr*, i.e., the mapping is a function.
- Not every *rr* is signalled by an explicit connective, i.e., the mapping is not surjective.
- It is possible (if rare) that two different *co*'s (not a single, discontinuous *co*!) signal the same relation. I.e., the function is not injective.

4 The mapping algorithm

For matching the overt connectives to a corresponding discourse relation, we first converted the data to a common representation: a list of token offsets. Both a *co* and a *rr* annotation consist of two (or more, for multinuclear *rr*) segments/arguments that can be represented as token offset boundaries.

Our mapping algorithm proceeds heuristically on these token offset lists and identifies different structural categories of *co-rr* correspondences.

central We identify centrally embedded *co*'s, for which Arg2 is located within the boundaries of Arg1. As stated above, these cannot be accounted for by our RST trees.

internal Those *co*'s whose two arguments are both part of the same smallest possible *rr* segment are called *internal*. They cannot be matched to an *rr*. For example: *"It cannot be the case that expensive model projects are funded, but basic needs not met."*

exact Next we test for the existence of an *rr* whose two segments exactly match the two *co* arguments. Here we map the *co* to the corresponding *rr*.

boundary If no exact match is found, we differentiate between *local co*'s (the two arguments are adjacent to each other) and *long-distance co*'s (arguments are nonadjacent, with some intervening material). In the local case, we identify the inner segment boundary between the two connective arguments. There should be exactly one *rr* that also shares this segment boundary between its nucleus and satellite. We match the connective to this RST relation.

no match For local *co*'s, if there is no *rr* which shares the *co*'s segment boundary, we conclude a no match. This indicates a segmentation difference.

relaxed In the long-distance case, we try to find a corresponding *rr* for a *co* by matching only the left segment boundary of the (linearly) second segment. Long-distance relations are typical for backward-referring adverbials (e.g. *instead* or *therefore*), which will be captured with this heuristic. In this relaxed setting, we also allow for a one-token difference between the segment boundary of RST and the connectives, to account for possible idiosyncrasies in the connective annotation, where the *co* itself may be included or excluded from Arg2.

non-adjacent Finally, if no match is found for long-distance *co*'s, these are marked as *non-adjacent*.

5 Experiments

5.1 Data: Potsdam Commentary Corpus (PCC)

We have applied our mapping algorithm to the PCC, which consists of 175 documents taken from the editorials page of a local newspaper. The typical text length is 8 to 10 sentences, with 15.8 words on average and 1.8 verbs per sentence; the total number of tokens is roughly 32,000. This collection contains 1104 annotated connectives and 2536 RST relations.

5.2 Results

The results of the mapping algorithm, sorted by category, are shown in Table 1. Altogether, 84.4%

452	**exact match**
431	**boundary match**
49	**relaxed match**
54	central
89	internal
18	no match
11	non-adjacent
1104	connectives

Table 1: Results of the mapping process

of *co*'s could be matched to a corresponding RST relation (the bold rows in the table). This includes 48 times that two *co*'s were matched to the same *rr*. Usually these are combinations of a conjunction and an adverbial (*aber dann* 'but then') etc. The remaining 16.6% could not be matched, mostly due to design differences between the two kinds of annotations: As noted earlier, centrally embedded segments (4.9%) are not annotated in the RST

trees. In addition, 89 (8%) *co*'s were included in the PDTB-style annotation that were not accounted for in RST (the "internal" case). This group consists in large part of coordinating conjunctions that relate phrases smaller than full finite clauses (e.g., VPs or infinitives). It also includes examples where one connective argument is elliptical and very short, such as *Furchtbar, wenn* ('[It's] Terrible, when').

The few remaining failures to match (no match or non-adjacent, 2.6% in total) point to difficult cases such as two-part adverbial connectives (*zwar...aber* 'admittedly... but') or true long-distance relations. The latter relate to the distinction made by (Webber, 2006), who points out that RST analysis corresponds to a constituency structure in syntax, and PDTB analysis also accounts for dependency structure (with a corresponding distinction between 'structural' and 'anaphoric' connectives). All cases in these groups bear future study.

The majority of *co*'s matches exactly one *rr*. The 12 most common *co*'s and the *rr*'s they frequently map to are shown in Table 2.

The results in the main confirm our basic assumptions (as presented in Section 3). The vast majority of connectives match exactly one RST relation. Mismatches are due to the different segment definition in the two annotation layers and to differences in the treatment of long-distance relations between the two approaches to discourse structure (local/lexicalized vs. global). On the other hand, of the 2536 RST relations, only 932 were marked by an explicit connective, showing that the majority of rhetorical relations in our corpus is unsignalled (63%), at least by connectives in the traditional sense. This number corresponds closely to previously reported signalling ratios (Stede, 2011, p. 110). Double marking of the same *rr* was rare (48 instances, less than 2%).

Connective	RST Relation
aber (74) 'but'	concession : 21 antithesis : 18 background : 6 list : 6 joint : 5 interpretation : 4
auch (29) 'also'	list : 13 background : 3 elaboration : 3 joint : 3
dann (35) 'then'	condition : 5 result : 5 sequence : 4 reason : 3
denn (50) 'since'	reason : 32 evidence : 8 cause : 4 interpretation : 4
deshalb (22) 'therefore'	reason : 13 cause : 2 interpretation : 2
doch (83) 'however'	concession : 30 antithesis : 20 contrast : 8 interpretation : 5 reason : 5
oder (25) 'or'	list : 6 disjunction : 6
so (25) 'thus'	reason : 7 condition : 4 evidence : 4
sondern (22) 'but instead'	antithesis : 16 conjunction : 2
und (243) 'and'	conjunction : 94 list : 22 joint : 15 cause : 9 elaboration : 8 ... and 13 further relations
weil (22) 'because'	cause : 16 reason : 3
wenn (75) 'if, when'	condition : 38 circumstance : 13 interpretation : 5

Table 2: Connectives and their signalled relations

6 Related Work

The general literature on coherence relations and their signals is vast but not the main issue of this paper. We mention here a recent study that realizes an annotation project somewhat similar to ours: (Taboada and Das, 2013) add a layer of signalling information to the existing RST annotations in the RST-DT. The authors emphasize that a very wide range of signals (syntactic constructions, layout, genre, etc.) "beyond connectives" is instrumental for coherence. Again, while the work shares our spirit of multi-layer annotation, the range of signals is a separate issue; we believe that connectives—used roughly in the sense as in PDTB—are the clearest class of signals and can be annotated with high reliability; and since both PDTB-style and RST-style annotation is used widely, we regard the task of mapping between the two as of general interest. Finally, we see it as important to correlate two annotations that arose independently; the goal of Taboada and Das is different in that they first inspect the RST relation and then, "assuming the relation annotation is correct" (p. 259) actively search for the signals of that particular relation.

Very recently, attention has centered on mapping different sense hierarchies characterizing discourse

relations to each other. In this line of research, (Rehbein et al., 2016) annotated the explicit and implicit connectives in a corpus of spoken dialogues with semantic relations from the PDTB schema and according to the Cognitive approach to Coherence Relations (CCR, (Sanders et al., 1992)). The authors then map the two sense hierarchies onto one another. Relatedly, (Lapshinova-Koltunski et al., 2015) take a multilingual view in annotating discourse relations and coherence devices across languages and genres, employing different annotation schemas. However, since both approaches use the same basic items as the carriers of discourse relations/structure for each of their annotations (namely, explicit or implicit connectives), structural differences cannot be identified using these methods, which operate on the level of semantic hierarchies.

7 Summary and Conclusion

We provided a procedure for mapping connective annotation (in PDTB style) to RST annotation on the same corpus. The underlying theories play somewhat different roles for discourse analysis, yet one would expect them to be in general compatible; therefore, a systematic comparison of annotated data can reveal points of ambiguity, lack of clarity, or simplification in one of the two conceptions. Here, we gave initial results on the connective-relation mapping in the Potsdam Commentary Corpus. Of particular interest for our future work are the cases where an argument of a connective is not present in the RST analysis, and where this is not due to a straightforward difference in grain size. We will inspect these cases in order to find out whether they are due to ambiguity (a relation can be read as involving a longer or a shorter argument span; both analyses are plausible) or to simplification on the part of RST (the relation perceived in the connective annotation is simply absent in the RST tree, because multiple relations are not allowed by the theory).

In addition to testing the theories, we pointed out that the technique can be useful for a mutual validation of the annotations – the mapping can be used to identify certain annotation errors or guideline inconsistencies.

The PCC data with the two annotation layers is available via our website[1].

[1] http://angcl.ling.uni-potsdam.de/resources/pcc.html

Acknowledgments

Part of the work reported in this paper was funded by Deutsche Forschungsgemeinschaft via SFB 632 *Information Structure*. We thank the anonymous reviewers for their constructive suggestions for improving the paper.

References

L. Carlson, D. Marcu and M.E. Okurowski. 2003. Building a Discourse-Tagged Corpus in the Framework of Rhetorical Structure Theory. In: J. van Kuppevelt and R. Smith, eds. *Current Directions in Discourse and Dialogue*. Kluwer, Dordrecht.

B. Fraser. 1999. What are discourse markers? *Journal of Pragmatics*, 31(7):931–952.

E. Lapshinova-Koltunski, A. Nedoluzhko and K.A. Kunz. 2015. Across Languages and Genres: Creating a Universal Annotation Scheme for Textual Relations. In *Proceedings of The 9th Linguistic Annotation Workshop*. Denver, Colorado, USA.

W. Mann and S. Thompson. 1988. Rhetorical Structure Theory: Towards a Functional Theory of Text Organization. *TEXT*, 8:243–281.

R. Pasch, U. Braußе, E. Breindl and U.H. Waßner. 2003. *Handbuch der deutschen Konnektoren*. Walter de Gruyter, Berlin/New York.

R. Prasad, N. Dinesh, A. Lee, E. Miltsakaki, L. Robaldo, A. Joshi and B. Webber. 2008. The Penn Discourse Treebank 2.0. In *Proc. of the 6th International Conference on Language Resources and Evaluation (LREC)*. Marrakech, Morocco.

I. Rehbein, M. Scholman and V. Demberg. 2016. Annotating Discourse Relations in Spoken Language: A Comparison of the PDTB and CCR Frameworks. In *Proc. of the 10th International Conference on Language Resources and Evaluation (LREC)*. Portoroz, Slovenia.

T. Sanders, W. Spooren and L. Noordman. 1992. Toward a taxonomy of coherence relations. *Discourse Processes*, 15:135.

M. Stede. 2011. *Discourse Processing*. Morgan & Claypool.

M. Stede and A. Neumann. 2014. Potsdam Commentary Corpus 2.0: Annotation for Discourse Research. In *Proc. of the International Conference on Language Resources and Evaluation (LREC)*. Reykjavik.

M. Taboada and D. Das. 2013. Annotation upon annotation: Adding signalling information to a corpus of discourse relations. *Dialogue & Discourse*, 4(2):249–281.

B. Webber. 2006. Accounting for discourse relations: Constituency and dependency. *Intelligent linguistic architectures*, 339–360.

Automatic normalisation of the Swiss German ArchiMob corpus using character-level machine translation

Yves Scherrer
CUI / Department of Linguistics
University of Geneva
Geneva, Switzerland
yves.scherrer@unige.ch

Nikola Ljubešić
Dept. of Knowledge Technologies
Jožef Stefan Institute
Ljubljana, Slovenia
nikola.ljubesic@ijs.si

Abstract

The Swiss German dialect corpus ArchiMob poses great challenges for NLP and corpus linguistic research due to the massive amount of variation found in the transcriptions: dialectal variation is combined with intra-speaker variation and with transcriber inconsistencies. This variation is reduced through the addition of a normalisation layer. In this paper, we propose to use character-level machine translation to learn the normalisation process. We show that a character-level machine translation system trained on pairs of segments (not pairs of words) and including multiple language models is able to achieve up to 90.46% of word normalisation accuracy, an error reduction of 45% over a strong baseline and of 34% over a heterogeneous system proposed by Samardžić et al. (2015).

1 Introduction

The term Swiss German covers a range of German varieties spoken in the Northeastern two thirds of Switzerland. Despite the widespread (almost exclusive) use of dialects in speech and in electronic media, only few resources adapted for research in NLP are currently available. One recent resource is the ArchiMob corpus of transcribed speech (Samardžić et al., 2016), which is used in the experiments presented here.

This paper addresses the orthographic inconsistency and dialectological variation typical for Swiss German texts through the addition of a normalisation layer. Normalisation, i.e. mapping the variants of what can be identified as the same word to a single representation, is necessary for any task that requires establishing lexical identities. Such tasks include building an efficient corpus query interface for linguistic research, semantic processing, and information retrieval.

We propose to view the normalisation task as a translation task from inconsistently written texts to a unified representation. We show that a single, optimised character-level machine translation system fares better than the heterogeneous system proposed by Samardžić et al. (2015).

2 Related work

Swiss German has been the object of extensive dialectological research for more than 100 years, which has led to major contributions such as dialect atlases (Hotzenköcherle et al., 1962–1997; Bucheli and Glaser, 2002) and comprehensive dialect dictionaries (Staub et al., 1881–). However, dialect corpora have started being collected only recently. Siebenhaar (2005) creates a corpus of interactions in Swiss German internet relay chat rooms. Hollenstein and Aepli (2014) compile a corpus of written Swiss German texts and use it to train and test part-of-speech tagging models. Stark et al. (2009–2015) collect, normalise, and part-of-speech tag a corpus of SMS messages. The ArchiMob corpus used in this work has been presented together with first experiments on automatic normalisation and part-of-speech tagging in Samardžić et al. (2015) and Samardžić et al. (2016).

Due to the lack of standardised spelling, dialect texts face problems that are similar to other types of non-standard data such as historical text, spoken language or computer-mediated communication. Normalisation (also called modernisation in the context of historical language) has been proposed to deal with this heterogeneity. Automatic word normalisation has been addressed through several approaches in the community of historical NLP, such as automatic induction of rules (Reffle, 2011; Bollmann, 2012), similarity-based form matching inspired by spellchecking (Baron and Rayson, 2008; Pettersson et al., 2013), and character-level machine translation (Pettersson et al., 2014; Scherrer and Erjavec, 2015). Character-level machine

translation (CSMT) has originally been proposed for translation between closely related languages (Vilar et al., 2007; Tiedemann, 2009), but has proven successful in many other settings where regular changes occur at the character level, including the normalisation of computer-mediated communication (De Clercq et al., 2013; Ljubešić et al., 2014).

3 Data

The ArchiMob corpus contains transcriptions of video recordings collected in the context of an oral history project (see `http://www.archimob.ch`) between 1999 and 2001. Currently, the corpus consists of 34 transcriptions of interviews conducted in various Swiss German dialects (Samardžić et al., 2016).

The recordings were transcribed manually by native speakers of Swiss German, using the Dieth guidelines (Dieth, 1986). These are general guidelines that can be interpreted and implemented in several ways, leading to some inconsistencies in the transcriptions. Furthermore, there is a considerable amount of pronunciation variation in the texts, on the intra-speaker level as well as on the dialect level. For example, the first person possessive pronoun in its masculine singular form (Standard German *mein*) has been transcribed in the four variants *min*, *miin*, *mi*, *mii* in a single text, reflecting different pronunciations by the same speaker. When other texts are considered, a fifth variant, *mine*, can be added. While the transcription inconsistencies could be eliminated with more precise guidelines, there is no obvious way to reduce the intra-speaker variation and the dialectal variation at the transcription level. Therefore, it was decided to annotate each original word form with a normalised form. The goal of normalisation is to reduce all variants that can be identified as "the same word" to a single form. At the moment, a subset of 6 recordings have been manually normalised, and the plan is to normalise the remaining documents in a semi-automatic way.

Normalisation is performed word by word. In most cases, the normalised forms resemble Standard German (see Figure 1 for an example), with two major divergences from this principle. First, Swiss German lexical items that do not have an etymologically related Standard German counterpart are not translated, but rather normalised using a convenient, etymologically motivated common construction. For instance, *gluegt* 'looked' is not translated to the semantic equivalent *geschaut*, but normalised to the reconstructed form *gelugt*.[1] Second, word boundaries in Swiss German may differ from the Standard German ones due to cliticisation effects, in which case one Swiss German word corresponds to more than one word in the normalisation layer, as illustrated by *dasch* 'this is' and its normalisation *das ist*.

Transcription	Normalisation	
jaa	ja	'yes'
de	dann	'then'
het	hat	'has'
me	man	'one'
no	noch	'still'
gluegt	gelugt	'looked'
tänkt	gedacht	'thought'
dasch	das ist	'this is'
ez	jetzt	'now'
de	der	'the'
genneraal	general	'general'

Figure 1: A transcribed and normalised utterance extracted from the corpus.

4 Automatic normalisation

First experiments aiming at learning the normalisation process were reported in Samardžić et al. (2015) and Samardžić et al. (2016). To this end, the words in the test set were partitioned in four classes, and different normalisation methods were chosen according to the word class:

- *Unique* words are associated with exactly one normalisation in the training set. At test time, these words were normalised using the normalisation seen during training.
- *Ambiguous 1* words are associated with more than one normalisation candidate, but a unique most frequent normalisation can be determined. In this case, the most frequent normalisation was proposed at test time.
- For *Ambiguous 2* words, no single most frequent normalisation can be selected because of tied frequency counts. For this class, it was proposed to select the best normalisation

[1] These reconstructions were generally inspired by the lemmas of the *Idiotikon* dialect dictionary (Staub et al., 1881–).

		Baselines and ceilings				Isolated words		Segments		Constr.
	Prop.	Ident.	Baseline	Combi	Ceiling	1 LM	2 LM	1 LM	2 LM	2 LM
Unique	46.63	22.84	98.79	98.79	98.98	98.30	98.22	97.25	97.64	98.69
Ambig.	42.12	23.52	84.06	84.20	84.64	83.45	82.52	86.27	87.54	87.92
New	11.25	9.46	9.46	35.33	99.57	53.15	53.91	52.50	63.59	65.87
All		21.62	82.54	85.51	93.00	86.96	86.62	87.59	89.56	90.46

Table 1: Percentages of correctly normalised words using the different comparison methods (left), the CSMT system applied to isolated words (center), the CSMT system applied to segments (right), and the segment-level system with constraints (rightmost). The first column shows the proportion of the three word classes in the test corpus. The *All* row refers to the micro-averages over the three word categories.

candidate either by character-level machine translation or by a word-level language model. The latter approach yielded the best results, but the overall impact was limited as this class only accounts for about 0.5% of words.

- *New* words have not been observed in the training set and therefore no normalisation candidates are available. These words were normalised using a character-level machine translation system trained on the training data.

In this contribution, we take advantage of additional annotations that have been made available in the meantime and show that a single normalisation method based on CSMT can outperform the heterogeneous method summarised above.

The current version of the ArchiMob corpus differs from previous versions in three respects. First, the manually annotated normalisations were double-checked to ensure best consistency (Samardžić et al., 2016). Second, the utterances were split into syntactically and prosodically motivated segments of 4-8 seconds. Third, hesitations and false starts were annotated as such and could be excluded from the normalisation task, since their normalisation is not meaningful.

We argue that the CSMT approach can be extended to all four categories of words, without loss of accuracy. To this end, four types of improvements are proposed:

- All CSMT models are tuned using MERT; this has not been done in previous work.

- We propose a setting in which each word is normalised in isolation, and a setting in which entire segments are translated. While normalisation is mainly performed at the token level in related work, we obtain significant improvements by normalising entire segments, thanks to the possibility of capturing parts of the context during the normalisation process.

- We add, beside the training data language model, an additional language model of spoken Standard German to the CSMT systems.

- We make use of the possibility to include translation constraints in the CSMT system. These constraints improve the normalisation of *Unique* words while maintaining the advantage of a single decision process.

Finally, instead of using cross-validation as in Samardžić et al. (2015), given that in these experiments we need development data for tuning, we create a single data split with 80% of utterances used for training, 10% for tuning and 10% for testing. As in earlier work, utterances from all six documents are represented proportionally in each file. The training set contains 8443 segments with 65 671 words, the development set contains 1054 segments with 9032 words, and the test set contains 1055 segments with 8212 words.

5 Experiments

5.1 Baselines and ceilings

The left half of Table 1 shows several measures that indicate the difficulty of the normalisation task. The *Proportion* column shows the distribution of the three word classes established above in the test set (we merge the *Ambiguous 1* and *Ambiguous 2* classes as their distinction is not relevant for the experiments presented here). The *Identical* column indicates for how many words the normalised form is identical to their original form; overall, only

Transcription	Normalisations	Occurrences	
de	der / dann	168	'the / then'
das	das / dass	168	'the / that'
es	ein / es	69	'a / it'
i	ich / in	59	'I / in'
mer	wir / man / mir	58	'we / one / me'
s	das / es	44	'the / it'
bi	bei / bin	22	'at / am'
sii	sie / sein	22	'she / be'
e	ein / eine	79	Neut / Fem
en	ein / einen	77	Masc / Neut
cho	kommen / gekommen	3	Pres / PP
chliine	kleiner / kleinen	3	Nom / Acc

Table 2: Normalisation ambiguities observed in the ArchiMob corpus – part-of-speech ambiguities in the upper part, inflectional ambiguities in the lower part. The third column shows the number of occurrences of the transcribed form in the test set. Example sentences and phrases can be found in Table 3 at the end of the paper.

about one fifth of all words are identical to their normalisations.

The *Baseline* normalises every word by assigning it the most frequent normalisation seen in the training set. In case of ties, a normalisation is chosen randomly; for new words, no normalisation process is applied at all. This exactly corresponds to the *Word-by-word* setting in Samardžić et al. (2015). The *Combi* column shows the figures obtained by applying the *Combi* method of Samardžić et al. (2015) and Samardžić et al. (2016) to the revised data set.

In order to estimate the ceiling of the approach normalising each word in isolation, we measure the level of word ambiguity by applying the *Baseline* method trained on the whole dataset (development and test sets included) and tested on the test set. We present the results in the *Ceiling* column.[2] The presented figures show that, as expected, the highest word ambiguity is present among the *Ambiguous* words. For the *New* words, which are generally low-frequency words, ambiguities are rarely observed due to the small size of the sample. The results in Table 1 show that overall 7% of words cannot be normalised regardless of the amount of training data available if contextual information is not taken into account.

Table 2 shows the most frequent ambiguities observed in the corpus. While most ambiguities concern short words of different parts-of-speech, there are also some ambiguities that arise due to the inflectional systems of Swiss German being less rich than the Standard German ones.

5.2 Applying CSMT to isolated words

The goal of this first experiment is to show that a single CSMT model, applied indiscriminately to all three word classes, performs equally well as the *Baseline* or the *Combi* model. To this end, we train a CSMT system on the training set, tune it using MERT on the development set and apply it to the test set.[3] Each word is considered in isolation for training and testing.[4] We have found 7-gram language models to work best, and we have disabled distortion throughout all steps of the process as there is no evidence of such phenomena in our data. Table 1 (*Isolated words*) shows results with

[2]To facilitate the comparison with the other methods, the attribution of the words to the categories *Unique, Ambiguous, New* is still based on the training data only. This means e.g. that 98.98% of words that were observed with a unique normalisation in the training set are effectively unique, whereas for the remaining 1.02% of words, a second normalisation was seen in the test data, making these words ambiguous.

[3]We use the Moses toolkit (Koehn et al., 2007) together with the KenLM language model toolkit (Heafield, 2011) for all experiments. We use the standard settings except for distortion, which is completely disabled, and for the MERT optimisation objective, where we choose WER (word error rate, which *de facto* becomes character error rate in a CSMT setting) instead of BLEU.

[4]For instance, the word *tänkt* is transformed to _ *t ä n k t* _ before feeding it to the translation system. The leading and trailing underscores have proved useful for explicitly modelling word boundaries.

two settings: in the first setting (*1 LM*), we use a single language model estimated on the target side of the training set, whereas in the second setting (*2 LM*) we add a second language model estimated on the Standard German OpenSubtitles2016 corpus (Lison and Tiedemann, 2016), 108 million tokens in size.[5]

The results show that the *1 LM* system achieves only slightly lower performance than the baseline for *Unique* and *Ambiguous* words, but generalises much better than the *Combi* system for *New* words, leading to a higher overall accuracy. A comparison of the two new systems shows that the second language model does not yield any improvements. Our assumption is that there are two reasons for that: (1) the data used for estimating the second language model (Standard German) is quite different to the target data (normalised Swiss German) and (2) word-level systems do not need as much target language data as segment-level systems because there is much more variation between words than inside words.

5.3 Applying CSMT to segments

Normalising each word in isolation means that contextual clues such as the preceding and following word cannot be used for disambiguation. By evaluating our *Ceiling* system we have shown that in this dataset we cannot correctly normalise 7% of words if we translate words in isolation, regardless of the amount of training data available. Therefore, in this second experiment we propose to translate complete segments.[6] By selecting phrases that span word boundaries, the system will be able to perform (at least local) context-dependent disambiguation. The evaluation is still performed word-by-word, as before.[7]

The training, tuning and testing steps in this experiment are the same as in the first one, except that in these experiments 10-gram language models have shown to perform best, not 7-gram language models. This is expected as this system requires as much word context information as possible. Using language models of greater order than 10 did not yield any significant improvements.

The results are shown on the right side of Table 1 (*Segments*). In the *1 LM* setting, the accuracy of *Ambiguous* words improves by 2.82%, as expected. However, contextual influence has a slight negative effect on the *Unique* (-1.05%) and *New* (-0.65%) words. In the *2 LM* setting, the disambiguation of *Ambiguous* words is even more successful (+5.02% compared with the equivalent single-word model). Even more striking is the 9.68% increase for *New* words. Here, the context clearly adds useful information, but only the *2 LM* model is able to take advantage of this information since, by definition, these words do not appear in the original language model.[8]

However, this system still makes proportionally most errors with *New* words. We have found several categories of words to be prone to normalisation errors:

- In 12% of cases, the root is correctly normalised but an erroneous inflectional affix is selected, due to the inflectional ambiguities mentioned in Table 2. Especially for long compound nouns, the context window of 10 characters is not sufficient to disambiguate the candidates: *muurermäischter* 'master mason' is normalised as *maurermeister* where *maurermeistern* would be the correct form, but the relevant case and number information encoded in the preceding determiner is not accessible.

- 9% of errors concern named entities like place or person names: *buechs* is normalised as *buchs* instead of *buochs* (a town name), *riintel* 'Rhine valley' is normalised as *reintal* instead of *rheintal*. These entities are unlikely to occur in the added language model.

- 8% of errors concern abreviations or foreign words, in which the learned normalisation patterns do not apply: *komfiserii* 'confectionary' is normalised as *konfiserei* instead of *confiserie*, *kaazèt* 'concentration camp' is normalised as *kazat* instead of *kz*, *plimut* is normalised as *pleinmut* instead of *plymouth* (a brand name).

[5] We removed punctuation and lowercased the corpus to make it most similar to our normalisation language.

[6] Recall that a segment is about 4–8 seconds long and contains around 8 words on average.

[7] Segments are transformed in the same way as isolated words, using underscores to mark word boundaries. After translation, the segments are split at the underscores in the source for evaluation. This step is not trivial as there may be different numbers of underscores in the source and target due to the differences in word boundaries illustrated in Figure 1.

[8] Similar trends have been observed for Slovene historical texts and user-generated content (Ljubešić et al., 2016), although the improvements are less marked in Slovene because token ambiguity is lower than in our Swiss German data.

- About 2% of mismatches were due to mistakes and typos in the gold normalisations.

5.4 Adding constraints to the segment model

While the segment-level system outperforms the baseline on *Ambiguous* words and has produced significant improvements for *New* words, it still lags behind the baseline by more than 1% regarding *Unique* words. One simple yet effective improvement is to constrain the segment-level system so that the baseline normalisation is chosen for *Unique* words. Moses supports XML annotations to this effect. We used the segment-level *2 LM* system as a basis, retuned it with the annotated development data and tested it on the annotated test data. We have found the *exclusive* strategy to work slightly better than the *constraint* strategy.[9]

The results of this hybrid system are shown in the rightmost column of Table 1 (for reasons of space, we only show results for the *2 LM* system). For the *Unique* words, the accuracy is now very close to the baseline. The remaining errors concern three very long words that are not normalised at all despite the presence of a baseline normalisation; we suspect this to be a bug in Moses.

While it is not surprising that the constrained system outperforms the segment-level system for *Unique* words, it is striking that the accuracies also rise for the *Ambiguous* and *New* words. The contextual information provided by the *Unique* word annotation also positively impacts the adjacent non-unique words. Overall, the constrained system outperforms the pure segment-level system by 0.9%. We assume that, if enough target language data was present in the system (and not the near-target data of Standard German), these constraints would not be necessary.

Given that this is the smallest recorded difference among all the comparisons throughout the paper, we ran three MERT tuning processes on both systems and calculated on each output the approximate randomisation statistical test (Yeh, 2000) with 1000 iterations to measure the probability of observing the difference by chance. The highest p-value measured on any of the three outputs was $p < 0.001$ showing that the observed difference of $\sim$1% on our test set is already highly significant.

[9] See Section 4.8.2 of the Moses manual, consulted at `http://www.statmt.org/moses/manual/manual.pdf` on 2016-06-06. We have also found that adding the word boundary symbol to the baseline normalisations is useful to prevent spurious suffixes from being appended.

6 Conclusion

In this paper, we have shown that character-level machine translation can be used successfully to learn the process of automatically normalising dialect texts with heterogeneous transcriptions. Translation systems operating on isolated words obtain accuracy levels comparable with previous work for *Unique* and *Ambiguous* words, whereas significant improvements are observed for *New* words. Systems operating on entire segments yield accuracy gains for *Ambiguous* words and, when combined with an additional language model, for *New* words. Constraining the translation of *Unique* words allows to further improve overall accuracy by nearly 1%.

However, there is still room for improvement. In particular, several extensions may be envisaged to improve the treatment of ambiguous words and long range dependencies:

- A language model that operates on the word level (instead of the character level) would allow us to keep track of larger context windows. Such an additional language model could be integrated into the translation process using Moses feature functions.

- Adding part-of-speech tags as an additional word-level feature may also be useful to disambiguate words. Samardžić et al. (2016) have showed that respectable tagging performance can be achieved without using the normalised forms, but it is open whether such a tagger is able to reliably resolve the ambiguities mentioned in Table 2.

- Neural language modeling could learn the morphosyntactic regularities and long distance dependencies of the language much better than the surface n-gram language model currently used.

- Increasing the language model order and/or the maximum phrase length could alleviate the difficulties observed when the normalised form is much longer than the original form. For example, *nüm* 'no more' should be normalised as *nicht mehr*, but incomplete and wrong forms such as *nicht m* or *nicht man* are produced instead by the current models.

Finally, it is expected that normalising additional unseen texts will yield a lot of *New* words that are

	Transcription	Normalisation	
de	jaa *de* het me no gluegt tänkt dasch ez *de* genneraal	ja *dann* hat man noch gelugt gedacht das ist jetzt *der* general	'yes then one still watched and thought, now this is the General'
das	ich wäiss aber nüme wele leerer *das* *das* gsii isch	ich weiss aber nicht mehr welcher lehrer *dass das* gewesen ist	'but I don't know any more which teacher it was'
es	*es* huus *es* isch	*ein* haus *es* ist	'a house' 'it is'
mer	*mer* händ *mer* hät er hät *mer* tanket	*wir* haben *man* hat er hat *mir* gedankt	'we have' 'one has' 'he thanked me'
s	*s* huus *s* isch	*das* haus *es* ist	'the house' 'it is'
bi	*bi* frauefeld ich *bi* nöd dehäim gsii	*bei* frauenfeld ich *bin* nicht daheim gewesen	'near Frauenfeld' 'I was not at home'
sii	wüssed *sii* ich han wele schwiizeri *sii*	wissen *sie* ich habe wollen schweizerin *sein*	'you know' 'I wanted to be Swiss'
e	*e* kino *e welotuur*	*ein* kino *eine* velotour	'a cinema' 'a bike tour'
en	si hät *en* gaarte ghaa und dän isch *en* bueb ufgschtande	sie hat *einen* garten gehabt und dann ist *ein* bub aufgestanden	'she had a garden' 'and then a boy stood up'
cho	de händs müse hääi *cho* si isch uf d wält *cho*	dann haben sie müssen heim *kommen* sie ist auf die welt *gekommen*	'then they had to come home' 'she was born'
chliine	er isch en *chliine* gsii im *chliine* chileli	er ist ein *kleiner* gewesen im *kleinen* kirchlein	'he was a small one' 'in the small church'

Table 3: Examples of the normalisation ambiguities shown in Table 2.

named entities. To address this issue, we investigate the inclusion of a lexicon containing toponyms and patronyms of German-speaking Switzerland.

Acknowledgements

Construction and distribution of the ArchiMob corpus was supported by the University of Zurich URPP Language and Space. In particular, we would like to thank Tanja Samardžić, Noëmi Aepli and Fatima Stadler for making the corpus available with the improvements mentioned in Section 4.

The research leading to these results has received funding from the Swiss National Science Foundation grant no. IZ74Z0_160501 (ReLDI).

References

Alistair Baron and Paul Rayson. 2008. VARD 2: A tool for dealing with spelling variation in historical corpora. In *Proceedings of the Postgraduate Conference in Corpus Linguistics*, Aston University.

Marcel Bollmann. 2012. (Semi-)automatic normalization of historical texts using distance measures and the Norma tool. In *Proceedings of the Second Workshop on Annotation of Corpora for Research in the Humanities (ACRH-2)*, pages 3–14, Lisbon, Portugal.

Claudia Bucheli and Elvira Glaser. 2002. The syntactic atlas of Swiss German dialects: Empirical and methodological problems. In Sjef Barbiers, Leoni Cornips, and Susanne van der Kleij, editors, *Syntactic Microvariation*, volume 2, pages 41–73, Amsterdam. Meertens Institute Electronic Publications in Linguistics.

Orphée De Clercq, Bart Desmet, Sarah Schulz, Els Lefever, and Véronique Hoste. 2013. Normalization of Dutch user-generated content. In *Proceedings of RANLP 2013*, pages 179–188, Hissar, Bulgaria.

Eugen Dieth. 1986. *Schwyzertütschi Dialäktschrift*. Sauerländer, Aarau, 2 edition.

Kenneth Heafield. 2011. KenLM: faster and smaller language model queries. In *Proceedings of the EMNLP 2011 Sixth Workshop on Statistical Machine Translation*, pages 187–197, Edinburgh.

Nora Hollenstein and Noëmi Aepli. 2014. Compilation of a Swiss German dialect corpus and its application to PoS tagging. In *Proceedings of the First Workshop on Applying NLP Tools to Similar Languages, Varieties and Dialects (VarDial)*, COLING 2014, Dublin, Ireland. Association for Computational Linguistics.

Rudolf Hotzenköcherle, Robert Schläpfer, Rudolf Trüb, and Paul Zinsli, editors. 1962–1997. *Sprachatlas der deutschen Schweiz*. Francke, Bern.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the ACL 2007 demonstration session*, pages 177–180, Prague, Czech Republic.

Pierre Lison and Jörg Tiedemann. 2016. OpenSubtitles2016: Extracting large parallel corpora from movie and TV subtitles. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*.

Nikola Ljubešić, Katja Zupan, Darja Fišer, and Tomaž Erjavec. 2016. Normalising Slovene data: historical texts vs. user-generated content. In *Proceedings of KONVENS 2016*.

Nikola Ljubešić, Tomaž Erjavec, and Darja Fišer. 2014. Standardizing tweets with character-level machine translation. In *Proceedings of CICLing 2014*, Lecture notes in computer science, pages 164–175, Kathmandu, Nepal. Springer.

Eva Pettersson, Beáta B. Megyesi, and Joakim Nivre. 2013. Normalisation of historical text using context-sensitive weighted Levenshtein distance and compound splitting. In *Proceedings of the 19th Nordic Conference of Computational Linguistics (Nodalida 2013)*, pages 163–79, Oslo, Norway.

Eva Pettersson, Beáta B. Megyesi, and Joakim Nivre. 2014. A multilingual evaluation of three spelling normalisation methods for historical text. In *Proceedings of the 8th Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities (LaTeCH)*, pages 32–41, Gothenburg, Sweden.

Ulrich Reffle. 2011. Efficiently generating correction suggestions for garbled tokens of historical language. *Natural Language Engineering*, 17:265–82.

Tanja Samardžić, Yves Scherrer, and Elvira Glaser. 2015. Normalising orthographic and dialectal variants for the automatic processing of Swiss German. In *Proceedings of The 4th Biennial Workshop on Less-Resourced Languages, Seventh Language and Technology Conference*.

Tanja Samardžić, Yves Scherrer, and Elvira Glaser. 2016. ArchiMob – a corpus of spoken Swiss German. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*.

Yves Scherrer and Tomaž Erjavec. 2015. Modernising historical Slovene words. *Natural Language Engineering*, pages 1–25. Available on Cambridge Journals Online.

Beat Siebenhaar. 2005. Die dialektale Verankerung regionaler Chats in der deutschsprachigen Schweiz. In Eckhard Eggers, Jürgen Erich Schmidt, and Dieter Stellmacher, editors, *Moderne Dialekte – Neue Dialektologie*, pages 691 – 717. Steiner, Stuttgart.

Elisabeth Stark, Simone Ueberwasser, and Beni Ruef. 2009–2015. Swiss SMS corpus, University of Zurich. `https://sms.linguistik.uzh.ch`.

Friedrich Staub, Ludwig Tobler, Albert Bachmann, Otto Gröger, Hans Wanner, and Peter Dalcher, editors. 1881–. *Schweizerisches Idiotikon: Wörterbuch der schweizerdeutschen Sprache*. Huber, Frauenfeld.

Jörg Tiedemann. 2009. Character-based PSMT for closely related languages. In *Proceedings of EAMT 2009*, pages 12–19, Barcelona, Spain.

David Vilar, Jan-Thorsten Peter, and Hermann Ney. 2007. Can we translate letters? In *Proceedings of the Second Workshop on Statistical Machine Translation*, pages 33–39, Prague, Czech Republic.

Alexander Yeh. 2000. More accurate tests for the statistical significance of result differences. In *Proceedings of COLING 2000*, pages 947–953.

Part-Of-Speech in Historical Corpora: Tagger Evaluation and Ensemble Systems on ARCHER

Gerold Schneider
University of Konstanz
and University of Zurich
gschneid@es.uzh.ch

Marianne Hundt
University of Zurich
m.hundt@es.uzh.ch

Rahel Oppliger
University of Zurich
rahel.oppliger@uzh.ch

Abstract

Tagger accuracy deteriorates when applied to texts different from the training corpus, e.g. with respect to register or time period. On historical data, accuracy can drop to and below 90%. We are tagging and parsing ARCHER, a historical corpus sampled from British and American texts from 1600-1999. We improve tagging accuracy by (1) using a version of the corpus that has been automatically mapped to PDE spelling with VARD, (2) by combining several part-of-speech taggers in an ensemble system – which improves tagging by about 1% over CLAWS and 2% over Tree-Tagger, and (3) by using a small amount of human intervention – which allows us to reach 98% accuracy from 1700 on.

1 Introduction

Part-of-speech tagging accuracy strongly deteriorates when a tagger is applied to texts which are different from the training domain. Typically, taggers are trained on present-day English (PDE) texts, specifically news texts, mostly from the Penn Treebank (Marcus et al., 1993). They then reach 95-97% accuracy on PDE texts of the same register given that tokenisation is perfect. If these conditions are not met, accuracy can drop to and below 90%. For example, Rayson et al. (2007) report that the CLAWS tagger (Garside and Smith, 1997) achieves 96 to 97% accuracy on PDE, while on Early Modern English, performance drops to 81.9% on Shakespeare texts and to 88.5% on pamphlets from the Lampeter corpus.

A major source for errors are historical spelling variants. There are two possible strategies for dealing with spelling variants: either the tagger is adapted to cope with the variant directly, or the spelling variants are normalised to PDE forms, as expected by the tagger. We have chosen the second option.

2 Data and Methods

2.1 The ARCHER Corpus

As corpus of application, we chose ARCHER (Biber et al., 1994), a historical corpus sampled from British and American texts from 1600-1999 and across several registers. Its current version (V 3.2) contains 3.2 million words. We improve tagging accuracy by using a version of the corpus that has been automatically mapped to PDE spelling with VARD, by combining several part-of-speech taggers in an ensemble system, and by using a small amount of human intervention.

2.2 Spelling Normalisation

A major source for errors are historical spelling variants. Simple variants like *call'd* for *called* typically result in wrong tagging (*call_NN d_MD*), in this case triggered by a tokenisation error, and as a consequence parsing quality is also affected.

There are two possible strategies for dealing with spelling variants. In the first strategy, the tagger is adapted to cope with the variant directly. Yang and Eisenstein (2016) present an approach using domain adaptation which has very high accuracy. They also argue that their approach circumvents the partly ill-defined task of normalisation. In the second strategy, the spelling variants are normalised to PDE forms, as expected by the tagger. We have chosen the second option. Compared to domain adapation, normalising approaches have the advantage that they allow linguists to search for all occurrences of a word form, with a single and obvious query. Spelling normalisation, according to Rayson et al. (2007), increases tagging accuracy to 85% for the Shakespeare texts, and to 89% for the Lampeter texts, when using the automatic normalisation tool VARD (Baron and Rayson, 2008). They also give an upper bound of their approach by using manual

Tree-Tagger: It_PRP adds_VBZ much_**JJ/RB** to_**TO** my_PRP$ satisfaction_NN ,_, that_IN her_PRP$ Character_**NNP** is_VBZ agreeable_JJ to_**TO** your_PRP$ Fancy_**NNP**
CLAWS Tagger: It_PRP adds_VBZ much_**RB/DT** to_**IN** my_PRP$ satisfaction_NN ,_, that_IN her_PRP$ Character_**NN** is_VBZ agreeable_JJ to_**IN** your_PRP$ Fancy_**NN**
CandC Tagger: It_PRP adds_VBZ much_**RB** to_**TO** my_PRP$ satisfaction_NN ,_, that_IN her_PRP$ Character_**NNP** is_VBZ agreeable_JJ to_**TO** your_PRP$ Fancy_**NNP**

Table 1: Sample outputs from Tree-Tagger, CLAWS tagger and CandC (ARCHER 1671cary_d2b)
PENN tags: JJ=adjective, RB=adverb, DT=determiner, TO='to', IN=preposition, NN=common noun, NNP=proper name

normalisation: 89% for Shakespeare, and 91% for Lampeter. In other words, about half of the tagging errors could be corrected.

2.3 Fully Automatic Ensemble System

Different taggers make different mistakes, as they use different algorithms, tags, and partly different training sets. They thus offer different perspectives on same data. Combinations of different systems, which are also called ensemble systems, can benefit from their mutual advantages, as long as the individual participating systems are quite accurate and diverse (Dietterich, 1997; van Halteren et al., 2001) We use the following three taggers: Tree-Tagger, CLAWS, CandC. They are presented briefly in the following.

The **Tree-Tagger** (Schmid, 1994) is a decision-tree tagger. In additon to the most likely tag, it also offers n-best tagging as an option. N-best tagging returns the *n* most likely tags, together with an estimate of the probability of each tag, given the language model.

The **CLAWS tagger** (Leech et al., 1994; Garside and Smith, 1997) is a hybrid system which combines probabilistic and rule-based approaches. Like the Tree-Tagger, it also reports n-best tags including probabilities. We map the original CLAWS5 tagset automatically to the Penn tagset. The mapping list is for example given in Wu (2010, 97). The CLAWS5 tagset comprises of 62 tags and is thus more fine-grained than the Penn Treebank tagset with its 39 tags. Mapping from CLAWS5 to the Penn tagset is mostly deterministic, and depends on the tag only. There are exceptions, though, the most notable being the fact that CLAWS5 disambiguates between *to* as infinitive particle and as preposition, while Penn gives the tag *TO* to both. We count both tags as correct in our evaluation.

The fact that CLAWS uses a different tagset offers an additional alternative perspective to us. While a larger tagset leads to a lower baseline and has the risk that the tagger needs to take potentially more difficult decisions, this potential disadvantage should disappear if a reliable mapping procedure to the more coarse-grained tagset is used. In fact, a larger tagset can also facililate the task: if particular features strongly point to a rare tag, the accuracy of recognition can in fact increase.

The **CandC tagger** (Curran et al., 2007; Grover, 2008) is a maximum-entropy tagger, as it distributed as part of the XML pipeline LT-TTT2[1].

Table 1 gives an example of the outputs by the three taggers. The differring parts are highlighted. Double-tags are given if the tagger in n-best mode outputs several tags. The tag closer to the word is the higher ranked tag. We see in this sentence that except for CLAWS, taggers tend to asssign proper name (NNP) to capitalised words. We also see that CLAWS aims to distinguish between *to* as preposition and infinitive particle.

After comparing the accuracies for each tagger in section 3.1, we show in section 3.2 that a fully automatic ensemble approach increases the accuracy. We experiment with the following methods:

Majority voting Majority voting checks if 2 of our 3 taggers agree. If they do, the majority tag is selected.

Best probability Best probability compares the probabilites that the two n-best taggers (Tree-tagger and CLAWS) return. The probabilities can be interpreted as scores, as an estimation of the tagger's confidence in its decision. The tag with the highest probability score is selected, giving precedence to the tagger which has higher confidence in its decision.

Systematic advantage It is also possible that one tagger can be trusted more, either generally or in specific cases, as it may be better adapted. For example, CLAWS correctly tags *hath* and *hast* as verb.

2.4 Semi-Automatic: Limited Human Intervention

In section 3.2 we apply methods that need limited human intervention. In these approaches, a human

[1] https://www.ltg.ed.ac.uk/software/lt-ttt2/

Tagger	Tree-Tagger	CLAWS	CandC	Best Ensemble	Error Rate Reduction	Best Oracle
16xx	87.4	87.8	82.8	88.8 (+1.0)	8.9	94.2
17xx	91.0	93.2	85.4	93.4 (+0.2)	3.0	98.2
18xx	95.2	95.0	91.8	95.6 (+0.4)	13.6	98.2
19xx	92.1	92.8	86.2	94.1 (+1.3)	22.0	98.3

Table 2: Accuracy (percent) of individual taggers and best combinations, split by century

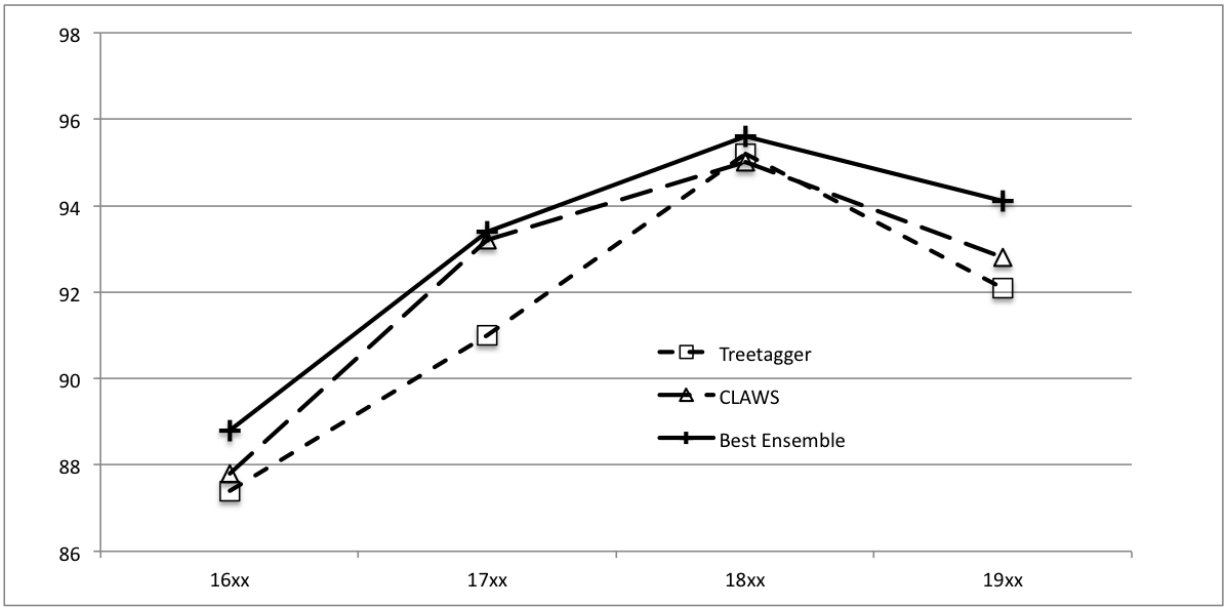

Figure 1: Tagging performance of Tree-Tagger, CLAWS, and the ensemble system

needs to chose between one of maximally three candidates. We use the following two methods. First, a tagger-internal choice: if a tagger offers several tags in n-best mode, is one of them correct? Second, the highest ranked tag suggested by each tagger is considered: if the taggers disagree, does one of them suggest the correct tag? These approaches can also be described as Oracle approaches which measure the upper bound of the taggers.

3 Results

3.1 Individual Taggers

We split the corpus into four periods – each comprising one century – and manually annotated at least 500 words from each period. The manual annotations were cross-checked by two authors and discussed until an agreement could be reached. The accuracy of each tagger is given in Table 2, columns 2 to 4. CLAWS is on average 0.78% better than Tree-Tagger. As the performance of CandC was considerably worse, we excluded it from most ensemble experiments, which we explain in the following.

3.2 Automatic Combinations

Probabilities for the most likely tags are delivered by CLAWS and Tree-Tagger in n-best tagging mode. The probabilities can be interpreted as confidence scores. If we always choose the tag whose confidence score is highest from these two taggers (Best Ensemble), we can automatically increase performance by 0.78% on average over the better performing tagger, CLAWS, as Figure 1 shows. The increase over Tree-Tagger alone is between 1 and 2 percent. The exact percentages are given in Table 2, column 4. In terms of error rate reduction, between 3% and 22% of the errors could be corrected by the best Ensemble approach, as column 5 shows.

We have also tried majority voting, but due to the relatively low performance of the CandC tagger the combined performance stays below CLAWS as single tagger.

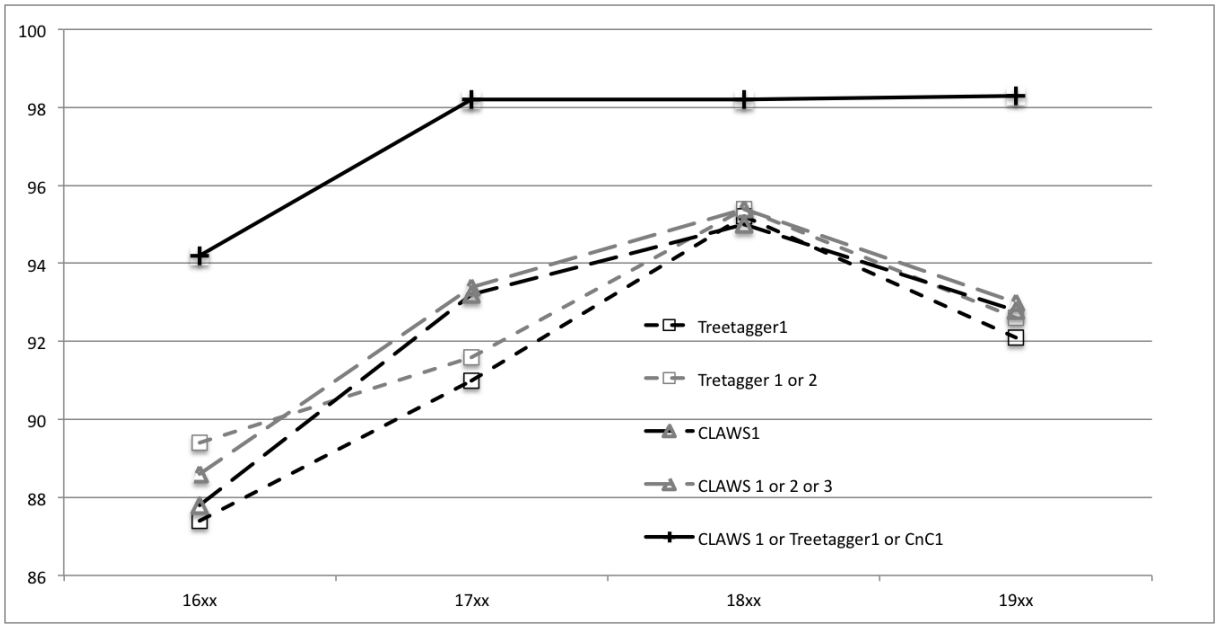

Figure 2: Performance with limited human intervention: choose one of three in ambiguous cases

3.3 Semi-Automatic Combinations

With limited human intervention, performance can be further improved if a human chooses either one of the maximally 3 most promising n-best tags from the same tagger, or the top tag from the three different taggers. Figure 2 shows the results of both approaches. Choosing between several options in n-best mode increases performance only slightly. A major reason for the modest improvement is that alternative tags are only suggested for a small minority of all word tokens: about 5% in the earliest texts, and about 2% in the 20^{th} century.

The second option – manually selecting the top tag if the taggers disagree – leads to a strong improvement, by 2-5 %, to above 98% except in the $17t^{th}$ century, as Table 2, last column, shows. Disagreement between taggers is quite frequent though: in the 20^{th} century, all three taggers suggest the same tag in 423 out of the 529 words in the evaluation set; in 106 cases (20% of all words) the user needs to select the correct tag. In terms of entropy, we can observe that on average, there are 1.31 tags to choose from per word. Split by century, there are 1.36 tags per word in the 16xx texts, 1.31 in the 17xx texts, 1.23 in the 18xx texts, and 1.33 in the 19xx texts.

The fact that the value is lowest for 18xx and not 19xx indicates that the texts from the 20^{th} century are in fact harder to tag for the tagger model than those from the 19^{th} century, which we discuss in the following.

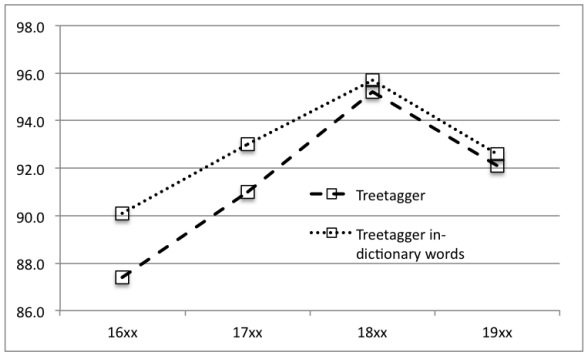

Figure 3: Influence of unknown words: tagging accuracy of the Tree-tagger on known words, and on all words

4 Discussion

4.1 Dropping Accuracy in the 20^{th} Century

One of the most surprising outcomes of our experiments was the fact that all taggers had lower accuracy on the 20^{th} century texts than on the 19^{th} century texts. One possible explanation is that this could be a random fluctuation caused by genre variation, for which we did not control. We extended our random sample and annnotated further 20^{th} century texts, but the performance did not change significantly. In future research, we will use an evaluation set that is stratified by genre. A second, more likely explanation is that some linguistic features of the 20^{th} century are harder to process. An important feature is the strong growth in vocabulary, for

Tree-Tagger Confusion	16xx	17xx	18xx	19xx	TOTAL
NN / NNP	8	14	1	6	29
VB / VBP	4	0	2	3	9
VBP / VB	3	1	1	3	8
VB / NN	1	0	4	1	6
VBD / VBN	2	1	1	2	6
JJ / VBN	0	0	3	2	5
JJ / NNP	0	2	0	3	5
VBD / NNP	0	2	0	3	5
RB / IN	3	1	0	1	5
NNS / NNP	1	4	0	0	5
RB / JJ	1	1	1	2	5
FW / NNS	4	0	0	0	4
NN / NNS	0	2	1	0	3
DT / NN	3	0	0	0	3
FW / NN	3	0	0	0	3
RB / NNP	1	0	0	2	3
VBG / NN	2	0	0	1	3
VBP / NN	2	0	0	1	3

Table 3: Most frequent tag confusions by the Tree-Tagger

CLAWS Tagger Confusion	16xx	17xx	18xx	19xx	TOTAL
NNP / NN	5	5	3	8	21
VB / VBP	8	2	4	2	16
DT / JJS	2	2	2	0	6
WP / WDT	2	0	1	2	5
CD / NN	4	0	0	0	4
DT / PRP	4	0	0	0	4
JJS / JJR	2	0	2	0	4
VBD / VBN	2	0	0	2	4
NN / VBP	1	2	1	0	4
WRB / IN	1	0	0	3	4
DT / JJS	0	2	2	0	4
NN / IN	0	2	0	2	4
FW / NN	3	0	0	0	3
FW / NNS	3	0	0	0	3
IN / RP	3	0	0	0	3
RB / IN	3	0	0	0	3
IN / RB	2	1	0	0	3
VBG / NN	2	0	0	1	3
JJ / NN	1	0	0	2	3
JJ / RB	1	2	0	0	3
NN / NNP	1	1	0	1	3
VBD / JJ	0	1	0	2	3
NNP / JJ	0	1	0	2	3

Table 4: Most frequent tag confusions by the CLAWS tagger

example the use of abbreviations. The Tree-tagger optionally marks out-of-vocabulary words. There are more out-of-vocabulary words in the 20^{th} century texts than in the 19^{th} century. Per century, the percentages of unknown words are: 5.2% in 16xx, 2.8% in 17xx, 2.2% in 18xx, and 3.0% in 19xx. While a higher amount of unknown words affects tagging accuracy, but also the accuracy of words that are known to the tagger decreases in 19xx, as Figure 3 shows. Out-of-vocabulary can thus only serve as a partial explanation.

We also noted that the 20^{th} century texts contain considerably more features which are particularly frequent in social media, for example telegram style and spoken features like contractions. Some do not contain apostrophes (e.g. *youre* instead of *you're*), which almost inevitably lead to tagging errors. Another feature are compressed and complex NPs. Two examples of sentences containing these features, and the relevant tags assigned by the CLAWS tagger are given in (1) and (2).

(1) Saturday 10 24 - A._NN Boiled_JJ sap_VBP this P.M. are having another good run of sap . (ARCHER 1920rich_y7a_s193)

(2) Specify Regal_JJ Mk V for 1960 Reliant_JJ 's Silver Jubilee year . (ARCHER 1960illn_a8b_s102)

4.2 Error Analysis

Error classes We have conducted an error analysis, to check which types of tagging error are particularly frequent, to find out if causes can be isolated, and if some tagging errors are more serious than others.

The most frequent types of confusion of the Tree-tagger, i.e. all errors that occur at least 3 times, are given in Table 3. The equivalent figures for the CLAWS tagger can be seen in Table 4. The most prominent cause of error is different capitalisation practice in previous periods. It needs to be pointed out that capitalisation is not normalised by VARD. An example of a sentence in which nouns are generally capitalised is given in (3).

(3) He had been very restless all Night, his Pulse irregular, his Tongue rough and dry, with Flushings in his Cheeks. (ARCHER 1735gool_m3b_s59)

While the Tree-Tagger tends to assign proper noun (*NNP*) to capitalized common nouns (*NN*) too often, the CLAWS tagger shows the opposite trend to overgeneralise *NN* to too many *NNP*s. An example is given in (4), the words in bold are incorrectly assigned common noun tags by CLAWS.

(4) Recently Whiting developed the **Bus** *and* **Car Washer** *, shown above , which shampoos a bus from end to end in only 45 seconds ...* (ARCHER 1942news_a7a_s132)

The second most frequent error is a confusion between infinite verb and inflected verb in the present. Due to the considerably freer word order in the earlier texts, material intervening between the auxiliary verb and the main verb frequently leads to situations in which the tagger's observation window is too small. An example which includes two errors of this type is given in (5), where the tagger assigned non-third person present tense (*VBP*) instead of nonfinite form (*VB*).

(5) ... whereas quite contrary they will without the least opposition **permit** *the first , but with the greatest difficulty* **admit** *of the last .* (ARCHER 1665head_f2b_s24)

The confusions involving the tag *FW* (foreign word) involve French and Latin expressions, which are more frequent in earlier texts. It is difficult to see further clear trends in the tagger confusion tables. The larger amount of unknown words in earlier texts and in the 20^{th} century typically leads to unspecific, context-dependent errors. Most of the remaining errors are too sparse in our small evaluation set to show clear trends or a sigificant decrease in PDE.

4.3 Underspecifiying Nouns

The most frequent tagger confusion, the one between common noun and proper name, is due to the fact that the distinction between common noun and proper name is particularly hard to make, because often the majortiy of nouns are capitalised in the earlier texts (see example 3), and it can also be argued that it is possibly inconsequential for the subsequent step of syntactic parsing.

We have therefore considered an evaluation variant in which the distinction between common noun (*NN(S)*) and proper name (*NNP(S)*) is not made.

Tagger Combination	16xx	17xx	18xx	19xx
Tree-Tagger	87.4	91.0	95.2	92.1
Tree-Tagger NN/NNP	89.0	93.8	95.4	93.6
CLAWS	87.8	93.2	95.0	92.8
CLAWS NN/NNP	88.8	94.4	95.4	94.5
Best Ensemble	88.8	93.4	95.6	94.1
Best Ensemble NN/NNP	89.6	94.4	96.0	95.8
Oracle	94.2	98.2	98.2	98.3
Oracle NN/NNP	94.4	98.4	98.4	98.7

Table 5: Accuracy of taggers if common noun (*NN*) and proper name (*NNP*) are not distinguished

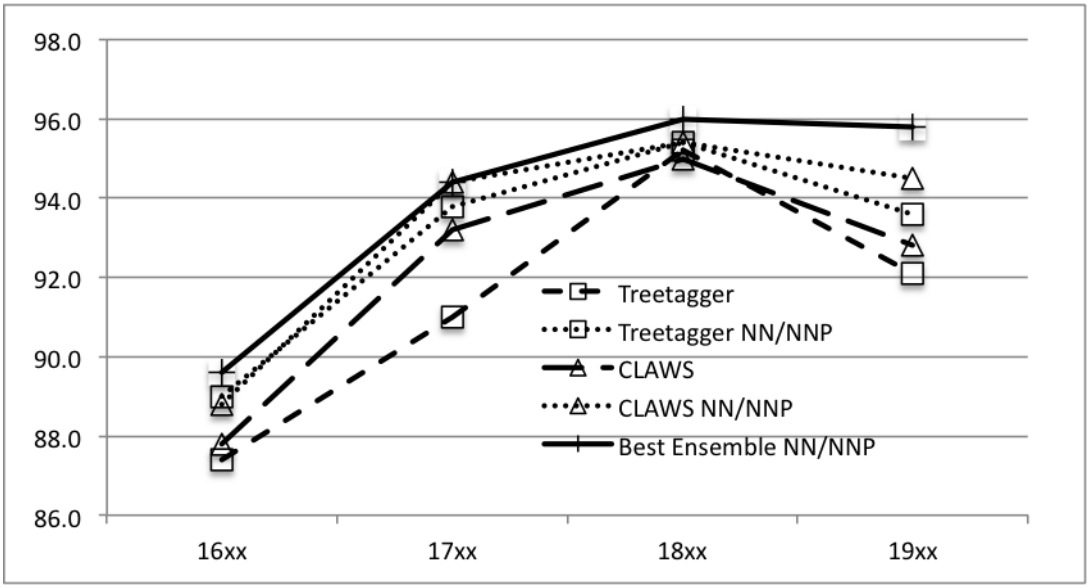

Figure 4: Proper vs. common nouns: Accuracy of Tree-Tagger, CLAWS tagger and Best Ensemble if common noun (*NN*) and proper name (*NNP*) are not distinguished

The accuracies are given in Table 5; Figure 4 contains a visualisation of the accuracy of the individual taggers, with and without the distinction, and the best ensemble, without the distinction between *NN* and *NNP*. We can also see that in this setting, where the tendency of CLAWS to overassign common noun tags to capitalised words is discounted, we reach the same level of accuracy for 19^{th} and 20^{th} century texts.

5 Related Approaches

First, we summarize approaches to present-day language data. Ensemble systems for POS tagging systems have been used by several authors. For example, van Halteren et al. (2001) use an ensemble system to tag two PDE English corpora, the Penn Treebank (Marcus et al., 1993) and LOB (Johansson, 1986). They combine four probilistic taggers with significantly different algorithms (HMM, memory-based, transformation rules, and maximum entropy), reporting that error rates could be reduced by 11% (Penn) to 24% (LOB). On Penn, the single best tagger reached 96.9% accuracy, the best combination increased to 97.2%. On LOB, the single best tagger reached 97.6% accuracy, the best combination increased to 98.1%. Loftsson (2008) combines a rule-based and two probabilistic systems for tagging Icelandic, a morphologically rich langauge in which data sparseness is particularly acute. The combined system, using a simple voting scheme, increases tagging accuracy by almost 1.5% over the best single tagger. In particluar, the improvement is much larger when including the rule-based tagger rather than using three probabilistic taggers, as the comparable approach of Helgadóttir (2004) did, which indicates that the different perspective which the rule-based tagger offers – like CLAWS has done in our approach – is

particularly beneficial.

For tagging historical data, we have mentioned in the Methods section that Rayson et al. (2007) also used the normalisation tool VARD, but a single tagger, they report that the normalised text leads to only about half as many tagger errors as the original text. In their experiments on Early Modern German texts, Scheible et al. (2011) measured improved tagging for 47% of the normalised words are tagged better, against a loss of correct results in 3% (and 50% which stay correct or incorrect). Schneider et al. (2014), again on English texts, report that on the subsequent level of syntactic parsing, 32% of the measured syntactic dependencies improve, 2% worsen, and 65% remain unaffected. Bollmann (2013) describes a similar approach using fully automatically normalised German data.

Approaches using domain adaptation exist for English, for example Yang and Eisenstein (2016). Kroch et al. (2004) train a tagger on the historical word forms directly, Dipper (2010) uses the same approach for Middle High German. These appraoches have the advantage that they reduce the risk of error accumulation, which is typical for pipeline systems, and the disadvantage that they are particularly susceptible to sparse data problems.

To our knowledge, there are only very few approaches using ensemble systems on historical data, which has motivated our current research.

6 Conclusions and Outlook

We have demonstrated that for the task of POS tagging of historical English, a careful mapping to PDE spelling with a normalisation tool such as VARD allows one to achieve almost PDE accuracy levels from about 1700 on. We have shown that automatically combining two taggers with sufficiently different approaches improves tagging performance by 0.78% on average. Levels stay slightly below state-of-the art results, as they assume perfect tokenisation, which is unrealistic for real-world texts.

Limited human intervention (choosing one of maximally three alternatives) improves tagging accuracy by an additional 2-5%, thus reaching above 98% on texts after about 1700. The hybrid (partly rule-based) CLAWS tagger performs considerably better on historical texts. It possibly profits from a more fine-grained tagset. Surprisingly, 19^{th} century texts can be easier to tag than PDE, which is due partly to more out-of-vocabulary words, partly to "social media" style, partly to complex nouns and abbreviations, and partly to the fact that CLAWS assign common noun tags to proper names too often.

In future research, we want to use more taggers, re-train taggers including more manually annotated historical texts, annotate a larger gold standard, and control for register variation. We are currently testing alternative spelling normalisation tools. We also want to test if the advantage of the CLAWS tagger can be related to its potential to profit from a more fine-grained tagset.

References

Alistair Baron and Paul Rayson. 2008. VARD 2: A tool for dealing with spelling variation in historical corpora. In *Proceedings of the Postgraduate Conference in Corpus Linguistics*, Birmingham. Aston University.

Douglas Biber, Edward Finegan, and Dwight Atkinson. 1994. Archer and its challenges: Compiling and exploring a representative corpus of historical english registers. In Udo Fries, Peter Schneider, and Gunnel Tottie, editors, *Creating and using English language corpora, Papers from the 14th International Conference on English Language Research on Computerized Corpora, Zurich 1993*, pages 1–13. Rodopi, Amsterdam.

Marcel Bollmann. 2013. POS tagging for historical texts with sparse training data. In *Proceedings of the 7th Linguistic Annotation Workshop and Interoperability in Discourse*, pages 11–18, Sofia, Bulgaria.

James Curran, Stephen Clark, and Johan Bos. 2007. Linguistically motivated large-scale nlp with c&c and boxer. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, pages 33–36, Prague, Czech Republic, June. Association for Computational Linguistics.

Thomas G. Dietterich. 1997. Machine learning research: Four current directions. *AI Magazine*, 18(4):97–136.

Stefanie Dipper. 2010. POS-tagging of historical language data: First experiments. In *Proceedings of KONVENS*.

Roger Garside and Nicholas Smith. 1997. A hybrid grammatical tagger: Claws4. In Roger Garside, Geoffrey Leech, and Tony McEnery, editors, *Corpus Annotation: Linguistic Information from Computer Text Corpora*, pages 102–121. Longman, London.

Claire Grover. 2008. LT-TTT2 example pipelines documentation. Technical report, Edinburgh Language Technology Group,.

Sigrún Helgadóttir. 2004. Testing data-driven learning algorithms for PoS tagging of icelandic. In Henrik Holmboe, editor, *Nordisk Sprogteknologi 2004*, pages 257–265, Copenhagen. Museum Tusculanums Forlag.

Stig Johansson. 1986. *The Tagged LOB Corpus: User's Manual*. Norwegian Computing Centre for the Humanities, Bergen, Norway.

Anthony Kroch, Beatrice Santorini, and Lauren Delfs. 2004. *Penn-Helsinki parsed corpus of Early Modern English*. Department of Linguistics, University of Pennsylvania, CD-ROM, first edition, release 3 edition.

Geoffrey Leech, Roger Garside, and Michael Bryant. 1994. Claws4: the tagging of the british national corpus. In *Proceedings of the 15th International Conference on Computational Linguistics (COLING 94)*, pages 622 – 628, Kyoto, Japan.

Hrafn Loftsson. 2008. Tagging icelandic text: A linguistic rule-based approach. *Nordic Journal of Linguistics*, 31(1).

Mitch Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1993. Building a large annotated corpus of English: the Penn Treebank. *Computational Linguistics*, 19:313–330.

Paul Rayson, Dawn Archer, Alistair Baron, Jonathan Culpeper, and Nicholas Smith. 2007. Tagging the bard: Evaluating the accuracy of a modern pos tagger on early modern english corpora. In *Proceedings of Corpus Linguistics 2007*. University of Birmingham, UK.

Silke Scheible, Richard J. Whitt, Martin Durrell, and Paul Bennett. 2011. Evaluating an 'off-the-shelf' pos-tagger on early modern german text. In *Proceedings of the ACL-HLT 2011 Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities (LaTeCH 2011)*, Portland, Oregon.

Helmut Schmid. 1994. Probabilistic part-of-speech tagging using decision trees. In *Proceedings of International Conference on New Methods in Language Processing*.

Gerold Schneider, Hans Martin Lehmann, and Peter Schneider. 2014. Parsing Early Modern English corpora. *Literary and Linguistic Computing*, first published online February 6, 2014 doi:10.1093/llc/fqu001.

Hans van Halteren, Walter Daelemans, and Jakub Zavrel. 2001. Improving accuracy in word class tagging through the combination of machine learning systems. *Computational Linguistics*, 27(2).

Shaoqun Wu. 2010. *Supporting Collocation Learning*. Ph.D. thesis, Department of Computer Science, University of Waikato, Hamilton, New Zealand.

Yi Yang and Jacob Eisenstein. 2016. Part-of-speech tagging for historical english. In *Proceeding of NAACL*.

Crosslinguistic Annotation of German and English Shell Noun Complexes

Fabian Simonjetz
Department of Linguistics
Ruhr-University Bochum
simonjetz@linguistics.rub.de

Adam Roussel
Department of Linguistics
Ruhr-University Bochum
roussel@linguistics.rub.de

Abstract

This contribution involves the manual annotation of shell nouns and their antecedents in a multilingual context. Shell nouns are abstract nouns, which like pronouns are semantically incomplete and derive their meanings from other parts of a text to which they refer, often anaphorically. Unlike pronouns, shell nouns also serve to characterize the content to which they refer. The annotation schema we introduce allows for the annotation of shell nouns along with their content and their translation in a parallel text. This approach should enable the production of data on shell nouns which encompasses various aspects of their behavior that have not yet been investigated in detail, including the use of multiple content phrases, nominalized content phrases, plural shell nouns or crosslinguistic behavior.

1 Introduction

Shell nouns are an open class of abstract nouns that refer to stretches of text, which complete the shell noun's semantic content and are simultaneously characterized by the shell nouns. Their name derives from the way they are said to *encapsulate* the content to which they refer (Schmid, 2000). Some typical examples are listed in (1). The shell nouns are printed in boldface and the content to which they refer is in italics.

(1) a. The **problem** was *that I had no money*. (Schmid, 2000)

b. *Pigs cannot fly.* That **fact** is well-known.

c. der **Plan** ist, *ein Auto zu kaufen* ('the plan is to buy a car')

Combining functions usually associated with pronouns (reference) and adjectives (characterization), shell nouns are a useful device for facilitating textual coherence. Yet they have received little attention so far, especially in languages other than English. The primary goal of this paper is to offer a set of annotated cross-linguistic data to serve as a basis for further exploration of shell nouns in English and German.

Previous studies on shell nouns have primarily focussed on the use of lexical patterns, such as in (2), for the discovery and analysis of shell nouns and their content phrases. Since previous work only looked at English shell nouns, this approach was more or less sufficient: though certain phenomena are systematically missed, it is thought that the bulk of relevant cases can be covered in this way, thanks to English's relatively fixed word order.

(2) Determiner + (Premodifier) + Noun + postnominal *that*-clause, *wh*-clause, or *to*-infinitive
The (deplorable) fact that I have no money (Schmid, 2000)

For this annotation task, we wanted to take advantage of the fact that we were conducting the annotation manually and annotate shell nouns in ways that are not amenable to automatic methods. Further, in order to gather data about the types of patterns in which shell nouns actually occur, we could not use Schmid's patterns for identifying shell noun instances. We also wanted to annotate in such a way so as to facilitate the crosslinguistic study of shell noun use. We were thus led to formulate three main criteria to guide our approach to shell nouns.

Incompleteness A shell noun (when used as such) is *incomplete* with regard to its semantic content. For example, a *fact* denotes a true state-of-affairs, whereas this same state-of-affairs might be cast as a *problem*, some undesirable situation

in want of a solution. An *aim* is something to be achieved in the future, a desirable situation, which has not yet come to pass. What exactly these various 'situations' or 'states-of-affairs' entail is only found in the co-text of the nouns, if it is made explicit at all. Unlike concrete nouns, shell nouns seem to possess a 'gap' or 'placeholder' for this additional information (Schmid, 2000, p. 79).

Reference A shell noun *refers* to linguistic content elsewhere in the discourse. This content could usually also occur without the shell noun itself, but the shell noun serves to describe or characterize this content and encapsulates it, allowing easier subsequent reference to it. Once a state-of-affairs has been summed up as a *problem*, a speaker can then go on to discuss this *problem* as they might do with some concrete entity. Reference can be achieved by a variety of means, for instance, with a copula verb linking the shell noun and its content (1a) or via anaphoric constructions (1b).

Abstractness The shell noun content must be *abstract*, in that it, for example, denotes entities which correspond semantically to the meanings of sentences, such as facts, states-of-affairs or propositions, i.e. *saturated* abstract objects or entities with truth values.[1]

2 Related Work

Schmid (2000) is the most extensive and detailed treatment of the topic of *shell nouns*, and it thus forms the basis of most later work on the topic. In this book, he addresses a whole range of aspects relating to shell nouns, including cognitive aspects, discourse functions, semantic categories, etc. This work is based on an extensive corpus-based study of shell noun instances. Shell noun instances are identified here primarily on the basis of lexical patterns, an approach which does not cover certain aspects, such as plurality and anaphoric shell noun complexes, but which is largely sufficient for English shell nouns, which are the sole focus of the book.

Shell nouns are in certain respects essentially a special case of *abstract anaphors*. Like abstract anaphors, they refer not to concrete entities, generally represented by NPs, but rather to propositions or proposition-like entities. The most obvious difference is that shell nouns are themselves full NPs as opposed to abstract anaphors in general, which are often pronouns, such as *this* or *it* in English and *dies* or *es* in German.

In contrast to abstract anaphors, shell nouns do not necessarily refer anaphorically to their content. Far more frequently, the content to which they refer is found in a *that*-phrase complement immediately adjoined to the shell noun. However, the similarity between the two constructions nevertheless means that annotation tasks relating to one involve techniques which are generally applicable to the other.

Dipper and Zinsmeister (2009) present guidelines and a pilot study for an annotation task similar to our own, though this task addresses abstract anaphors as opposed to shell nouns as such. Annotators were asked to identify antecedents of 48 instances of *dies* 'this' by freely marking spans of text. These guidelines introduce the 'paraphrase test',[2] for identifying anaphoric content phrases, which we also use in our guidelines. As the authors note, this appears also to have been the first study to approach abstract anaphors in German. Interestingly, the guidelines also recommend the use of shell nouns, such as *Ereignis* 'event', *Ansicht* 'view' or *Tatsache* 'fact', in order to identify the semantic type of abstract anaphors (the 'replacement test'). In a later study, Dipper and Zinsmeister (2012) expanded this approach, annotating 643 instances and investigating correlations between the abstractness of referents and antecedents. Dipper et al. (2011) use a cross-linguistic bootstrapping approach in order to expand the set of abstract anaphors under comparison and undertake an extensive contrastive study of their realization in German vs. English. All three of these studies also employ the Europarl Corpus, focusing on English and German parallel data.

Kolhatkar (together with Zinsmeister and Hirst) has approached the topic of shell nouns in a series of publications in an explicitly computational context, with the ultimate goal of resolving shell noun instances to their content automatically. The first of these, Kolhatkar and Hirst (2012), involves an annotation task quite similar to our own: annotators were asked to mark arbitrary spans of text corresponding to the content phrases of 183 instances of the shell noun *issue*. The authors then describe an automatic resolution algorithm developed using this data. Later work (Kolhatkar et al., 2013; Kolhatkar and Hirst, 2014) expanded the annotation to other shell nouns, increased the amount of data

[1] See Asher (1993) for more on the relevant typology.

[2] Later 'namely test' (Dipper and Zinsmeister, 2012).

annotated via crowdsourcing, and improved the resolution algorithm. Kolhatkar (2015) describes, in addition to these studies, extensive work concerning the annotation and automatic resolution of shell nouns in general. Relating as it does to these topics, the annotation guidelines which appear there are of direct relevance to the current study.

Simonjetz (2015) argues that Schmid's (2000) procedure of retrieving shell nouns may not be suitable for languages with a more flexible word order than English. It recommends the use of dependency-based syntactic patterns instead of simple string-based patterns in order to identify German shell nouns. With no evaluation data being available, Simonjetz (2015) relied on a manual examination of the results, making it impossible to reach a clear conclusion as to whether or not dependency patterns are superior to linear patterns for the task of identifying shell nouns in German. The resulting data on German shell nouns proved nonetheless useful for our study and has been the basis for the selection of shell noun candidates described in the next section.

3 The Annotation Process

Our project involves the annotation of parts of the Europarl Corpus (Koehn, 2005) with information pertaining to the usage of shell nouns in the text. The data of the Europarl Corpus is divided into plenary sessions, speaker turns and sentences. In order to achieve a degree of homogeneity in the data we filtered out long turns for our annotation project, which are likely to be recitations of written documents. On the other hand we also filtered out very short turns, which are unlikely to contain any shell noun occurrences and, if they do, their content is likely to be located in another turn, thus complicating the annotation. The thresholds for filtering were based on the distribution of text lengths in the corpus, in which three prominent bumps were visible, which we presumed to correspond to each of the types of text.

A complex annotation project such as this makes special demands on its annotators, requiring not only time and patience, but also special expertise. Often, in order to present a task to naive annotators, the task must be simplified and restricted, however this simplification requires certain assumptions to be made by researchers as to the nature of the phenomenon under investigation. Wanting to make as few such assumptions as possible and to encompass the whole spectrum of shell noun phenomena, we performed the annotations ourselves. Thus the data were annotated by two linguistically-informed annotators, one native speaker of English and one native speaker of German, both fluent in their respective non-native languages. This arrangement put us in a position to produce data that, though not without its own shortcomings, would have been prohibitively difficult to produce otherwise.

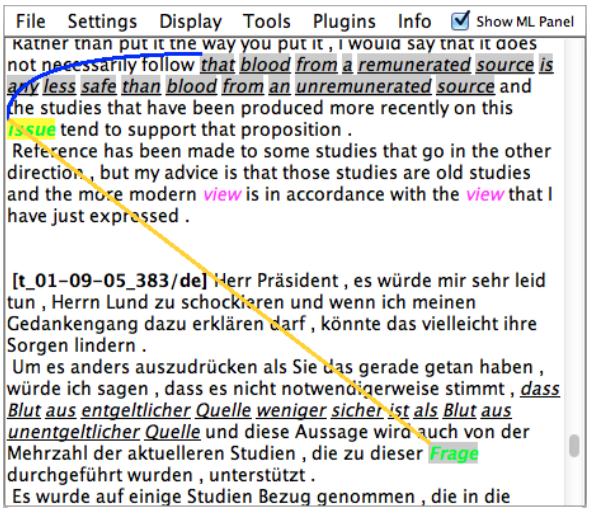

Figure 1: Screenshot of the annotation software as used in this study.

In order to assist the annotators in annotating as many instances as possible while keeping the data sets comparable across different annotators, shell noun candidates were determined beforehand and highlighted in the annotation software MMAX2 (Müller and Strube, 2006). Only highlighted instances were annotated to ease comparisons between annotators and alignment between languages. In case some shell noun candidate was highlighted in one language and its translation was not, the annotators added the translation to the annotation set.

The annotators first annotated instances of *Möglichkeit* and *possibility* using a preliminary version of the guidelines in Section 4. The data from this practice annotation are not included in the final data set. Afterwards the annotators convened and further developed the guidelines on the basis of this experience. We split the data to be annotated into three parts. For the first part, difficult cases were discussed. The last two parts were annotated completely independently, and accordingly only these two parts were used in the calculation of inter-annotator agreement measures.

Using the statistics from Schmid (2000) for English and Simonjetz (2015) for German, we chose 50 shell nouns for each language which have (a) a high ratio of shell noun vs. non–shell noun uses and (b) a high absolute frequency. Both of these factors are important since, if the nouns are too infrequent, then there may not be very much data to annotate (and it would be questionable how generalizable the data would be), and if the chosen nouns are used as shell nouns too infrequently, then it would be difficult to investigate the shell nouns' relationship to their content, since most of the instances would be negative in that case. (Note that these 50 German/English shell nouns are not necessarily translations of each other. Consequently, the shell nouns appearing in the final data slightly deviate from the original selection, as translations are added to the annotation set.)

Turns in both languages are presented to the annotators in parallel and three main levels are annotated:

- "Shell noun" – Annotators mark whether or not the given noun constitutes a usage as a shell noun. Options to mark unclear instances or instances whose content is just outside of the given turn are provided.
- "Content phrase" – Annotators may freely select spans of tokens which comprise the content to which a shell noun instance refers. This content may occur before or after its shell noun instance and may encompass multiple sentences. Each shell noun instance contains a pointer to its content phrase(s), of which there may be more than one. (Likewise, multiple shell nouns may point to the same content phrase.) Content phrases are also marked as being either 'nominal' or 'sentential'.
- "Alignments" – Annotators are then asked to associate corresponding shell noun instances in one language with their translations in the other language (insofar as a counterpart is present). The same is done for content phrases.

We hope that our manual annotation approach will mean that our data cover a greater variety of shell noun–related issues than would be possible with pattern-based methods, which could systemically miss particular unexpected properties and behaviors. For instance, patterns which do not allow for pluralized shell nouns will necessarily preclude any study of their properties as opposed to singular shell nouns, e.g. whether or not plurals tend to refer to multiple content phrases. Furthermore, pattern-based approaches are likely to be inadequate for languages, such as German, which have less strict word order. Therefore we see this project in part as an attempt to discover attributes that will be useful for studying shell nouns in languages other than English.

4 Guidelines

4.1 General features of shell nouns

Shell nouns may be identified by means of three criteria:

1. A shell noun has *incomplete* semantic content.
2. A shell noun *refers* to linguistic content elsewhere in the discourse. This content could usually also occur without the shell noun itself. The shell noun serves to describe or characterize this content.
3. The shell noun content must be *abstract*. It will generally denote entites such as facts, propositions or eventualities.

4.2 Determining shell noun content phrases

Mark the shortest possible, but complete, instance of the content to which the shell noun refers.[3] The syntactic type of the content (e.g. *that*-clause or infinitive clause) should be apparent when viewed in isolation. Content phrases should be complete constituents. Often content phrases can be deleted and the sentence will remain well-formed, as in (3). This is however not the case for all content phrase types, and annotators may need to employ other constituent tests to determine the appropriate boundaries of a content phrase.

(3) a. Die **Entscheidung**, *inwieweit die EZB die allgemeine Wirtschaftspolitik der Gemeinschaft unterstützt*, hängt also von deren Einschätzung einer möglichen Beeinträchtigung des Ziels der Preisstabilität ab. [t_98-04-01_154]

b. The **decision** *on how far the European Central Bank supports the general economic policy of the Community* thus depends on its assessment of a possible effect on the aim of price stability.

[3] After Kolhatkar (2015).

4.3 Nominalizations

Though the content phrase (owing to its propositional nature) generally has a verbal head (is either a VP or CP), there are some deverbal nouns which still take complements and thus possess a similar semantics to these verbal phrases (at least in German). These phrases can also act as the content phrase in a shell noun complex at least if they follow the shell noun in a postnominal prepositional phrase. The content of the shell noun in this example could be equivalently expressed either with a VP or an NP:

(4) a. Hier gibt es die **Möglichkeit** *zur Aktualisierung der Software*.
'Here there is the opportunity for the updating of the software.'

b. Hier gibt es die **Möglichkeit**, *die Software zu aktualisieren*.
'Here there is the opportunity to update the software.'

If such a paraphrase is not possible, as in (5), then it is unlikely that the given noun phrase's meaning is propositional and that the token in question constitutes a shell noun usage.

(5) a. Mein Antrag *zur Geschäftsordnung* lautet wie folgt: [...] [t_99-11-16_145]

b. The point of order is as follows: ...

Such cases may look similar to conventional coreferential nouns, but the syntactic behavior of nominal shell noun complexes differs from conventional nominal coreference. For instance, in constructions like (4a), coreference appears to be only possible if the involved nouns are a shell noun and (typically) a deverbal noun.

4.4 Anaphoric shell noun complexes

Dipper and Zinsmeister (2009) introduce the 'paraphrase test', which assists annotators in locating the content of anaphoric expressions – this test may also be applied to anaphoric shell noun complexes. Upon encountering an anaphoric expression, such as *this problem*, add a '*namely* clause' along with a paraphrase which best completes the *namely* clause. The content of this paraphrase (or the most similar formulation) should be marked in the text as the shell noun content.

(6) a. Dieser Artikel in seiner jetzigen Fassung würde nämlich verhindern, *daß in Fragen des dritten Pfeilers präjudizielle Beschwerde von den Gerichten eingelegt werden könnten*. Das wäre sehr gefährlich, denn damit würde man den Gerichten eine Möglichkeit nehmen; ließe man zumindest dem höchstinstanzlichen Gericht diese **Möglichkeit**, so wäre es eine Garantie für die Bürger, denn der Gerichtshof spielte dann eine wichtigere Rolle. [t_97-05-28_93]

b. It is precisely that which, in its current version, would prevent *any presentation by the Court of Justice of appeals which would prejudice matters affecting the third pillar*. That really would be very dangerous, because it would mean cutting off a possibility which those courts have and, if at least that **possibility** were left to the Supreme Court, that would be a guarantee for citizens and would give the Court of Justice a more important function.

The 'namely' paraphrase:

(7) a. [...] ließe man dem höchstinstanzlichen Gericht diese **Möglichkeit**, *nämlich daß in Fragen des dritten Pfeilers präjudizielle Beschwerde von den Gerichten eingelegt werden könnten*, so wäre es [...]

b. [...] if at least that **possibility**, namely *(some) presentation by the Court of Justice of appeals which would prejudice matters affecting the third pillar*, were left to [...]

4.5 Cataphoric shell noun complexes

The content of cataphoric shell noun complexes can generally be found in the same sentence as the shell noun, in a subordinated phrase. However, in some cases, the content is farther away and difficult to localize, such as is more often the case with anaphoric shell noun complexes.

(8) a. Mein **Antrag** zur Geschäftsordnung lautet wie folgt: Dies ist ein so wichtiges Thema, von dem Landwirte im gesamten Vereignigten Königreich betroffen sind, *daß uns wirklich mehr Zeit für Fragen an den Kommissar*

zur Verfügung stehen sollte. [t_99-11-16_145]

b. The **point of order** is as follows: this is such an important issue which affects British farmers across the UK *that we should surely have more time to question the Commissioner*.

To help with the localization of the content for such phrases, one might pose *clarification questions*. After a shell noun (such as *Antrag* or *request*) has been identified, one might pose the question, *was wurde beantragt?* or *what did the speaker request?*. Then select as the content phrase the text which most succinctly answers this question. There may be cases where multiple phrases seem to answer the question equally well. These phrases might even be literal restatements of the same content. In such cases, the annotator should choose the statement which is located closest to the shell noun.

4.6 Content phrase types

It is possible that the content phrase for a particular shell noun usage is not to be found in the present turn, either because the speaker has intentionally left this information implicit or because the shell noun refers to content located in some other turn. Further, it is possible that, for some shell nouns, it might be unclear whether the information is indeed located elsewhere in the text or intentionally omitted. The following choices are provided to annotators:

given	The shell noun content is present in the given text (and accordingly marked).
external	Wording implies that the speaker is referring to a specific linguistic entity, located nearby, though not in the current text.
unclear	It is unclear whether the noun is used as shell noun or not.

4.7 Multivalent shell nouns

Some shell nouns (most notably *reason*) can accept multiple content phrase complements, such as distinct causes and effects. In the case of *reason*, the shell noun content consists primarily of the 'cause' complement, since this is the content being described as the 'reason' for some other state of affairs.

Example of an *attempt*-class SN:

(9) a. *[Die Ausweitung das Emissionshandelssystems (ETS) der EU auf den Luftverkehr]*$_1$ ist vielleicht die beste **Möglichkeit**, *[um diese Emissionen zu begrenzen und um dafür zu sorgen, dass der Luftverkehr so wie alle anderen Sektoren einen Beitrag zur Senkung der schädlichen Treibhausgase leistet.]*$_2$ [t_06-07-04_136]

b. *[The extension of the EU Emissions Trading Scheme (ETS) to the aviation sector]*$_1$ may be the best **way** forward *[to limiting these emissions and to ensuring that aviation, like all other sectors, contributes to reducing harmful greenhouse gases]*$_2$.

The first clause contains the content of the shell noun, but the clause which would match most conventional patterns, the second one, contains what might better be construed as a goal or result of the content in the first clause, rather than the entity to which the shell noun actually refers. Clarification questions can be helpful to identify the correct referent in such cases (Kolhatkar, 2015):

(10) The primary *reason* that the archdiocese cannot pay teachers more is that its students cannot afford higher tuition.

(11) Q. What was the reason?

A. Because *its students cannot afford higher tuition*.

4.8 Coordination

Another instance in which a shell noun can accept multiple content phrase complements is that of coordination:

(12) a. Doch die **Feststellungen**, (1) *dass Europa kein neues Wissen schafft*, (2) *dass es nicht in der Lage ist, Wissen gemeinsam zu nutzen*, und (3) *dass es Europa nicht gelingt, Wissen finanziell zu fördern*, sollten uns doch sehr zu denken geben. [t_06-07-04_200]

b. However, the **statements** (1) *that Europe does not seek to acquire new knowledge*, (2) *that it cannot share knowledge* and (3) *that it does not support knowledge financially* all have a very ominous ring to them.

Each of these individual content phrases could stand alone. Hence, they should be regarded as separate content phrases of the same type and annotated accordingly.

(13) a. Es kam die **Frage** auf *wann wir diesen Punkt besprechen und endlich abschließen*.
b. The **question** *of when we would discuss this issue and be finished* was posed.

Here the NP *diesen Punkt* complements both *besprechen* and *abschließen*, which means that here two content phrases are not being coordinated, rather two verbs (or subordinated VPs). In this case, *endlich abschließen* could not stand alone and is dependent on the rest of the phrase, therefore this is an example of just one content phrase.

Contents may also be described with multiple shell nouns, which does not result in any particular consequences for our annotation schema, i.e. multiple references are possible in both ways – a single shell noun instance can point to a number of different content phrases, while one and the same content phrase can be pointed to by multiple shell nouns:

(14) a. Es ist unser **Wunsch** und unsere **Absicht**, *ein [. . .] Wahlsystem [. . .] einzuführen*. [t_97-06-11_76]
b. It is our **wish** and **intention** *to introduce a new electoral system . . .*

In example (14) both shell nouns, *wish* and *intention*, should be annotated with a pointer to the same content phrase entity, marked here in italics.

4.9 Punctuation

Pairwise punctuation (such as quotation marks or parentheses) should be included in a shell noun phrase when one of the elements occurs within a content phrase. Other punctuation should be treated like whitespace in sentence-internal content phrases, i.e., when it occurs within the phrase it is included, but at the beginning or end, it is ignored. Punctuation at the beginning or end of sentences, however, is regarded to belong to the sentence and is thus included, which appears to be more natural than excluding them.

4.10 Alignment

Both shell noun instances and their associated content phrases should be manually aligned cross-linguistically. In many cases it is rather straightforward what elements are to be aligned, but if expressions are not formulated analogously across languages it might be difficult to decide what elements belong together. Furthermore, elements occurring in one language do not necessarily correspond to a linguistic item in the other language, i.e. there might be occurrences of shell nouns or content phrases without any alignment:

(15) a. I would also ask that the Commission take note of the **fact** *that the European people would welcome Mr Mobutu as much as they would the greatest criminal*. [t_97-05-28_22]
b. Ich fordere die Kommission auch auf, zur Kenntnis zunehmen, *daß die Bürger Europas Herrn Mobutu ebenso freundlich wie den größten Kriminellen begrüßen würden*.

Shell noun phrases can be referred to by a number of lexical items that do not belong to the class of shell nouns, e.g. pronouns or – in in the case of German – pronominal adverbs (such as *deshalb*, *daher*). Such entities should be marked as negative shell noun instances, but only if their counterpart in the other language is an actual shell noun.

5 Discussion

In total, about 2140 potential shell noun instances were annotated by both annotators. (The first third of these served as practice data, such that only two thirds of these instances are reflected in the statistics for inter-annotator agreement.) Of these, a little less than half were marked by both annotators as positive instances. Subsequently, since content phrases can only be marked along with actual shell nouns, there are approximately half as many content phrase annotations in the data. Figure 2 provides an overview of the relative frequencies of positive and negative shell noun instances.

In general, shell nouns appear to have been used more frequently in English in our data. This is likely due, at least in part, to the tendency of certain predicates in English to only accept NP complements (as opposed to sentential complements). For example, *to take note of X* requires *X* to be an NP, and this NP often takes the form of a shell noun complex, such as *the fact that [. . .]*. This stands in contrast to a number of German expressions which follow the pattern *zu Kenntnis nehmen, dass . . .*

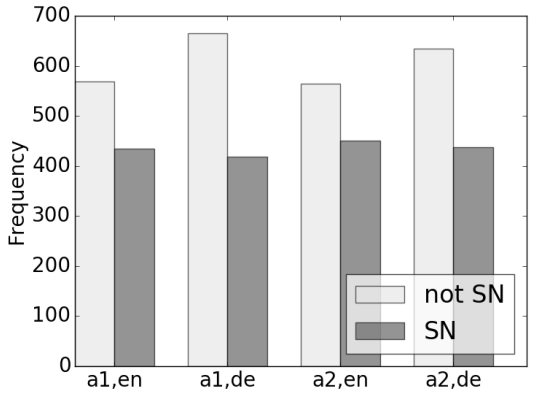

Figure 2: Comparison of shell noun vs. non–shell noun instances. ("a1" = Annotator 1, "a2" = Annotator 2)

	undef	false	true	unclear
undef	26	8	6	0
false	2	681	90	14
true	1	60	433	4
unclear	0	1	3	0

Figure 3: Confusion matrix for the two annotators ("undef" = Unannotated instance, "false" = Not a shell noun, "true" = shell noun, "unclear" = Not clear whether instance is shell noun or not).

(roughly, 'to take note that . . . '); here *zu Kenntnis nehmen* accepts a CP directly.

For the shell noun annotation level, in which annotators must mark a shell noun candidate as being an actual shell noun instance or not, raw agreement between annotators was 86%. Inter-annotator agreement, calculated according to Scott's π and Cohen's κ (Artstein and Poesio, 2008) both provide values of 0.73 for both languages taken together (though these values were minimally lower for English alone, 0.72). (Figure 3 provides more detail.)

Where a shell noun was marked as a positive instance, annotators are also asked to locate its content phrase as a span of text. When we take these spans to be sequences of token IDs, then each positive shell noun instance can be associated with a set of such spans. When comparing the sets corresponding to overlapping shell noun instances directly, approximately 65% of such sets were marked identically by both annotators. This number is comparable to Dipper and Zinsmeister's (2012, p. 47) observed agreement on exact matches, for which they report a value of 40%.[4] If, however, we require only that each annotated span overlap with some span from the other annotator, then 96% of the annotated content phrase spans could be considered matches (compared to 84% in the above-mentioned study).

Since the annotators could mark multiple, potentially discontinuous sequences of tokens for this task, determining annotator agreement is a nontrivial problem. We decided to use Krippendorff's α (Krippendorff, 2011), which was used by Kolhatkar and Hirst (2012) for a similar annotation task. This not only means that we were able to use an agreement measure appropriate to our data, but also that our values will be comparable to those resulting from a similar annotation task. We obtained a value of $\alpha = 0.84$, which is a relatively good value (by Krippendorff's standard) and a plausible one too, since it is only slightly worse than the reported agreement in Kolhatkar and Hirst's study ($\alpha = 0.86$, p. 1258).

We also analysed the distances between shell nouns and their content, for instance, in order to determine whether anaphoric shell noun complexes (in which the content precedes the shell noun instances) might be more frequent in one language or the other. In fact, as Figures 4 and 5 show, there do appear to be differences between English and German in this regard. Namely, German content phrases appear to occur more frequently at a greater distance to their shell noun, whereas English content phrases follow in the vast majority of cases the shell noun directly. The two-sample KS-test[5] confirms that the difference between these two distributions is statistically significant ($p = 1.28 \times 10^{28}$). The distribution of English content phrases shows that Schmid's (2000) pattern-based approach was appropriate for English, in that it is likely to have covered most of the data. However, our data for German show that such an approach is unlikely to suffice for the study of shell nouns in other languages.

Noting that content phrases are often headed by deverbal nouns (see Section 4.3 above), we also annotated the data with information regarding the syntactic status of the content phrases, i.e., whether they were nominal or not. There appear to be in-

[4]NB: Though that study involved the annotation of the antecedents of abstract anaphors, annotating the content phrases of shell nouns is, in many ways, the same task.

[5]The two-sample KS-test tests the null hypothesis that two independent samples come from the same distribution.

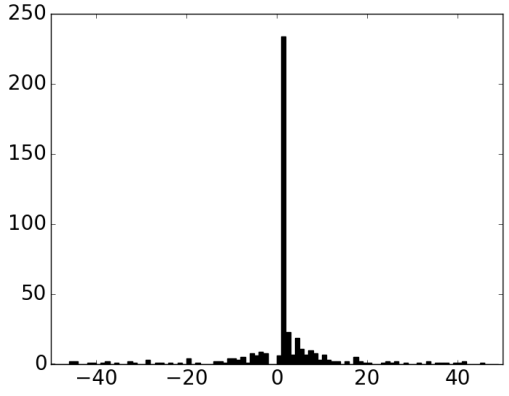

Figure 4: Distance between shell nouns and their content in tokens (English).

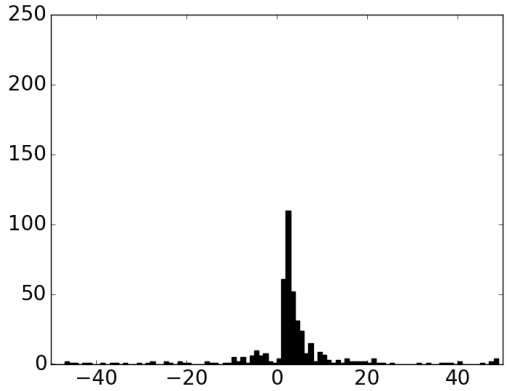

Figure 5: Distance between shell nouns and their content in tokens (German).

teresting cross-linguistic differences in this regard as well, for instance that nominal content phrases are more common in German, which could be of interest in future studies (cf. Figure 6).

6 Outlook

The data which was produced in this study and which can be produced using our annotation schema allow for the investigation of a number of questions which would be difficult to approach otherwise, such as those concerning the relative usage of shell nouns *in general* as well as the relative usage of *particular* shell nouns in German and English.

These data could furthermore serve as training data for clustering algorithms or other machine learning algorithms for categorizing content phrases or categorizing shell nouns based on the content phrases which they prefer. Such a typology of shell nouns, apart from its theoretical value, could aid in the automatic resolution of shell nouns and their content phrases.

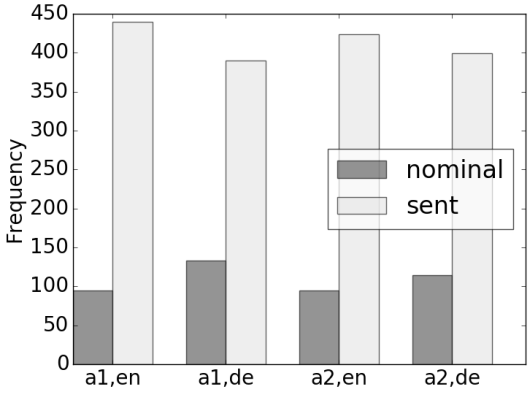

Figure 6: Shell noun content types. ("a1" = Annotator 1, "a2" = Annotator 2)

The annotated data can be found at: `https://github.com/ajroussel/shell-nouns-data`.

Acknowledgments

Many thanks to our anonymous reviewers for their helpful comments and criticism!

References

Ron Artstein and Massimo Poesio. 2008. Inter-Coder Agreement for Computational Linguistics. *Computational Linguistics*, 34(4):555–596.

Nicholas Asher. 1993. *Reference to Abstract Objects in Discourse*. Kluwer, Dordrecht.

Stefanie Dipper and Heike Zinsmeister. 2009. Annotating discourse anaphora. *Proceedings of the Third Linguistic Annotation Workshop, ACL-IJCNLP*, (August):166–169.

Stefanie Dipper and Heike Zinsmeister. 2012. Annotating abstract anaphora. *Language Resources and Evaluation*, 46(1):37–52.

Stefanie Dipper, Christine Rieger, Melanie Seiss, and Heike Zinsmeister. 2011. Abstract anaphors in German and English. In *Lecture Notes in Computer Science*, pages 96–107. Springer.

Philipp Koehn. 2005. Europarl: A parallel corpus for statistical machine translation. In *MT Summit X*, volume 5, pages 79–86.

Varada Kolhatkar and Graeme Hirst. 2012. Resolving "this-issue" anaphora. In *Proceedings of the 2012 Conference on Empirical Methods in Natural Language Processing*, pages 1255–1265, Jeju Island, Korea.

Varada Kolhatkar and Graeme Hirst. 2014. Resolving shell nouns. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 499–510, Doha, Qatar.

Varada Kolhatkar, Heike Zinsmeister, and Graeme Hirst. 2013. Annotating anaphoric shell nouns with their antecedents. In *Proceedings of the 7th Linguistic Annotation Workshop & Interoperability with Discourse*, pages 112–121, Sofia, Bulgaria.

Varada Kolhatkar. 2015. *Resolving shell nouns*. Ph.D. thesis, University of Toronto.

Klaus Krippendorff. 2011. Computing Krippendorff's Alpha-Reliability. Retrieved from `http://repository.upenn.edu/asc_papers/43`.

Christoph Müller and Michael Strube. 2006. Multi-level annotation of linguistic data with MMAX2. In Sabine Braun, Kurt Kohn, and Joybrato Mukherjee, editors, *Corpus Technology and Language Pedagogy: New Resources, New Tools, New Methods*, pages 197–214. Peter Lang, Frankfurt a.M., Germany.

Hans-Jörg Schmid. 2000. *English Abstract Nouns as Conceptual Shells: From Corpus to Cognition*. de Gruyter, Berlin.

Fabian Simonjetz. 2015. Retrieving German shell nouns using dependency patterns. `http://www.researchgate.net/publication/306020586_Retrieving_German_Shell_Nouns_Using_Dependency_Patterns`.

A Sample of Annotations

Most common shell nouns in German and then in English. Cases in which the ratio of shell noun usages to non–shell noun usages was 1.0 have been filtered out; generally, whenever this ratio was 1.0, the shell noun only occurred once and was only added to the test set for the sake of alignment with a shell noun instance in the other language. Cases are also not listed here if the total number of shell noun usages was less than 2. Shell nouns which were not in the original list of 50 shell noun candidates (which were pre-marked for annotation) are listed here in **boldface**.

Legend:

Undef.	Number of instances left unannotated.
False	Number of instances *not* functioning as a shell noun.
True	Number of instances functioning as a shell noun.
Unclear	Annotator was unable to determine whether or not instance constitutes use as a shell noun.
%SN	Ratio of shell noun vs. non–shell noun instances.

A.1 Annotator 1

Lemma	Undef.	False	True	Unclear	%SN
Frage	5	84	73	0	0.450617
Möglichkeit	0	11	30	0	0.731707
Ziel	2	17	22	0	0.536585
Forderung	0	7	19	0	0.730769
Tatsache	2	0	19	0	0.904762
Vorschlag	1	69	19	0	0.213483
Auffassung	0	6	17	0	0.739130
Ansicht	2	23	17	0	0.404762
Recht	6	59	17	0	0.207317
Grund	0	7	14	1	0.636364
Meinung	0	18	12	0	0.400000
Entscheidung	0	41	9	0	0.180000
Plan	0	5	7	0	0.583333
Gelegenheit	1	5	7	0	0.538462
Gefahr	0	6	7	0	0.538462
Aufgabe	0	8	7	0	0.466667
Verpflichtung	0	4	7	0	0.636364
Antrag	0	17	6	0	0.260870
Überzeugung	0	2	6	0	0.750000
Versuch	0	1	6	0	0.857143
Voraussetzung	0	9	6	0	0.400000
Hoffnung	0	4	5	0	0.555556
Schlussfolgerung	0	5	4	0	0.444444
Pflicht	0	4	4	0	0.500000
Bereitschaft	0	1	3	0	0.750000
Hinweis	1	6	3	0	0.300000
Absicht	0	1	3	0	0.750000
Wunsch	0	3	3	0	0.500000
Argument	0	3	2	0	0.400000
Standpunkt	1	17	2	0	0.100000
Lage	3	31	2	0	0.055556
Argumentation	0	1	2	0	0.666667
Zielsetzung	0	3	2	0	0.400000
Wille	0	6	2	0	0.250000

Lemma	Undef.	False	True	Unclear	%SN
fact	6	18	47	0	0.661972
question	0	29	46	0	0.613333
reason	1	13	30	1	0.666667
need	4	19	25	0	0.520833
opportunity	0	18	18	0	0.500000
right	0	23	17	0	0.425000
proposal	2	69	17	0	0.193182
issue	0	19	16	0	0.457143
aim	1	0	14	0	0.933333
decision	1	41	10	0	0.192308
objective	0	23	10	0	0.303030
view	0	16	8	0	0.333333
plan	0	4	7	0	0.636364
possibility	0	6	7	0	0.538462
idea	1	9	6	0	0.375000
hope	1	3	6	0	0.600000
effort	0	13	6	1	0.300000
conclusion	2	19	6	0	0.222222
requirement	0	8	5	0	0.384615
risk	0	2	5	0	0.714286
opinion	0	9	5	0	0.357143
intention	0	1	5	0	0.833333
argument	0	7	5	0	0.416667
demand	0	3	5	0	0.625000
duty	0	4	4	0	0.500000
commitment	0	3	4	0	0.571429
point	0	2	4	0	0.666667
problem	0	1	3	0	0.750000
suggestion	0	2	3	0	0.600000
attempt	0	1	3	0	0.750000
indication	0	1	3	0	0.750000
matter	0	13	2	0	0.133333
occasion	0	1	2	0	0.666667
courage	0	2	2	0	0.500000
promise	0	1	2	0	0.666667
danger	0	4	2	0	0.333333
request	0	1	2	0	0.666667
wish	0	4	2	0	0.333333
option	0	2	2	0	0.500000
doubt	0	10	2	0	0.166667

A.2 Annotator 2

Lemma	Undef.	False	True	Unclear	%SN
Frage	0	71	85	6	0.524691
Vorschlag	1	55	32	1	0.359551
Möglichkeit	0	11	30	0	0.731707
Ziel	2	15	22	2	0.536585
Tatsache	2	0	19	0	0.904762
Forderung	0	8	18	0	0.692308
Ansicht	2	23	17	0	0.404762

Auffassung	1	6	16	0	0.695652
Grund	0	6	15	1	0.681818
Meinung	0	18	12	0	0.400000
Entscheidung	0	37	12	1	0.240000
Recht	3	61	11	0	0.146667
Voraussetzung	0	7	8	0	0.533333
Plan	0	4	8	0	0.666667
Aufgabe	0	8	7	0	0.466667
Gelegenheit	1	6	6	0	0.461538
Überzeugung	0	2	6	0	0.750000
Notwendigkeit	0	1	6	0	0.857143
Lage	4	27	5	0	0.138889
Verpflichtung	0	6	5	0	0.454545
Hoffnung	0	5	4	0	0.444444
Antrag	0	19	4	0	0.173913
Gefahr	0	9	4	0	0.307692
Schlussfolgerung	0	6	4	0	0.400000
Hinweis	1	6	3	0	0.300000
Pflicht	0	5	3	0	0.375000
Absicht	0	1	3	0	0.750000
Wille	0	6	2	0	0.250000
Standpunkt	1	17	2	0	0.100000
Argument	0	2	2	1	0.400000
Wunsch	0	4	2	0	0.333333

Lemma	Undef.	False	True	Unclear	%SN
question	0	21	53	2	0.697368
fact	3	21	47	0	0.661972
reason	0	12	31	1	0.704545
need	1	20	27	0	0.562500
proposal	1	57	26	1	0.305882
opportunity	0	11	20	0	0.645161
issue	0	19	12	2	0.363636
decision	1	37	12	1	0.235294
objective	0	18	12	2	0.375000
aim	1	2	12	0	0.800000
right	0	30	11	0	0.268293
possibility	0	3	10	0	0.769231
plan	0	4	8	0	0.666667
call	0	1	7	0	0.875000
view	0	17	7	0	0.291667
argument	0	3	7	1	0.636364
idea	1	9	6	0	0.375000
effort	0	13	6	1	0.300000
conclusion	1	20	6	0	0.222222
position	0	14	5	0	0.263158
opinion	0	10	5	0	0.333333
demand	0	3	5	0	0.625000
point	0	2	5	0	0.714286
hope	0	5	4	0	0.444444
commitment	0	4	4	0	0.500000

desire	0	1	3	0	0.750000
failure	1	1	3	0	0.600000
duty	0	5	3	0	0.375000
indication	0	1	3	0	0.750000
requirement	0	7	3	0	0.300000
promise	0	1	2	0	0.666667
option	0	2	2	0	0.500000
danger	0	4	2	0	0.333333
wish	1	4	2	0	0.285714
condition	0	2	2	0	0.500000
matter	0	12	2	0	0.142857
request	0	2	2	0	0.500000
courage	0	2	2	0	0.500000

Rule-based Automatic Text Simplification for German

Julia Suter **Sarah Ebling** **Martin Volk**
Institute of Computational Linguistics, University of Zurich
Andreasstrasse 15, 8050 Zurich, Switzerland
suter@cl.uni-heidelberg.de, {ebling|volk}@cl.uzh.ch

Abstract

Automatic text simplification is capable of rendering texts comprehensible and accessible to persons with difficulties in reading and processing written language. In this paper, we report on the development of a rule-based automatic text simplification system for German. We show that the complexity of the output of our system is comparable to that of simplifications produced by a human.

1 Introduction

Simplified language aims to make texts comprehensible and accessible to persons with difficulties in reading and processing written language.[1] It is aimed not just at cognitively impaired persons but also at functionally illiterate and deaf persons, persons suffering from dementia and other neurodegenerative diseases, and immigrants.

Simplified language is characterized by reduced lexical and syntactic complexity, the addition of explanations for difficult words, and a clearly structured layout. Text in simplified language is usually obtained by simplifying a text written in standard language. By definition, simplification should not alter the meaning and informative value of a standard-language text (Coster and Kauchak, 2011a); this is what distinguishes it from other text-to-text generation tasks such as text compression.

Automatic text simplification, the process of automatically producing a simplified text, has only recently become an established research topic. It offers the potential of both increasing readability and comprehensibility for humans and improving processability for machines. As an example of the latter, text simplification as a preprocessing step can increase performance of natural language processing tasks such as parsing, machine translation, information retrieval, and text summarization (Chandrasekar et al., 1996).

Automatic text simplification systems have been developed for languages such as English, Swedish, and Portuguese. While tools exist for *detecting* complex structures in German texts,[2] to the best of our knowledge, no system exists for automatically *simplifying* these structures. On a more general level, Matausch and Nietzio (2012) state that "plain language is still underrepresented in the German speaking area and needs further development".

In this paper, we report on the development of a rule-based automatic text simplification system for German. Our approach builds on simplification rules extracted from guidelines for simplified German. We show that the complexity of the output of our system is comparable to that of simplifications produced by a human.

The remainder of this paper is structured as follows: Section 2 discusses the guidelines we used as a basis for our German simplification system (Section 2.1) as well as previous approaches to automatic text simplification for languages other than German (Section 2.2). Section 3 introduces our simplification system, discussing the resources used (Section 3.1) and simplification method applied (Section 3.2) as well as presenting an evaluation (Section 3.3) and discussing the results thereof (Section 3.4).

2 Simplified German

2.1 Guidelines

Guidelines specifying the character set, vocabulary, linguistic structures, and layout permitted for

[1] Related terms are *plain language*, *simple language*, or *easy-to-read language*. The term *simplified language* is used throughout this paper to emphasize the fact that the underlying concept is by no means standardized, as will become obvious in Section 2.1.

[2] An example is the LanguageTool (https://www.languagetool.org/de/leichte-sprache/).

simplified language are essential for systematically simplifying a standard-language text. For simplified German, four well-known guidelines exist: the guidelines by Inclusion Europe (2009),Netzwerk Leichte Sprache (2009), the BITV 2.0 rule set (Bundesministerium der Justiz und für Verbraucherschutz, 2011), and the guidelines by Maaß (2015).

Simplified language is still a young phenomenon, with profound research on the concept, its target groups, and guidelines still ongoing. No standardized version of simplified German exists. Accordingly, the four guidelines introduced above do not always agree on the best way to simplify complex language. The guidelines by Maaß (2015) provide the most coherent, linguistically motivated recommendations for transforming standard German into simplified German. We therefore based our work on these guidelines, well aware that some of our simplification rules might need adjustment at a later stage as more research is carried out in the respective area.

The guidelines by Maaß (2015) are divided into five categories according to the level of language they concern: character level, word level, sentence level, text level, and layout.

2.1.1 Character level

The character set of simplified German contains the letters of the German alphabet, an extension of the Latin alphabet with umlaut vowels (ä, ö, ü). In addition, digits and the special characters . ? ! „ “ : · are permitted. Other special characters such as the paragraph symbol (§) or dollar sign ($) are not allowed. The comma is not part of the inventory of simplified German according to Maaß (2015), as subordinate clauses and enumerations, which are typically introduced by or contain commas, should not be used. Numbers should be written as digits rather than words, with the exception of the indefinite article *ein* (‘a’), which is to be written as a word to prevent ambiguity with the cardinal number 1. Since compounds are productive in German, Maaß (2015) proposes the use of a typographical device called *Mediopunkt* (‘center dot’) to visually segment compounds, e.g., *Unfall·versicherung* (‘accident insurance’). Other guidelines suggest using hyphens in compounds; however, this requires capitalization of the compound segments and can lead to non-standard spelling.

2.1.2 Word level

Simplified language may contain only simple, short, and well-known words. Technical terms, foreign words, and abbreviations should be avoided, though common acronyms such as *CD* may be used. In cases where a difficult word is unavoidable, the word should be explained in simple terms. So far, no vocabulary list for simplified German exists.

2.1.3 Sentence level

Each sentence in simplified language should only contain one piece of information. Therefore, coordinate and subordinate clauses should be transformed into independent main clauses. Main clauses should preferably contain subject-verb-object (SVO) word order, active voice, and present or past perfect tense. Negations, nominal style, and metaphors should be avoided. Rare morphological forms may be unknown to inexperienced readers, so genitive case, subjunctive mood, and past simple tenses need to be eliminated.

2.1.4 Text level

In simplified language, consistency is given preference over style: word repetition and linear syntactic structures are encouraged, even though this conflicts with stylistic conventions common in standard language. Synonyms and third-person pronouns should be replaced with their antecedent noun phrases. Indirect speech is to be rephrased as direct speech. Additionally, a text may be enhanced with examples and explanations. Pictures, charts, and graphics should only be used if they are meaningful and appropriate for the target readership.

2.1.5 Typography and layout

Simplified German is always displayed one sentence per line. If a sentence takes up more than one line, it should be segmented at syntactic phrase boundaries. Text should be set in a large sans-serif font type and structure emphasized by means of headlines and indentations.

2.2 Automatic text simplification

Automatic text simplification can be performed using rule-based or corpus-based (mostly statistical) approaches. Rule-based automatic text simplification systems have been developed, e.g., for English, Swedish, French, Spanish, and Portuguese. These systems perform, among other tasks, lexical simplification (Kandula et al., 2010; Paetzold and

Specia, 2015), explanation generation (Watanabe et al., 2010), and syntactic simplification such as splitting long coordinate and subordinate phrases, rephrasing appositives and relative clauses (Aluísio and Gasperin, 2010), resolving third-person pronouns (Siddharthan, 2006), changing passive to active voice, and rearranging irregular word order (Rennes and Jönsson, 2015).

Corpus-based approaches have taken, e.g., the form of simplification via statistical machine translation (SMT) in the past (Coster and Kauchak, 2011a; Coster and Kauchak, 2011b; Specia, 2010; Stymne et al., 2013). Klaper et al. (2013) created a parallel German/Simple German corpus containing 70,000 tokens for use in SMT. However, they did not train an SMT system.

3 Rule-based simplification for German

We decided to follow a rule-based approach to text simplification for a number of reasons. Most importantly, statistical approaches require large amounts of data, something that is not available for simplified German to date. The parallel corpus by Klaper et al. (2013) mentioned in Section 2.2 cannot be expected to be sufficiently large to train an SMT system that works reasonably well. Secondly, if text simplification is used as an assistive technology, it is essential that it produces accurate results. Meaningless paraphrasings produced by an SMT system or other statistical methods can be even more confusing than the original (non-simplified) text (Shardlow, 2014). Finally, the guidelines by Maaß (2015) suggest simplification steps that can only be achieved through syntactic transformation rules. Statistical approaches are not well equipped to handle simplifications that require syntactic reordering, morphological transformations, and insertions due to lack of explicit linguistic knowledge (Siddharthan, 2014).

3.1 Resources

Our system makes use of a number of external resources for German. For example, it is based on the output of syntactic parsing of the source text. We employ the hybrid dependency parser ParZu (Sennrich et al., 2009), which performs sentence segmentation and tokenization. For compound segmentation, we use the tool Gertwol, which returns all possible segmentations of a word and provides further morphological analysis (Haapalainen and Majorin, 1995). For selecting the best segmentation, we implemented the algorithm suggested by Volk (1999), which ranks compound candidates according to their internal complexity of composition and derivation boundaries.

To retrieve abbreviations and their corresponding full forms, we extracted a list of 405 abbreviations and 278 acronyms from Wikipedia.[3] For verb conjugation, we rely on a web service that provides conjugation tables for most German verbs in all tenses and modes.[4] Nominals are inflected using CanooNet, an online dictionary that contains more than 250,000 manually checked German word entries.[5] Short definitions of difficult words are extracted from Hurraki, a Wiki-style encyclopaedia for simplified German consisting of more than 2,400 articles.[6]

3.2 Method

We implemented a subset of the rules described in Section 2.1 to perform automated simplification on the character, word, sentence, and text level and to adjust the layout of the output. The architecture of our system is shown in Figure 1. Following preprocessing and syntactic parsing of the source text (cf. Section 3.1), the simplification rules are applied sentence by sentence.

3.2.1 Character- and word-level rules

Prior to the parsing step, parentheses and their enclosed contents are removed from the source text, and abbreviations are expanded to their full forms. Both steps simplify the text and improve parsing performance. Following the parsing step, numbers written as words and special characters are replaced by digits and appropriate word substitutions using manually created dictionaries. All nouns longer than five characters that are not proper names are examined as to whether they are compounds; if this is the case, they are split using the *Mediopunkt* (cf. Section 2.1.1). Sample character- and word-level transformations are shown in Example 1.

(1) **German**
Prof. Müller kauft sich den siebten Band (den letzten) seiner Lieblingskrimireihe für 8$ 50¢ inkl. MwSt.

[3] https://de.wikipedia.org/wiki/Portal:Abkürzungen/Gebräuchliche_Abkürzungen
[4] http://www.verbformen.de/
[5] http://www.canoo.net/
[6] http://hurraki.de/

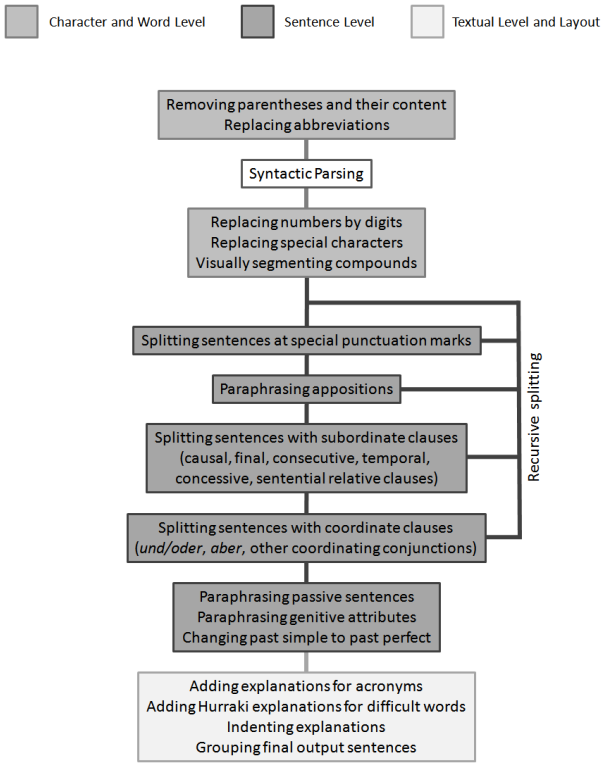

Figure 1: Architecture of rule-based text simplification system.

'Prof. Müller buys the seventh volume (the last one) of his favorite crime novel series for 8$ 50¢, incl. VAT.'

Simplified German
Professor Müller kauft sich den 7. Band von seiner Lieblings·krimi·reihe für 8 Dollar 50 Cent inklusive Mehrwert·steuer.

'Professor Müller buys the 7th volume of his favorite crime novel series for 8 dollars 50 cents, including value-added tax.'[7]

3.2.2 Sentence-level rules

On the sentence level, a series of syntactic simplification rules are executed. These rules split and/or rephrase the sentences. Syntactic simplification is applied recursively. The individual simplification rules are either executed once per iteration or are triggered by "keywords" (words or special characters), as described in what follows.

Syntactic simplification begins by looking for semicolons and dashes and splitting sentences at these characters. Sentences are also split after colons if the segment after the colon is a complete sentence and not just an enumeration.

Appositions are replaced by sentences in which the noun phrase referred to by the apposition forms the subject (*X*) and the apposition itself becomes the predicative noun (*Y*), yielding an *X is Y* structure (cf. Example 2).

(2) **German**
Der Artikel wurde von Dr. Meier, dem Leiter der Universitätsklinik, verfasst.
'The article was written by Dr Meier, head of the university hospital.'

Simplified German
Doktor Meier hat den Artikel verfasst.
Meier ist der Leiter von der Universitäts·klinik.
'Doctor Meier has written the article.
Meier is head of the university hospital.'

Rules for rephrasing subordinate clauses all have a similar structure: If a subordinate conjunction is found, the sentence is split at the conjunction and both resulting segments are edited and rephrased to form independent sentences. Suitable connectives that express the rhetorical relation are added to preserve the original meaning, and the correct word order is restored. For instance, sentences containing causal clauses, e.g., with *weil* ('because') or *da* ('since'), are split into two main clauses, and the latter clause is complemented with the connective *deshalb* or *denn* ('thus') to maintain the causal relation (cf. Example 3).

(3) **German**
Weil der Gastgeber noch nicht da ist, müssen die Gäste warten.
'Since the host is not there yet, the guests have to wait.'

Simplified German
Der Gastgeber ist noch nicht da.
Deshalb müssen die Gäste warten.
'The host is not there yet.
Therefore, the guests have to wait.'

Concessive clauses with subjunctions like *obwohl* ('although') are rephrased using the connective *trotzdem* ('however') (cf. Example 4). Consecutive clauses starting with *sodass* ('so that')

[7] Note that different from the English acronym ‚VAT', the German abbreviation ‚MwSt.' is always expanded to its full form ‚Mehrwertsteuer' in reading or speaking.

are rephrased using *deshalb* ('therefore'), possibly shifting the meaning slightly but essentially retaining the information. We found acceptable connectives for rephrasing temporal clauses in *nachdem* ('after'), *bevor* ('before'), *seit* ('since'), and *während* ('while'), yet we could not find a suitable solution for the conjunction *als* ('when/as'). Final clauses are rephrased using the modal verb *wollen* ('want') and the connective *deshalb* ('therefore'). Since the subject is not mentioned overtly in German final clauses containing *um zu* ('in order to'), it has to be retrieved from the main clause (cf. Example 5).

There is no general way of simplifying relative clauses, so we focused on sentential relative clauses, which do not refer to the preceding noun but to the whole sentence or clause. Such sentences can be split at the pronominal adverb, which is then replaced by its cataphoric adverb (cf. Example 4).

(4) **German**
Obwohl er seine Rechnungen immer pünktlich bezahlte, bekam er eine Mahnung, worüber er sich sehr ärgerte.
'Although he always paid his bills on time, he received a reminder, which really bothered him.'

Simplified German
Er hat seine Rechnungen immer pünktlich bezahlt.
Trotzdem hat er eine Mahnung bekommen.
Darüber hat er sich sehr geärgert.
'He has always paid his bills on time.
However, he has received a reminder.
This has really bothered him.[8]

(5) **German**
Um den Text verständlicher zu machen, verwenden wir nur einfache Wörter.
'To make the text easier to understand, we only use simple words.'

Simplified German
Wir wollen den Text verständlicher machen.
Deshalb verwenden wir nur einfache Wörter.
'We want to make the text easier to understand.
Therefore, we only use simple words.'

Coordinate clauses are split at coordinating conjunctions (e.g. *und* ('and'), *oder* ('or'), *aber* ('but'), *dennoch* ('however')). If the second resulting clause is elliptic, the missing subject or predicate is retrieved from the previous clause and the subject shortened, i.e., adjectives, genitive attributes, and prepositional phrases are removed. We allowed for sentences to start with *und* ('and') and *oder* ('or') to emphasize that they are linked to the previous sentence.

(6) **German**
Der junge Beamte an der Grenze überprüft die Reisepässe und kontrolliert das Gepäck der Fluggäste.
'The young officer at the border checks the passports and examines the passengers' luggage.'

Simplified German
Der junge Beamte an der Grenze überprüft die Reise·pässe.
Und der Beamte kontrolliert das Gepäck von den Flug·gästen.
'The young officer at the border checks the passports.
And the officer examines the luggage of the passengers.'

If a passive construction is detected, our system retrieves the grammatical agent indicated by a prepositional phrase starting with *von* ('by'), the object (the subject of the passive phrase), and the action verb (past participle) and generates a sentence in active voice. If the agent is not mentioned, we use the impersonal pronoun *man* ('one') as subject in the active-voice sentence (cf. Example 7). Although impersonal language should be avoided, we decided to accept the pronoun *man* when resolving passive constructions without explicit agent, as it is likely to be less difficult than the original passive construction. To rephrase genitive attributes, the entire attribute is transformed into dative case and complemented with the preposition *von* ('of') (cf. Example 6).

(7) **German**
Der Dieb wurde von der Polizei gefasst. Er wurde in Handschellen abgeführt.
'The thief was arrested by the police.
He was taken away in handcuffs.'

Simplified German
Die Polizei hat den Dieb gefasst.
Man hat ihn in Hand·schellen abgeführt.
'The police has arrested the thief.
One/they have taken him away in handcuffs.'

[8] Note that present perfect tense fulfills a slightly different function in German than it does in English.

If the sentence is in simple past, the tense is changed to past perfect. The auxiliary verb *sein* ('be') or *haben* ('have') is conjugated accordingly; the past participle is simply added at the end of the sentence (cf. Example 7). This works well, since the sentence is already highly simplified and shortened at this point. Auxiliary and modal verbs remain in past simple tense because they are well-known forms and their past perfect use is often deemed unnatural (Maaß, 2015).

3.2.3 Text-level and layout rules

Simplified language requires explanations for difficult words. We regard as difficult vocabulary acronyms (derived from Wikipedia) and words that are explained in the Hurraki online dictionary (cf. Section 3.1). Acronyms are explained after their first occurrence in the text but are not expanded like abbreviations, to avoid long and difficult words. For non-trivial words with a Hurraki entry, the short Hurraki definition is retrieved and inserted into the text. Some Hurraki explanations do not conform to the guidelines of Maaß (2015); we refrained from modifying them. To mark added explanations automatically and make the text more readable, explanations are indented (cf. Example 8). When printing the final simplified text, all sentences resulting from one original sentence are grouped together in a paragraph to emphasize which information belongs together.

(8) **German**
Andreas Meyer ist der Chef der SBB.
'Andreas Meyer is the director of SBB.'

Simplified German
Andreas Meyer ist der Chef der SBB.[9]
'Andreas Meyer is the director of SBB.'

SBB ist die Abkürzung für
Schweizerische Bundesbahnen.
Chef ist ein schwieriges Wort.
Hurraki erklärt es so:
Ein Chef ist im Betrieb der Vorgesetzte oder Verantwortliche.

'SBB is the abbreviation for Swiss Federal Railways.
Director is a difficult word.
Hurraki explains it as follows:
A director is the supervisor or responsible person in a company.'

[9] Since the parser does not recognize *der SBB* as a genitive attribute, it is not modified.

3.3 Evaluation

A common way of evaluating simplified texts is to apply readability metrics. Readability metrics typically assess one or multiple surface features such as word or sentence length. Well-known examples are the Flesch Reading Ease Score (Flesch, 1948) and the *Läsbarhetsindex* ('readability index', LIX) (Björnsson, 1968). Flesch Reading Ease measures word length in syllables and sentence length in words and delivers a score on a scale from 0 to 100, with higher scores indicating better readability. Flesch is frequently used to assess writings of students in U.S. grade schools. LIX computes the sum of the average sentence length and the ratio of long words (i.e., words with more than six letters). Like Flesch, the resulting score ranges between 0 and 100; however, with LIX, higher scores correlate with lower readability.

Metrics like Flesch and LIX are generally understood to cover only a part of what constitutes the readability of a text (Chall, 1958). Heimann Mühlenbock (2013) developed the more sophisticated SVIT model for assessing the readability of Swedish texts. Since the model is partly language-specific, we could not rely on it for evaluating the simplifications produced by our system. Addressing "the current problem in the text simplification community that there are no common standards and evaluation methodologies which would enable fair comparison of different ATS [automatic text simplification; the authors] systems" was the aim of a workshop at the Language Resources and Evaluation Conference.[10]

We evaluated the output of our system both quantitatively, by computing its LIX score (well aware of the shortcomings of this score), and qualitatively, by comparing it to a simplification produced by a person who had undergone the six-month *Leicht Lesen* ('easy-read') training offered for German by the *capito* network.[11]

3.3.1 Data

The evaluation text is a short article on the arrival of the Swiss team at the Special Olympics in Korea. It consists of 135 words in six sentences and features many aspects of standard language: long, difficult, and foreign words, exclamation marks, dashes and colons, appositions, long and elliptic sentences, coordinate clauses, one participle construction, final

[10] http://qats2016.github.io/index.html
[11] http://www.capito.eu/de/Leicht_Lesen/

LIX score	Description	Text type
<25	very easy	children's literature
25-30	easy	young adults' literature
30-40	standard	fiction and daily news
40-50	fairly difficult	informative texts, non-fiction
50-60	difficult	specialist texts
>60	very difficult	scientific texts

Table 1: Description of LIX scale (table from (Heimann Mühlenbock, 2013, p. 32).

clause, and sentential relative clause, passive constructions, genitives, and past simple forms. The text was chosen on the basis that it contains many complex structures. It had not been used to develop the system.

3.3.2 Quantitative evaluation

The human-simplified text contains 152 words in 16 sentences, the text resulting from our simplification consists of 146 words in 14 sentences (with added Hurraki explanations: 217 words in 24 sentences). The original (standard-language) version of the evaluation text has a LIX score of 53, which corresponds to the level of difficulty of specialist texts according to the classification shown in Table 1 (Heimann Mühlenbock, 2013). The LIX score of the human simplification is 35,[12] assigning a "standard level" of difficulty to the text, similar to fictional and daily news texts. The simplification generated by our system has a LIX score of 41, which identifies it as a "fairly difficult" text. Consequently, our system reduced the complexity of the evaluation text from "difficult" to "fairly difficult", while the human simplification was able to further reduce it to a "standard" level of difficulty.

3.3.3 Qualitative evaluation

Upon manual inspection of the differences between the human and the automatic simplification, we observed that the guidelines adhered to in the human simplification slightly differ from the ones we based the simplification rules of our system upon. For example, the human simplification contains commas and does not feature the *Mediopunkt*. Moreover, coordinate clauses with *und* ('and') are not split, explanations are not marked by indentations, and long simplified sentences are displayed on several lines.

[12]Recall that a lower LIX score points towards higher readability.

The biggest difference between the human and the automatic simplification is in lexical complexity. In the human simplification, many difficult words and expressions are replaced by simpler alternatives. For example, in the human simplification, the idiomatic expression *(jemanden) unter die Fittiche nehmen* ('to take (somebody) under the wings') is replaced with *sich kümmern* ('to take care of').

Our system segmented long words like *Meditations·techniken* ('meditation techniques') with the *Mediopunkt* and added Hurraki explanations for the words *Botschaft* ('embassy') and *Chef* ('boss'). Especially the explanation for *Botschaft* seems helpful; the human simplification does not explain this word. The human simplification introduces a new term *Schweizersportler* ('Swiss athletes'), which is not only long and possibly hard to read but also an incorrect compound word (correct: *Schweizer Sportler*).

For both simplifications, an exclamation mark was removed and sentences were split at dash signs and colons. An apposition was rephrased in a similar fashion in both texts. A final clause, sentential relative clause, and coordinative clauses were split in both simplifications, except for two *und* ('and') sentences in the human simplification, which were split only visually through line breaks. Since we had not implemented rules for rephrasing participle constructions, an occurrence of such a construction in the evaluation text remained unchanged by our system, while it was rephrased in the human simplification text.

In both the human and the automatic simplification, passive constructions and genitive attributes are resolved, although our system naturally returns more literal rephrasings. In one sentence, the dependency parser returned incorrect output, as a result of which a prepositional phrase could not be identified as agent. Apart from that, passive constructions were resolved correctly, even elliptic passive sentences. Genitive attributes were also rephrased correctly, with the exception of one case in which the wrong lemma was assigned by the parser, resulting in an incorrect dative form. In both simplifications, the past simple forms were changed to past perfect, with the exception of one sentence in the automatic simplification, where the predicate was not identified correctly by the parser.

While our system simply prints each output sentence on a new line, the human simplification text

splits long sentences at syntactic borders and displays them on several lines to improve readability.

3.4 Discussion

Our system still produces incorrect or unsimplified output for some sentences. For example, several subordinate clauses are not simplified because we have not found a general way of rephrasing them, e.g. conditional clauses, relative clauses, or clauses starting with *dass* ('that'). Furthermore, verbs with separable prefixes such as *ankommen* ('arrive') are not handled well, sometimes because the parser fails to identify the correct lemma, e.g., produces the lemma *kommen* ('come') instead of *ankommen* ('arrive'), and sometimes because our system does not incorporate a full-fledged grammar.

Moreover, our system provides limited support for reducing lexical complexity. However, even though it does not rephrase difficult vocabulary as readily as a human simplifier would, visual compound segmentation and the addition of explanations still aid in improving readability on the lexical level.

Overall, when applied to the evaluation text, our simplification system produced a readable simplified text. Especially on the syntactic level, the output of our system is comparable in complexity to the human simplification.

4 Conclusion and outlook

In this paper, we have reported on the development of a rule-based text simplification system for German. The rules underlying our system are based on linguistically motivated guidelines for transforming standard German into simplified German. Our system applies rules that perform simplification on the character, word, sentence, and text level and adjust the layout of the output.

We evaluated our system both quantitatively and qualitatively. With regard to quantitative assessment, our system was capable of reducing the complexity of the evaluation text from "difficult" to "fairly difficult" as measured by the LIX readability metric. The qualitative evaluation showed that our system does not reduce difficult vocabulary as readily as a human simplifier would; however, because compounds are segmented visually and explanations are added, readability is still increased at the lexical level. The syntactic complexity of the text output by our system was comparable to that of the human simplification.

In further developing our system, we intend to put more emphasis on lexical simplification, as even a text with short and simple sentences can be hard to read for inexperienced readers if it features high lexical density and contains difficult words. Apart from that, we are going to extend the syntactic simplification component in our system. As more resources for simplified German become available, we will be able to include synonym replacement, more elaborate explanation generation, and picture extraction in our system.

An automatic simplification system can only ever be as good as the rules it is built upon. In this sense, further theoretical research on simplified language is needed. In particular, the (potentially diverging) needs of the different target groups should be studied to greater detail. Preliminary efforts in linking the concept of simplified language to different levels of the Common European Framework of Reference for Languages (CEFR) are underway. For example, the *capito* network proposed three gradations of simplified language corresponding to the CEFR levels A1, A2, and B1.[13] Once these gradations become more formalized, it will be possible to implement them into automatic text simplification systems. As a result, these systems will be capable of producing different degrees of simplifications.

In further pursuing this work, collaboration with target readers will be most valuable for both designing rules and evaluating the system.

References

Sandra Maria Aluísio and Caroline Gasperin. 2010. Fostering digital inclusion and accessibility: The PorSimples project for simplification of Portuguese texts. In *Proceedings of the NAACL HLT 2010 Young Investigators Workshop on Computational Approaches to Languages of the Americas*, pages 46–53, Los Angeles, CA.

Carl-Hugo Björnsson. 1968. *Läsbarhet*. Liber, Stockholm.

Bundesministerium der Justiz und für Verbraucherschutz. 2011. Verordnung zur Schaffung barrierefreier Informationstechnik nach dem Behindertengleichstellungsgesetz (Barrierefreie-Informationstechnik-Verordnung BITV 2.0). http://www.gesetze-im-internet.de/

[13] https://www.capito.eu/de/Angebote/Barrierefreie_Information/capito_Qualitaets-Standard/Guetesiegel_fuer_Leicht_Lesen/

bitv_2_0/BJNR184300011.html. Online. Last accessed March 3, 2016.

Jeanne Sternlicht Chall. 1958. *Readability: An appraisal of research and application*. Bureau of Educational Research, Ohio State University, Columbus, OH.

Raman Chandrasekar, Christine Doran, and Bangalore Srinivas. 1996. Motivations and methods for text simplification. In *Proceedings of the 16th Conference on Computational Linguistics*, pages 1041–1044, Copenhagen, Denmark.

William Coster and David Kauchak. 2011a. Learning to simplify sentences using Wikipedia. In *Proceedings of the Workshop on Monolingual Text-To-Text Generation (MTTG)*, pages 1–9, Portland, OR.

William Coster and David Kauchak. 2011b. Simple English Wikipedia: A new text simplification task. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (HLT)*, pages 665–669, Portland, OR.

Rudolph Flesch. 1948. A New Readability Yardstick. *Journal of Applied Psychology*, 32:221–233.

Mariikka Haapalainen and Ari Majorin. 1995. GERTWOL und morphologische Disambiguierung für das Deutsche. In *Proceedings of the 10th Nordic Conference on Computational Linguistics*, Helsinki, Finland.

Katarina Heimann Mühlenbock. 2013. *I see what you mean: Assessing readability for specific target groups*. Ph.D. thesis, University of Gothenburg.

Inclusion Europe. 2009. Information für alle: Europäische Regeln, wie man Informationen leicht lesbar und leicht verständlich macht. http://www.inclusion-europe.org/images/stories/documents/Project_Pathways1/DE-Information_for_all.pdf. Online. Last accessed June 21, 2015.

Sasikiran Kandula, Dorothy Curtis, and Qing Zeng-Treitler. 2010. A semantic and syntactic text simplification tool for health content. In *AMIA Annual Symposium Proceedings*, volume 2010, pages 366–370.

David Klaper, Sarah Ebling, and Martin Volk. 2013. Building a German/Simple German Parallel Corpus for Automatic Text Simplification. In *Proceedings of the 2nd Workshop on Predicting and Improving Text Readability for Target Reader Populations*, pages 11–19, Sofia, Bulgaria.

Christiane Maaß. 2015. *Leichte Sprache: Das Regelbuch*. LIT-Verlag, Berlin.

Kerstin Matausch and Annika Nietzio. 2012. Easy-to-Read and Plain Language: Defining Criteria and Refining Rules. http://www.w3.org/WAI/RD/2012/easy-to-read/paper11/. Online. Last accessed: November 13, 2015.

Netzwerk Leichte Sprache. 2009. Die Regeln für Leichte Sprache. http://www.leichtesprache.org/images/Regeln_Leichte_Sprache.pdf. Online. Last accessed June 21, 2015.

Gustavo Henrique Paetzold and Lucia Specia. 2015. LEXenstein: A Framework for Lexical Simplification. *ACL-IJCNLP 2015*, 1(1):85.

Evelina Rennes and Arne Jönsson. 2015. A tool for automatic simplification of Swedish texts. In *Proceedings of the 20th Nordic Conference of Computational Linguistics*, pages 317–320.

Rico Sennrich, Gerold Schneider, Martin Volk, and Martin Warin. 2009. A new hybrid dependency parser for German. *Proceedings of the German Society for Computational Linguistics and Language Technology Conference*, pages 115–124.

Matthew Shardlow. 2014. A survey of automated text simplification. *International Journal of Advanced Computer Science and Applications (IJACSA), Special Issue on Natural Language Processing*, pages 58–70.

Advaith Siddharthan. 2006. Syntactic simplification and text cohesion. *Research on Language & Computation*, 4(1):77–109.

Advaith Siddharthan. 2014. A survey of research on text simplification. *International Journal of Applied Linguistics*, 165(2):259–298.

Lucia Specia. 2010. Translating from Complex to Simplified Sentences. In *Computational Processing of the Portuguese Language. Proceedings of the 9th International Conference, PROPOR 2010*, pages 30–39, Porto Alegre, Brazil.

Sara Stymne, Jörg Tiedemann, Christian Hardmeier, and Joakim Nivre. 2013. Statistical Machine Translation with Readability Constraints. In *Proceedings of the 19th Nordic Conference of Computational Linguistics*, pages 375–386.

Martin Volk. 1999. Choosing the right lemma when analysing German nouns. In *Proceedings of the 11. Jahrestagung der GLDV*, pages 304–310, Frankfurt, Germany.

Willian Massami Watanabe, Arnaldo Candido Jr, Marcelo Adriano Amâncio, Matheus De Oliveira, Thiago Alexandre Salgueiro Pardo, Renata PM Fortes, and Sandra M Aluísio. 2010. Adapting Web content for low-literacy readers by using lexical elaboration and named entities labeling. *New Review of Hypermedia and Multimedia*, 16(3):303–327.

Building a Parallel Corpus on the World's Oldest Banking Magazine

Martin Volk **Chantal Amrhein** **Noëmi Aepli** **Mathias Müller** **Phillip Ströbel**

University of Zurich
Institute of Computational Linguistics
volk@cl.uzh.ch

Abstract

We report on our processing steps to build a diachronic parallel corpus based on the world's oldest banking magazine. The magazine has been published since 1895 in German, with translations in French and partly in English and Italian. Our data sources are printed issues (until 1997), PDF issues (since 1998) and HTML files (since 2001). The corpus building poses special challenges in article boundary recognition and cross-language article and sentence alignment. Our corpus fills a gap in parallel corpora with respect to genre (magazine articles), domain (banking and economy articles), and its time span (120 years).

1 Introduction

Translated documents in multiple languages (parallel corpora) are highly regarded as valuable resources for natural language processing and linguistic research. But the genres of available parallel corpora are limited to the proceedings of multilingual parliaments (e.g. Europarl), law collections of international bodies (e.g. Acquis Communautaire), subtitles and transcripts (e.g. OpenSubtitles or TED talks) and software manuals. Our goal is to complement these collections with corpora in new genres, domains and a diachronic dimension.

Towards this goal we are building a corpus on the basis of the world's oldest banking magazine, the Credit Suisse Bulletin, which has been published since 1895. The texts in this publication series revolve around all banking issues such as investments, savings, stock prices, but also cover a wide range of topics such as sports, traveling and culture. Over the years, the Bulletin has developed into a full-fledged magazine, currently being published 5 times per year with around 80 pages each in four languages: English, French, German and Italian. Thus we are compiling a parallel corpus in a text genre (magazine articles) that is otherwise seldomly translated. Eventually our corpus will cover a period of more than 120 years.

In addition to the Bulletin, Credit Suisse publishes web news some of which are summaries of the articles in the Bulletin. We will include these news in the corpus since they are valuable resources for comparison.

This paper presents our design decisions for the Credit Suisse Bulletin corpus. First, we introduce the source documents for our collection, then we describe our processing steps and output format, and finally we discuss the challenges in this undertaking. In particular, we propose an algorithm to disambiguate lemmas based on the parallel documents. Overall, this paper focuses on sharing experiences in building a parallel diachronic corpus.

2 Corpus Sources

The Credit Suisse Bulletin has been published since 1895. The copies from the start until 1997 are available only as printed and bound journals in the Credit Suisse archives and some libraries. Scanning bound books is a time-consuming endeavour which requires either manual page turning or expensive scan robots. Therefore, we initially decided that we will use only those copies that we are allowed to cut at the spine for scanning with automatic paper feed. The Credit Suisse archive donated their duplicates from the years 1981 to 1996, in total 262 magazine issues in the four languages English, French, German and Italian. In addition, the library of the Swiss National Bank has offered their issues for cutting, scanning and rebinding, and the Swiss National Library in Berne has agreed to digitize the remaining copies for us.

Second, the Bulletin issues from 1998 to date are available as PDF documents from the Credit

Suisse website[1]. We have downloaded a total of 396 Bulletins as PDF files from which we extracted text content and document structure.

In neither the printed issues nor the PDF issues all articles are translated into all four languages. French and German are mostly given, while English and Italian are sometimes missing. This leads to interesting challenges in document alignment.

The third source of documents for our corpus are the news that are published as HTML documents on the bank's web page. Many of them represent modified versions of articles that were published in the magazine. We crawled all these news (from 2001 to date) which amount to roughly 1500 articles (1.7 million tokens) per language. With a cleaning script, we extracted the text and annotated it with XML markup for title, author, date, and category (such as economy, entrepreneurs, investing, Switzerland).

3 Steps in Corpus Building

The initial steps in corpus building differ depending on the corpus sources.

3.1 Converting Scanned Documents

We converted our scanned Bulletin issues into text with the Abbyy Recognition Server[2]. This OCR program outputs a detailed layout XML with character level information of the coordinates, the system's recognition confidence and the font size as well as word-level information on whether the word is in the OCR system's internal lexicon. The page position coordinates allow us to ignore text in header or footer lines which we do not want in the corpus. They also allow us to detect the page numbers. The font size provides basic information for the detection of the article boundaries. Subsequently, we convert and tokenize the Abbyy XML output into an intermediate XML format.

OCR leads to a small word error rate. For the period which we have digitized so far (1981 to 1996), the recognition accuracy is very high (in text blocks we found less than one incorrect letter in 1000 letters; there are more errors in the occasional words on images). We expect the error rate to be somewhat higher for older Bulletin issues, not least because of more words outside the OCR system's lexicon. If necessary, we will employ the correction methods that we developed in previous projects (Volk et al., 2011). In addition, we consider new methods for word error corrections based on automatic word alignment across the different language versions.

[1] http://publications.credit-suisse.com/index.cfm/publikationen-shop/bulletin/

[2] We gratefully acknowledge support by Abbyy GmbH.

3.2 Converting PDF Documents

There are a number of tools for the extraction of text from PDF documents. Some are freely available, others are part of commercial tools such as Adobe Acrobat. We found that many of those tools deliver ill-formed text, which in our case meant words that were glued together (e.g. *JapansWirtschaft, ein radikalanderes Schulsystem*). After a thorough evaluation we purchased PDFLib TET since it gave the best results and outputs layout XML in a format similar to Abbyy's Recognition Server. Again we get character coordinates and font sizes.

We had hoped that font sizes for article headers in the Bulletin are consistent for certain publication periods. Unfortunately, font sizes for titles are neither consistent nor unique. They differ from year to year and from issue to issue. Sometimes they even differ within the different language versions of the same issue. For example, if an English title is short and set in font size 48, it happens that the corresponding German title is longer and therefore decreased to font size 44 in order to fit the space on the page. Moreover, a large font does not always mean an article headline. We found cases where the same large font is part of an illustration or an advertisement. With this approach the precision of article boundary detection varies from 75% to 95% while recall is as low as 65% and 70% respectively.

3.3 Article Boundary Recognition

Since our results for automatic title detection using only font sizes were not satisfactory, we performed article boundary recognition based on the table-of-content. While testing our prototype system, we realized that the layouts of the individual issues differ more than expected. Sometimes advertisement pages were not included in the page numbering. Therefore, it was not possible to work with incremental page counts, but rather the page numbers had to be extracted directly from the page. Unfortunately, in some years the page numbers where located at the bottom of the pages and in others at the top. Additionally, for a small number of issues the table of contents was stretched across two non-consecutive pages, while in some special issues

Figure 1: Bulletin 2014, number 2, title page in German and English

the table of contents was missing entirely. Thus, our system required high adaptability to individual layouts.

Another problem arose because of the differences in wording between the article listing in the table of contents and the actual titles in the magazines. For example, we find "Family and Career. Six Women Show How It's Done" in the table of contents of the English version, but the title of the article is "The Art of Compromise". For the majority of articles, it was not possible to match the strings from the table of contents and the article headers. We cannot use this information to identify article boundaries. Moreover, it happens that article titles span over two pages, and that they are integrated in images which led to errors in the text extracted by PDFLib TET. For this reason, we provide the option to confirm or reject the title candidates suggested by our system.

This is how it works: First, we automatically analyze the table-of-content page(s) in order to extract the page numbers where a new article starts. We navigate to the corresponding pages either with an offset, if pagination is continuous, or with page numbers extracted at the top or bottom of the pages. We then search for title candidates. A font size threshold is used to identify large segments on the page. The resulting candidates can be manually confirmed or rejected. If the system does not find an article header, e.g. if it is part of an image, there is also an option to mark a starting article at the beginning of the page. It is possible to extract the article headers fully automatically, however the results will not be as good.

Precision		Recall	
Range	Issues	Range	Issues
0.5 - 0.6	-	0.5 - 0.6	3
0.6 - 0.7	-	0.6 - 0.7	3
0.7 - 0.8	9	0.7 - 0.8	2
0.8 - 0.9	39	0.8 - 0.9	43
0.9 - 1.0	115	0.9 - 1.0	89
1.0	207	1.0	228
avg: 0.961	sum: 370	avg: 0.964	sum: 370

Table 1: Article Boundary Detection Quality: Rows 1 to 6 show the distribution of precision and recall. The last row presents the average results for precision and recall as well as the total number of issues evaluated

Since we achieved satisfactory results when testing our approach on a small number of magazine issues, we performed a large-scale evaluation while we used the system to detect and mark the article boundaries of all PDF issues. If one of the proposed headers was confirmed to be an actual article title, we counted it as a true positive; if none of the proposed candidates matched the actual title, they were collectively counted as one false positive. The number of false negatives was calculated by subtracting the number of found articles from the total number of articles in the table of contents.

Table 1 shows that the F-measure for the majority of articles is between 0.9 and 1.0. These are extremely good results compared to our initial experiments, when we only used font sizes to identify article boundaries. Of course, one has to keep in mind that the evaluation includes a semi-automatic check of the article titles. Without this step the quality would clearly be lower. However, manually checking the proposed candidates does not take much time and is therefore recommended.

Our approach is related to techniques described in (Dejean, 2015) for identifying specific data fields in architectural plans. The tasks are similar in that we also work with unstructured text and use layout characteristics such as font size and text position on the page in order to identify potential article headings. However, unlike content tagging presented in (Dejean, 2015), we do not combine our layout analysis with textual information, for example by making use of the descriptions in the content page or measuring the length of a title candidate. For our task, we achieve very good results by only using the layout. Additionally, we do not generate a data model for every magazine individually. Instead, we use one font size threshold for all magazines, which can of course be adapted for individual issues if needed.

3.4 Corpus Size of the PDF Magazines

All the different corpus sources are stored in an intermediate XML format. To build our parallel corpus, we tokenized and tagged all files. The current size of the Bulletin corpus based on PDF documents is displayed in table 2. For German, French and Italian we have more than 3 million tokens per language. For English we have close to 2 million tokens because English translations have only recently become available for the PDF series.

As for lemmas, table 3 shows the number of unique lemma types as well as the absolute number of tokens whose lemma is unknown for all languages. This should be read as follows: The German corpus which was built on the Bulletin PDF magazines has 3.48 million tokens which account for 201,611 different types (leaving upper case untouched). Out of these 3.48 million tokens we were unable to compute a lemma for 161,644 tokens. Those were unknown to our dictionary and to our morphology analyzer (e.g. loan words like *E-commerce*, names like *Calderón*, uncommon spellings like *.com-Manie*). The remaining 3.32 million tokens can be mapped to 102,215 different lemmas. German has about three times the number of unique lemma types compared to the other languages due to frequent compounds in German.

	Sentences	Tokens	Types
DE	343,620	3,482,804	201,611
EN	166,397	2,077,319	75,272
FR	307,555	3,838,037	113,942
IT	259,868	3,232,256	117,381

Table 2: Corpus size: number of sentences, tokens and word types for all languages in the PDF corpus

	Lemma Types	Unknown Lemmas
DE	102,215	161,644
EN	32,527	70,156
FR	24,494	205,769
IT	24,992	212,824

Table 3: Corpus size: number of unique lemma types (excluding type "unknown") and unknown lemmas (counted per token) for all languages in the PDF corpus

4 Parallel Corpus Alignment

In order to exploit our parallel corpus, we need to compute cross-language alignments on all levels. First, we need to determine document alignments. This is particularly tricky for those Bulletin issues where not all articles were translated. For example, in the 1980s only about half the articles in the parallel German and French issues were translated

```
<corpus>
  <article n="a223" id="cs-2013-03-15-Bali">
      <h1 cat="Society">
        <s n="a223-s1">
            <w n="a223-s1-w1" lemma="a" pos="DT">A</w>
            <w n="a223-s1-w2" lemma="New" pos="NP">New</w>
            <w n="a223-s1-w3" lemma="Life" pos="NP">Life</w>
            <w n="a223-s1-w4" lemma="in" pos="IN">in</w>
            <w n="a223-s1-w5" lemma="Northern" pos="NP">Northern</w>
            <w n="a223-s1-w6" lemma="Bali" pos="NP">Bali</w>
        </s>
      </h1>
      <p class="date">
        <s n="a223-s2">
            <w n="a223-s2-w1" lemma="@card@" pos="CD">05.03.2013</w>
        </s>
      </p>
      <p class="abstract">
        <s n="a223-s3">
            <w n="a223-s3-w1" lemma="for" pos="IN">For</w>
            <w n="a223-s3-w2" lemma="@card@" pos="CD">35</w>
            <w n="a223-s3-w3" lemma="year" pos="NNS">years</w>
      <!-- ... -->
</corpus>
```

Figure 2: XML structure of the corpus

into English and Italian. So, which articles are present across languages? And are they full-length translations or only abbreviated versions?

As a first step towards article alignment, we use the automatically detected article boundaries together with author information. We then check for overlapping names and numbers in the documents. Finally, we compare the article structure (number of paragraphs) and length (in characters) to decide on full-length vs. abbreviated translation. Our first results indicate that in most cases we have full-length translations which makes for a valuable parallel corpus.

Based on the aligned articles we compute sentence alignment. Since the articles in the OCR version and the PDF version contain noise at different places in the text, we need a robust alignment method. We use Bleualign (Sennrich and Volk, 2011) with machine translation of language 1 into language 2 in order to align the sentences in the language 1 text to the corresponding sentences in language 2. The machine translation output is compared with the help of a simplified version of the BLEU metric to the sentences in the language 2 text. This metric, combined with a diagonalization heuristic, results in high precision sentence alignments. Figure 3 shows the final representation of sentence alignments in XML.

Subsequently, word alignment can be performed with GIZA++, Berkeley aligner or any other word aligner of choice. Word alignment will not be included in the corpus release because it is clearly application-specific. Machine translation needs a recall-oriented word alignment while linguistic research requires a high precision word alignment.

5 Corpus Annotation and Lemma Disambiguation

We use the TreeTagger to annotate the corpus with Part-of-Speech (PoS) tags and lemmas in all four languages. The TreeTagger assigns a PoS tag to each token and additionally assigns a lemma if it has seen the token in its training corpus. In case it has seen multiple lemmas for a specific word form, it will assign multiple lemmas.

Since we often find such ambiguous words with multiple lemmas in our German corpus, we investigated two methods for the disambiguation of these lemmas. First, we select among lemma options when we re-attach separated verb prefixes to the lemma. For example, the 1st/3rd person plural verb form *drängen* can have the lemmas *drängen* (EN: to urge) or *dringen* (EN: to insist). The TreeTagger assigns both lemmas to this verb form. If *drängen* occurs with the separated prefix *auf*, then our reattachment algorithm finds that only the combination *aufdrängen* is possible (EN: to force on, to impose), and we remove the other lemma option. This approach is described in more detail in (Volk et al., 2016).

Second, we use the parallel texts for the resolution of these lemma ambiguities. The following example illustrates the advantage of using an English translation to disambiguate German lemmas:

(1) German: *Viele Wege führen nach Rom.*

English: *Many roads lead to Rome.*

The TreeTagger annotates the German word "führen" with the two lemmas "fahren|führen", because when interpreted as a finite verb form, "führen" can be 1st and 3rd person plural in present tense of "führen" (EN: to lead) or a subjunctive form of "fahren" (EN: to go, to drive). However, if the English translation is considered, it becomes clear that the intended meaning in example 1 is "führen". Therefore, we use the parallel articles in English to disambiguate lemmas in the German corpus with the following approach.

First, we computed token alignments between lemmas with GIZA++ over the whole sentence-aligned news part of our corpus (roughly 1.7 million tokens per language) and obtained the lexical translation probabilities in both directions: German to English and English to German. Then, we extracted the German sentences in which a word has multiple lemmas. Using sentence alignment, we retrieved the English translation of each sentence. Next, we searched for the most likely lexical translation of each possible lemmas (as one token) in the English sentence. We checked whether the lexical translation probability from the English lemma to one of the German lemma options is higher than the others. If so, we accepted the more likely lemma and in this way disambiguated the German word.

This is in essence similar to using parallel corpora for word sense disambiguation (as e.g. described in (Shahid and Kazakov, 2013) for Europarl and (Lefever and Hoste, 2014)). Of course, we capture different word senses only if they are reflected in different lemmas. This means that we work more coarse-grained than word sense disambiguation methods which distinguish WordNet senses.

Let us exemplify our algorithm with the above example sentence. We determine the most likely translation of the ambiguous lemma "fahren|führen" by checking the lexical translation probabilities for each token in the parallel English sentence. The pair "fahren|führen – lead" is the top candidate. We then compare the lexical translation probabilities for "fahren – lead" and for "führen – lead". Since the probability for the latter is higher, the lemma "führen" wins.

In this approach, we excluded the combined lemmas "er|es|sie" (which is the lemma that the TreeTagger assigns to the frequent reflexive pronoun "sich") as well as ambiguities due to polite pronoun forms "sie|Sie" (EN: they, you). They cannot be disambiguated reliably with our approach because English has no grammatical gender and no explicit polite forms. We evaluated the quality of our disambiguation system with 100 disambiguated lemmas and reached a precision of 0.97 as shown in table 4 in the first row. However, only about 16% of the lemma ambiguities could be resolved. Therefore, we generalized our approach in order to disambiguate more lemmas.

We were able to improve our recall by assuming that whenever only one of the possible lemmas occurs in the lemma alignments, this should be the disambiguated form. Of course, we cannot avoid creating some false positives with this method, but the number of true positives is far larger. For example, the German word form "Stunden" is ambiguous (EN: "hours" or the nominalized form of "to defer"). The lemma can either be "Stunde" or "Stunden". However, only "Stunde" occurs in our corpus in the lemma alignments on its own. Therefore, all lemmas "Stunde|Stunden" in the corpus are disambiguated to "Stunde". Table 4 shows that with this assumption, we were able to disambiguate about 75% of all ambiguous lemmas. Again, we tested the quality on 100 disambiguated lemmas and reached a precision of 0.93.

As another measure to improve recall, in all cases where no lemma has a higher probability than the others, we checked whether any of the individual lemmas occurs more often in lowercased form. The usefulness of this approach can be observed in the following example:

(2) German sentence:
Es gibt ja den Spruch "Lesen bildet."

English translation:
It is said that: "Reading makes you smarter."

The German word "Lesen" is ambiguous because it could either be the gerund of "to read" but it could also be the plural form of "harvest". The most likely translation of "Lese|Lesen" is the English lemma "reading". Neither the pair "Lese – reading" nor "Lesen – reading" occurs in the lemma alignments in our corpus. Consequently,

```
<linkGrp toDoc="CS_news_corpus_en.xml" fromDoc="CS_news_corpus_de.xml">
        <link type="1-1" xtargets="a96-s1; a1-s1"/>
        <link type="1-1" xtargets="a96-s2; a1-s2"/>
        <link type="1-1" xtargets="a96-s3; a1-s3"/>
        <link type="1-1" xtargets="a96-s4; a1-s4"/>
        <link type="1-1" xtargets="a96-s5; a1-s5"/>
        <link type="1-1" xtargets="a96-s6; a1-s6"/>
        <link type="1-1" xtargets="a96-s7; a1-s7"/>
        <link type="1-1" xtargets="a96-s8; a1-s8"/>
        <link type="1-2" xtargets="a96-s9; a1-s10 a1-s11"/>
        <link type="1-2" xtargets="a96-s10; a1-s12 a1-s13"/>
        <link type="1-1" xtargets="a96-s11; a1-s14"/>
        <!-- ... -->
</linkGrp>
```

Figure 3: XML stand-off annotation for sentence alignments

	Disambig.	Precision
only LTP	16%	97%
LTP & General	75%	93%
LTP & General & Lower	84%	91%

Table 4: Results for automatic lemma disambiguation (LTP = lexical translation probability, General = generalized method, Lower = lemmas also checked in lowercase form). The table shows for each method the percentage of lemmas that we were able to disambiguate and the precision calculated over 100 disambiguated lemmas.

the lemma of the word “Lesen” would remain ambiguous. But when applying the method described above, we also check whether the lowercased German lemmas occur in the word alignments together with “reading”. Since the pair “lesen – reading” exists in our corpus and “lese – reading” does not, “Lesen” is accepted as the correct lemma and the word “Lesen” is disambiguated. The results of this refined method can be found in table 4 in the last row. We were able to disambiguate about 84% of the ambiguous lemmas. Tested on 100 disambiguated lemmas, we achieved a precision of 0.91.

We have shown that our lemma disambiguation approach works well for German when using English alignments for the disambiguation. It remains to be investigated how the results change when using French or Italian as parallel texts either alone or in combination. For example, both languages use polite forms which we can exploit to disambiguate even more German lemmas.

Another idea is to use a machine translation system to first translate sentences for which we do not have a parallel text. Then, using our trained lemma alignments from the corpus, we can also disambiguate lemmas for which we do not have translations. We hope to enhance our results further by including more aligned sentences for the training of the lemma alignments. As of now, we evaluated the lemma disambiguation only on the news part of our corpus but we will include the larger Credit Suisse Bulletin corpus in future experiments.

6 Corpus Representation

We store the corpus in a custom XML format[3], see Figure 2 for an example. The outermost element is called “corpus”, and it contains a sequence of “article” elements. “article” elements come with an “id” attribute to facilitate document alignment across languages. Within articles, we preserve headers and paragraph structure. Also, we retain information on the general category of the news articles (banking, economy, society, sport etc.). Similar to TEI documents, the “s” element represents a sentence and it contains “w” elements for single words. Each word element has attributes to store lemma and PoS information. We distribute the corpus files together with information on sentence alignments and a document type definition (DTD) with which the validity of the documents can be proven.

We computed the sentence alignment of our corpus with HunAlign in the version that is integrated into InterText (Vondřička, 2014). We store these sentence alignments in separate files as stand-off annotations that link two corpus files, see figure 3.

[3]We will provide an XSLT transformation to convert the corpus into valid TEI documents.

The format allows a quick overview of the alignment types. For example, in the English-German news part of our corpus we have 77,651 sentence alignments of type 1-1 and 8230 alignments of type 1-2, in contrast to 1616 of type 2-1. The aligner also returns zero alignments (2736 0-1 vs. 905 1-0 alignments). There are also a few unbalanced alignments (e.g. 3 times 1-6 alignments and one 7-1 alignment) which indicate omissions in one of the languages.

7 Related Work

This work is a continuation of our efforts to build large diachronic parallel corpora for different text genres. In the past, we have built a multilingual corpus of alpine texts (mountaineering reports, articles about the climate, the geology and geography of the world's mountain regions) (Volk et al., 2010). That corpus is special in terms of genre and also because it spans over 150 years (from 1864 until today), with parallel texts in German and French from 1957 until today (and additional Italian translations since 2012). It also includes untranslated English texts by the British Alpine Club.

Our work is related to others in various ways. Work on building large parallel corpora is, for instance, described in publications by (Steinberger et al., 2006) for the JRC Acquis, by (Lison and Tiedemann, 2016) for OpenSubtitles, and by (Ziemski et al., 2016) for a United Nations corpus over six languages. But there is no literature on building genre-specific diachronic parallel corpora which involve OCR of documents and a longer time span. With respect to the banking domain, our Bulletin corpus is related to the European Central Bank corpus available from the OPUS website[4]. But that corpus is based solely on web site information from recent years.

Regarding applications, there are many papers on using parallel corpora for tasks as diverse as translation studies (Zanettin, 2012) or bilingual terminology extraction, see e.g. (Bertaccini and Tadolini, 2011) and (Macken et al., 2013), not to mention statistical machine translation. Noteworthy are also the various usage-oriented online lexicon systems that are based on parallel corpora and word alignment like Linguee, Glosbe or Multilingwis (Volk et al., 2014, Clematide et al., 2016).

[4]http://opus.lingfil.uu.se/ECB.php

8 Conclusions

We have described the necessary considerations when building a large multilingual corpus based on source documents in various formats (printed, PDF, HTML files). We have shared our experiences with OCR tools and PDF converters. We argued that precise article boundary recognition is central to high quality article alignment across languages. We have suggested a method to semi-automatically detect the article boundaries with high precision.

We also developed two methods for selecting among multiple lemmas which were assigned by the PoS tagger. The first method exploits constraints given by separated verb prefixes. The other, more general method relies on cross-language word alignment and translation probabilities in the parallel corpus.

Our current version of the Bulletin corpus consists of roughly 5 million tokens in each of the three languages French, German, and Italian (1.7 million from the news and 3.3 million from the PDF files) and somewhat less for English. Digitization and corpus building of the issues prior to 1998 is ongoing. We expect to collect a total of 20 million each for French and German by the end of the project.

The corpus is distributed in XML with PoS tags, lemmas and sentence alignments. It is freely available for research purposes.

Acknowledgments

We are grateful to the many students who have contributed to the Bulletin4Corpus project. In particular, we acknowledge valuable contributions by Till Salinger (PDF conversion), Dolores Batinic and Fabienne Leuenberger (sentence alignment), Dominique Sandoz (web crawling), and Katrin Affolter (processing of the Abbyy OCR output).

We would like to thank Credit Suisse for their consent that the Bulletin texts can be made available for language technology research. We also acknowledge valuable support by the library of the Swiss National Bank in Zurich and the Swiss National Library in Berne.

References

Franco Bertaccini and Marianna Tadolini. 2011. Banking terminology: creation of a terminology database Italian-German. In *Proceedings of the First International Conference on Terminology, Languages, and Content Resources*, Seoul.

Simon Clematide, Johannes Graën, and Martin Volk. 2016. Multilingwis – a multilingual search tool for multi-word units in multiparallel corpora. In Gloria Corpas Pastor, editor, *Computerised and Corpus-based Approaches to Phraseology: Monolingual and Multilingual Perspectives / Fraseología computacional y basada en corpus: perspectivas monolingües y multilingües*. Tradulex, Geneva.

Hervé Dejean. 2015. Extracting structured data from unstructured documents with incomplete resources. In *Proceedings of the 13th International Conference on Document Analysis and Recognition (ICDAR)*, pages 271–275, Nancy.

Els Lefever and Véronique Hoste. 2014. Parallel corpora make sense: Bypassing the knowledge acquisition bottleneck for word sense disambiguation. *International Journal of Corpus Linguistics*, 19(3):333 – 367.

Pierre Lison and Jörg Tiedemann. 2016. Opensubtitles2016: Extracting large parallel corpora from movie and TV subtitles. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Portorož, Slovenia.

Lieve Macken, Els Lefever, and Veronique Hoste. 2013. Texsis: Bilingual terminology extraction from parallel corpora using chunk-based alignment. *Terminology*, 19(1):1–30.

Rico Sennrich and Martin Volk. 2011. Iterative, MT-based sentence alignment of parallel texts. In *Proceedings of The 18th International Nordic Conference of Computational Linguistics (Nodalida)*, Riga.

Ahmad R. Shahid and Dimitar Kazakov. 2013. Using parallel corpora for word sense disambiguation. In *Proceedings of Recent Advances in Natural Language Processing*, pages 336–341, Hissar, Bulgaria.

Ralf Steinberger, Bruno Pouliquen, Anna Widiger, Carmelia Ignat, Tomaz Erjavec, Dan Tufiş, and Daniel Varga. 2006. The JRC-Acquis: A multilingual aligned parallel corpus with 20+ languages. In *Proceedings of LREC*, Genoa.

Martin Volk, Noah Bubenhofer, Adrian Althaus, Maya Bangerter, Lenz Furrer, and Beni Ruef. 2010. Challenges in building a multilingual alpine heritage corpus. In *Proceedings of LREC*, Valletta, Malta.

Martin Volk, Lenz Furrer, and Rico Sennrich. 2011. Strategies for reducing and correcting OCR errors. In C. Sporleder, A. van den Bosch, and K. Zervanou, editors, *Language Technology for Cultural Heritage: Selected Papers from the LaTeCH Workshop Series*, Theory and Applications of Natural Language Processing, pages 3–22. Springer-Verlag, Berlin.

Martin Volk, Johannes Graën, and Elena Callegaro. 2014. Innovations in parallel corpus search tools. In *Proceedings of LREC*, Reykjavik.

Martin Volk, Simon Clematide, Johannes Graën, and Phillip Ströbel. 2016. Bi-particle adverbs, PoS-tagging and the recognition of German separable prefix verbs. In *Proceedings of KONVENS*, Bochum.

Pavel Vondřička. 2014. Aligning parallel texts with InterText. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, Reykjavik, Iceland. European Language Resources Association (ELRA).

Federico Zanettin. 2012. *Translation-driven corpora: corpus resources for descriptive and applied translation studies.* St. Jerome Publishing.

Michał Ziemski, Marcin Junczys-Dowmunt, and Bruno Pouliquen. 2016. The United Nations parallel corpus v1.0. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Portorož, Slovenia.

Bi-particle Adverbs, PoS-Tagging and the Recognition of German Separable Prefix Verbs

Martin Volk, Simon Clematide, Johannes Graën, Phillip Ströbel
University of Zurich
Institute of Computational Linguistics
volk@cl.uzh.ch

Abstract

In this paper we propose an algorithm for computing the full lemma of German verbs that occur in sentences with a separated prefix. The algorithm is meant for large-scale corpus annotation. It relies on Part-of-Speech tags and works with 97% precision when the tags are correct. Unfortunately there are multi-word adverbs with particles that are homographs with separated verb particles and prepositions. Since the usage as separated particle and preposition is much more frequent, these multi-word adverbs are often incorrectly tagged. We show that special treatment of these bi-particle adverbs improves the re-attachment of separated verb particles.

1 Introduction

Particle verbs in German often occur with verb stem and particle split over long distances. This happens in matrix clauses when the verb is finite and occurs in present or past tense, or when the verb is in imperative form. Examples:

(1) So **wies** eine bekannte Studie der Harvard University aus dem Jahr 2007 **nach**, dass ... (EN: A well-known study by Harvard University from 2007 **proved** that ...)

(2) **Nimm** das und das **mit**. (EN: **Take** this and that **along**.)

In all other tenses and forms the particle is prefixed to the verb (e.g. ... *wie eine Studie nachwies*). Therefore the particle is often called a separable prefix. When analyzing German sentences we have to re-attach the separated prefix to the verb in order to compute the correct verb lemma. Unfortunately, Part-of-Speech taggers (like the TreeTagger) assign the lemma locally and do not consider the long-distance dependency between the verb and the prefix. Hence, we need to correct the verb lemma after PoS tagging. In example 1, the PoS tagger will assign the lemma *weisen* (EN: to point) to the past tense verb form *wies*. Only the re-attachment of the prefix will lead to the correct lemma *nach+weisen* (EN: to prove) and thus to the correct meaning of the verb.

Some annotated corpora of German leave the re-attachment of separated verb prefixes open. For example, the German TIGER treebank marks only the lemma of the finite verb as in figure 1. Since the finite verb and the separated prefix are children of the same mother node S, the prefix can be assigned unambiguously to the verb. Still, this makes querying the treebank for verbs with separable prefixes a complex undertaking. However, recent versions of the TüBa-D/Z treebank do contain verb lemmas with re-attached prefixes (Versley et al., 2010). These lemmas are represented in the same way as the lemmas of the corresponding verbs in unseparated form (e.g. *nach#weisen*).

We work on the annotation of large corpora for linguistic research and information extraction. Therefore we have developed an efficient and robust algorithm to compute the lemmas of German verbs that occur with separated prefixes. In this paper we will present the algorithm. We will then argue that multi-word adverbs cause some confusion to the PoS tagger and thus require special treatment. The correct handling of these adverbs, in return, improves the precision of the re-attached lemmas.

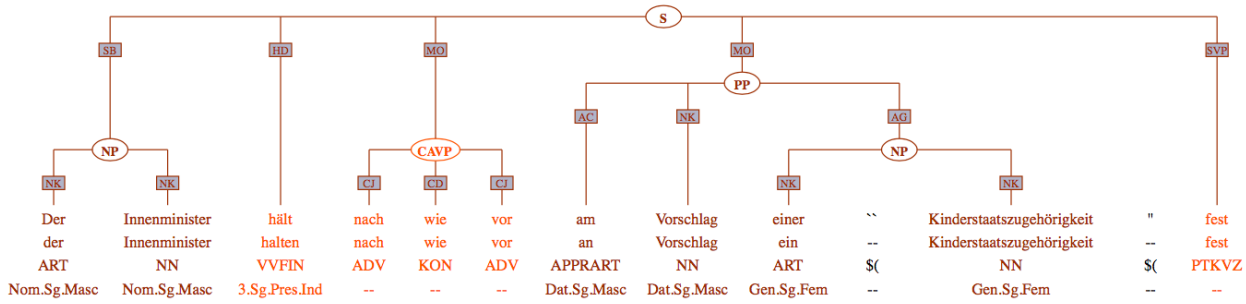

Figure 1: German syntax tree with separated verb prefix (*hält ... fest*) and multi-word adverb (*nach wie vor*) from the TIGER treebank. The multi-word adverb is annotated as coordinated adverbial phrase (CAVP). (English translation: The Interior Minister still maintains the proposal of a children citizenship.)

2 The Re-attachment Algorithm

We re-attach the separated prefix to the verb with the following algorithm. After Part-of-Speech tagging we search for a separated verb prefix (tagged as PTKVZ) and the most recent preceding finite full verb (VVFIN) or imperative verb (VVIMP) in the same sentence. In order to increase the precision we also check whether the re-combined prefix + verb lemma occurs in our corpora and is licensed by the morphology analyzer GerTwol. In this way we have compiled a list of 8500 separable German verbs.

German auxiliary verbs and modal verbs do not take separable prefixes. This means that the auxiliary verbs *haben* (EN: have) and *werden* (EN: become) must be interpreted as full verbs when they take a separable prefix. Consider for instance the verb *innehaben* in *er hat ein Amt inne* (EN: he holds an office). Other examples are *vorhaben* (EN: to intend), *fertigwerden* (EN: to be done with), or *loswerden* (EN: to get rid off).

Similarly, the modal verb *müssen* functions as full verb in combination with the prefix *durch* resulting in *durchmüssen* (EN: to have to go through), and *können* functions as full verb in *wegkönnen* (EN: to be able to leave). Our re-attachment algorithm needs to account for these cases even though state-of-the-art PoS taggers for German label all occurrences of *haben* and *werden* as auxiliary and all occurrences of *müssen* and *können* as modal verbs. Therefore we include PoS correction in the re-attachment of separated verb prefixes for these cases.

This is different from the treatment of these auxiliary and modal verbs in the TüBa-D/Z treebank. The treebank includes the re-attachment of the separated prefixes but leaves the PoS tags unchanged. This means, that in the TüBa-D/Z treebank the verb *innehaben* is a finite full verb, when the prefix is attached, but it is an auxiliary verb, when the prefix is separated. We consider this a misleading inconsistency.

Our re-attachment algorithm leads to high precision re-combined verb lemmas. We first evaluated our method against our corpus of 1.7 million German tokens from banking news (Volk et al., 2016). PoS tagging leads to a total of 9200 tokens marked as separated verb prefixes. Our algorithm re-combines 7630 prefix + verb stems (resulting in 976 types). The re-combined verbs with the highest frequencies are: *ausgehen* (345 occurrences, EN: to go out, to die down), *darstellen* (226, EN: to depict, to represent), *aussehen* (169, EN: to look like, to appear), *stattfinden* (149, EN: to take place), and *beitragen* (136, EN: to contribute). These counts do not include the occurrences of these verbs where the prefix is part of the verb form (i.e. non-separated forms): *ausgehen* (148 occurrences), *darstellen* (216), *aussehen* (106), *stattfinden* (151), and *beitragen* (292).

As a side effect we disambiguate between multiple lemma options. For example, the 3rd person singular verb form *fällt* can have the lemmas *fallen* (EN: to fall) or *fällen* (EN: to fell). The TreeTagger assigns both lemmas to this verb form. If *fällt* occurs with the separated prefix *auf*, then our re-attachment algorithm finds that only the combi-

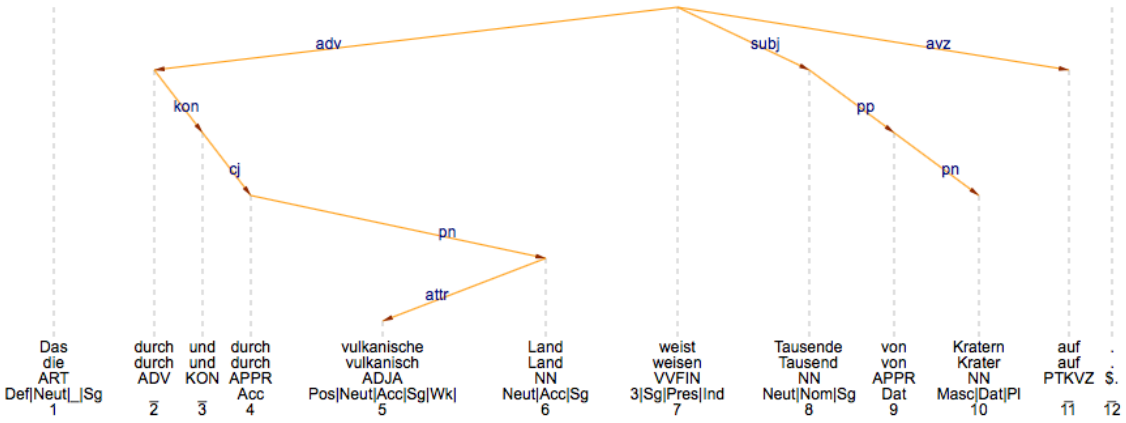

Figure 2: ParZu parser error due to incorrect recognition of the multi-word adverb *durch und durch*. (EN: The totally volcanic land has thousands of craters.)

nation *auffallen* is possible (EN: to stand out, to strike), and we eliminate the other lemma option. Obviously, this disambiguation method is dependent on the separated prefix being only acceptable with one lemma option.

We are aware of three limitations of our algorithm, all of which concern rare cases. First, the re-attachment algorithm will fail for topicalized verb prefixes that precede the finite verb. The TIGER treebank contains 33 examples of separated verb prefixes that precede the finite verb (in roughly 0.9 million tokens of manually annotated newspaper text). The topicalized prefixes in these examples are semantically heavy prefixes (e.g. *Zurück bleibt auch die Erinnerung ...*) and pronominal adverbs (e.g. *Hinzu kommt die Konkurrenz ..., Zugrunde legten die Wiesbadener ...*). We label them as adverbs and pronominal adverbs.

More serious, our re-attachment algorithm will also fail for rare cases of nested finite clauses that occur between the verb and its separated prefix. For example:

(3) Das Konsumwachstum **büsst** im Vergleich zu den vergangenen beiden Jahren, in denen die Wachstumsrate deutlich über 2,0 Prozent **lag**, markant an Schwung **ein**.
(EN: The growth in consumption considerably **loses** momentum in comparison to the past two years, in which the growth rate was clearly above 2 percent.)

This example sentence has a relative clause between the verb *büsst* and its separated prefix *ein*. Since our algorithm assigns the prefix to the most recent finite verb, it will erroneously assign it to the verb *lag* which is the finite verb in the intermediate relative clause. This problem can only be avoided by (at least) a shallow parser which detects the clause boundaries.

Thirdly, our algorithm has no provision for coordinated prefixes.

(4) In einer Deflation **nimmt** der Wert des Geldes **zu** statt **ab**, ... (EN: During a deflation the value of the money increases instead of decreases, ...)

In such examples the verb has basically two different lemmas. We could represent this in the same way as ambiguous lemmas by assigning both lemmas *zunehmen / abnehmen*, but currently this is not part of our implementation.

Other than that, if the PoS tagger recognized all verb forms and all separated prefixes correctly, then our re-attachment algorithm should work perfectly.

We evaluated our algorithm against the TüBa-D/Z treebank. Version 10 of the treebank contains a total of 9181 verb forms that have lemmas with re-attached prefixes. In the standard configuration our program correctly re-attaches 8341 separated prefixes (91%). Most of the remaining cases are verbs missing in our list of possible separable prefix verbs (which we compiled on the basis of our

alpine corpus and our banking corpus). For example *abbürsten, abwiegeln, einigeln* (EN: brush off, play down, curl up into a ball) occur in the TüBa newspaper texts but not in our list. For all these verbs, we let our morphological analyzer decide whether they are German separable prefix verbs. This adds 468 verbs to our list of acceptable separable prefix verbs and boosts the precision of our program to 96.8% re-attachments.

Some of the remaining cases are coordinations with two separated prefixes for the same verb stem (22 occurrences). Another 92 cases (roughly 1%) in the TüBa treebank are separated prefixes that precede the verb. Some are clear cases of separated prefixes (*Denn fest steht: ...; Ziemlich die Post ab geht dagegen bei ...*), many others are debatable on wether they are verb prefixes or adverbs (*Hinzu kommt, daß ...*). Only 145 prefixes (1.5%) are incorrectly attached due to a nested clause.

These numbers are computed based on manually corrected, i.e. perfect PoS tags in the TüBa-D/Z treebank. But in corpus annotation we have to rely on automatically computed PoS tags. Unfortunately, the TreeTagger has problems with the recognition of separated verb prefixes since many of them can also function as prepositions, adverbs and some other word classes. In particular, we noticed errors with the prefix *nach* (EN: after). We manually evaluated all 118 verbs with a re-attached prefix *nach* in our banking news corpus. 41 of these re-attachments (35%) were wrong.

3 Multi-word Adverbs

Closer inspection revealed that in many cases the TreeTagger had erroneously tagged an adverb or a preposition as separated prefix. We found that multi-word adverbs that are created with the coordination pattern "particle *und/wie* particle" (as e.g. *ab und zu, auf und ab, durch und durch, nach und nach, nach wie vor*; see table 1 for glosses and translations) often lead to particles that are mistakenly tagged as separated prefixes (or preposition).[1] We call this special class of multi-word adverbs **bi-particle adverbs** in analogy to the binominals as described by Gereon Müller (1997).

[1] Similar multi-word adverbs in English are *by and large, over and over, to and fro, little by little, side by side*. See also (Müller, 1997) page 3.

Mistagging of these bi-particle adverbs not only disturbs the recognition of verb lemmas but may also lead to erroneous syntax structures as in the parser output in figure 2. There, the second particle in the adverb *durch und durch* is mistagged as preposition which triggers an incorrect dependency of the following noun phrase.

For example, the PoS tagger often assigns the following tags to *nach/PTKVZ wie/KOKOM vor/APPR*, but correctly the tags should be *nach/ADV wie/KOKOM vor/ADV*. Because of these tagging mistakes we observe the following problems in the re-attachment of the separated verb prefix.

(5) Es **gibt** *nach wie vor* im deutschen Erbschafts- und Schenkungsrecht eine Privilegierung für gewerbliche Vermögen.
(EN: There is still a privilege for commercial properties in the German inheritance and donation law.)

In example 5 the TreeTagger marked *nach* as separated prefix which erroneously led to the verb lemma *nachgeben* (EN: to give in) instead of *geben* (EN: to give, there is) which does not have a separated prefix in this sentence.

(6) Schliesslich **stellen** die meisten Luxusgüterfirmen *nach wie vor* den Grossteil ihrer Produkte in Europa **her**, ...
(EN: After all, most luxury merchandise companies still produce the majority of their goods in Europe, ...)

In example 6 the same tagger error leads to the verb lemma *nachstellen* (EN: to imitate) and blocks the re-combination with the true prefix *her* into *herstellen* (EN: to produce).

(7) Wir trauen europäischen Peripherieanleihen **nach wie vor** eine gute Wertentwicklung zu.
(EN: We trust that European peripheral bonds will still have a good value development.)

In example 7 the sanity check correctly blocked the verb lemma **nachtrauen* (which does not exist), but the incorrectly tagged *nach* also blocked the re-combination of the true prefix *zu* to result in *zutrauen* (EN: to dare).

	EN glosses	EN translation	treebank freq	banking news freq	T+B corpus freq
ab und an	from and on	sometimes	3	1	10
ab und zu	from and to	sometimes	1	13	601
auf und ab	up and down	up and down	2	1	310
auf und davon	up and thereof	away	1	-	14
durch und durch	through and through	thoroughly	3	3	89
hin und wieder	to and again	sometimes	1	11	375
nach und nach	after and after	gradually	4	34	702
nach wie vor	after like before	still	62	356	396

Table 1: Multi-word adverbs with particles that also function as prepositions and separable verb prefixes. Frequencies are from the TIGER treebank (890,000 tokens, newspaper texts), from our banking news corpus (1.7 million tokens), and from our Text+Berg corpus (22.5 million German tokens).

In order to identify multi-word adverbs that contain particles which interfere with separated verb prefixes, we searched the German TIGER treebank (890,000 tokens) for coordinated adverb phrases (CADVP). There we found the bi-particle adverbs with verb prefix homographs listed in table 1. The glosses and translations prove that most of them are true multi-words whose meanings are not compositional. They contain particles that can also function as prepositions and separated verb prefixes (*ab, an, auf, durch, hin, nach, vor, zu*). Table 2 gives an overview of their tag frequencies in the treebank.

Note that table 1 is not an exhaustive list but only contains the most frequent bi-particle adverbs in our corpora. Other candidates are *aus und vorbei* (EN: clearly over), *samt und sonders* (EN: completely), *über und über* (EN: over and over).

The most frequent separated prefixes in the TIGER treebank are: *an* (669 times), *aus* (521), *ab* (433), *auf* (405), *vor* (399), *ein* (392), *zu* (244), *zurück* (227) and *mit* (220). The words *ein* and *zurück* cannot function as prepositions. Therefore we disregard them here. *mit* and *zu* are special cases since they can function as adverbs in non-conjunct constructions. *mit* can stand as adverb by itself in the sense of 'jointly' (example: *der die neue CD mit produziert hat*, EN: who has jointly produced the new CD), and *zu* functions as adverb mostly in combination with *bis* (in 121 out of the 127 cases; for example: *bis zu sechs Wochen*, EN: up to six weeks).

Since the frequencies for usages as preposition and separated prefix are much higher than the adverb usage for the particles in question, the PoS tagger is likely to mistake an adverb usage as either a preposition or verb prefix. Therefore we automatically correct the PoS tags of the multi-word adverbs (listed in table 1) in our banking corpus.

In principle, the multi-word adverbs listed in table 1 could also be coordinated prepositions or coordinated separated prefixes, except for the reduplications *durch und durch, nach und nach*. But coordinated separated prefixes are very rare and occur in word plays. Coordinated prepositions are also rare, but they still occur 24 times in the TIGER treebank. Typical examples are *mit und ohne* (EN: with and without), *in und durch* (EN: in and through), and *für und wider* (EN: for and against). It speaks for the idiomaticity of our multi-word adverbs that we have not found a single instance where they are used as coordinated prepositions.

3.1 Bi-particle Adverbs in Text+Berg

We checked how prominent the PoS tagger errors are for the bi-particle adverb *nach wie vor*. Out of 396 occurrences of this candidate in our corpus of alpine texts (the Text+Berg corpus with 22 million tokens in German), we find that *nach* is mistagged as separated prefix in 218 cases (55%), as preposition in 56 cases (14%), and even as postposition 24 times (6%). Only in 25% (98 cases) it is correctly tagged as adverb. Interestingly, in none of these 98 cases, the remainder of the multi-word adverb is correctly tagged. Some tag in this bi-particle

	preposition APPR	sep. prefix PTKVZ	adverb ADV	miscellaneous
ab	77	**433**	9	
an	**2900**	699	6	111 APZR, 1 APPO
auf	**5578**	405	3	2 APZR
aus	**2322**	521	4	65 APZR, 1 APPO
durch	**1277**	37	9	1 APPO
hin	**-**	79	63	7 APZR
mit	**6039**	220	21	
nach	**2612**	54	71	32 APPO, 1 APZR
vor	**1814**	399	67	
zu	2084	244	127	**4413 PTKZU**, 277 PTKA

Table 2: Part of Speech tag frequencies in the TIGER treebank for particles that occur in multi-word adverbs (lower case usage only). Miscellaneous PoS tags include postposition (APPO), right element of circumposition (APZR), infinitive marker (PTKZU), and adjective modifier (PTKA).

adverb is always wrong. This is clear evidence that only a special treatment or a completely different PoS tagging approach for multi-word adverbs will lead to high quality PoS tags.

Occasionally the bi-particle candidates are not multi-word adverbs. In example 8, the candidate is really a sequence of the adverb *ab* and the preposition *zu*. This is very rare. In 200 occurrences of *ab und zu* in our Text+Berg corpus we found one such occurrence.

(8) ... führen von der ursprünglich appenzellischen Weise **ab** und **zu** den Rhythmen eines ganz fremden Volkes. (EN: ... lead away from the traditional Appenzell customs and to the rhythms of a totally foreign people.)

This problem is more prominent with the candidate *ab und an* (see example 9). It occurs only 10 times in our Text+Berg corpus, but 5 of these are non-adverb cases (all predating 1925).

(9) Die Haare standen von den Köpfen **ab** und **an** der Stirne, wo das seidene Band um die Hüte ... (EN: The hair stood off from the heads and on the forehead where the silk braid around the hats ...)

Text+Berg which is a corpus with texts from the last 150 years also leads to multi-word adverbs which were prominent in the past but are no longer used, as for example **je und je** (attested 23 times from 1868 to 1958) in the meaning *always*.

(10) Von nah und fern, von dies- und jenseits des Alpengebirges sind **je und je** Geologen und Mineralogen ins Tessin gewandert, ... (EN: From near and far, from both sides of the Alpes geologists and mineralogists have always migrated to the Tissino, ...)

This multi-word adverb has been superseded by *eh und je* (EN: always) which is attested in our Text+Berg corpus 40 times since 1940.

It is striking that *nach wie vor* is the most frequent bi-particle adverb both in the TIGER treebank and in our Credit Suisse news corpus whereas *nach und nach* is the clear top frequency adverb in our Text+Berg corpus. A closer inspection revealed that this is due to the fact that the Text+Berg corpus is a collection that spans 150 years whereas the TIGER treebank and the Credit Suisse news corpus has only texts from the last 20 years. Google n-gram viewer shows that *nach wie vor* is on the upswing in recent decades whereas *nach und nach* has lost popularity during the same period (cf. figure 3).

3.2 Bi-particle Adverbs Overview

The above section on bi-particle adverbs exemplifies that many adverbs of this kind are true multi-word expressions (with non-compositional seman-

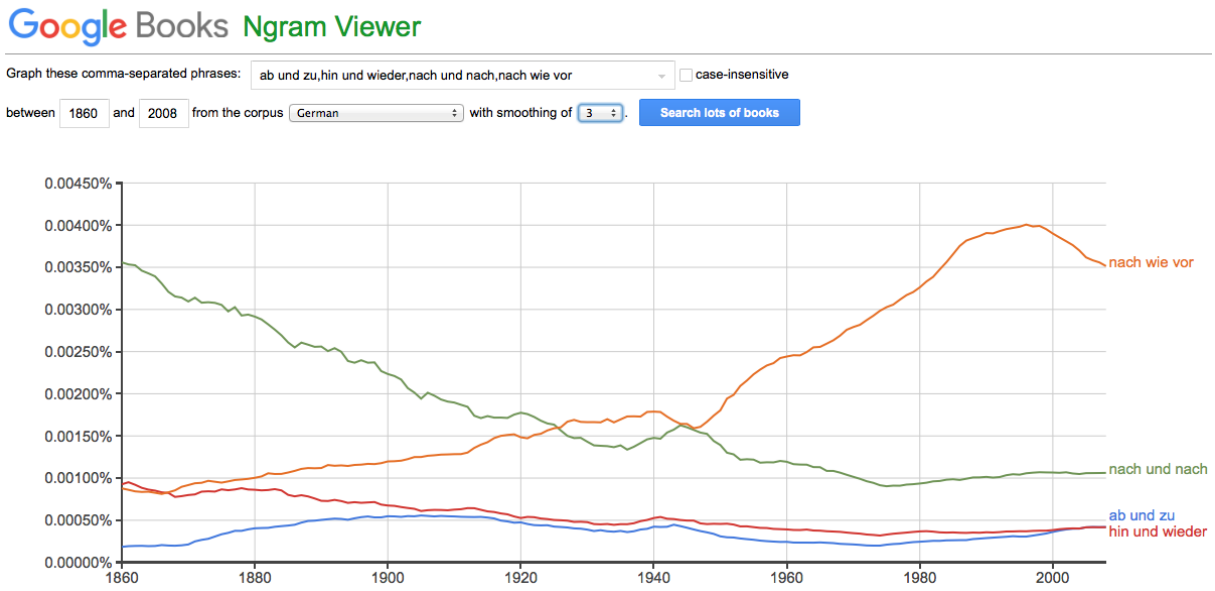

Figure 3: Google n-gram statistics showing the frequency development of four bi-particle adverbs over the last 150 years.

tics) that need special treatment in natural language processing. In order to detect the whole range of these adverbs we computed collocation scores for all patterns with words that are tagged as non-inflected adjectives (ADJD), adverbs (ADV, PAV, PWAV), prepositions (APPR), and separated prefixes (PTKVZ) in coordinated constructions with the conjunctions *als, oder, und, wie* (KON, KOKOM).

In this way we found fixed expressions like *fix und fertig; klipp und klar* (EN: wiped out; concisely) at the top of the list, but also pairs that stress opposition *hüben und drüben* (EN: here and there) or reinforce the meaning through synonym repetition *nie und nimmer* (EN: never ever) or reduplication *dunkler und dunkler* (EN: darker and darker)[2]. In addition we find pairs that form idiomatic adverbials within larger expressions *(über) kurz oder lang; (mehr/eher) schlecht als recht* (EN: sooner or later; badly). We also noted that bi-particle adverbs may also involve truncated words (tagged as TRUNC) as first conjunct as in *niet- und nagelfest, sang- und klanglos* (EN: nailed down; quietly).

In conclusion, bi-particle adverbs are an understudied category among multi-word expressions which deserves a lot more attention. These adverbs cover the whole spectrum of idiomaticity and can only be interpreted correctly when their collocation strength is appropriately considered.

4 Evaluation of Bi-particle Adverb Recognition on Separable Prefix Verbs

After automatic correction of the PoS tags in the bi-particle adverbs in table 1 we observe improved precision in the re-attachment of separated verb prefixes with 7600 prefix + verb combinations. We manually checked the re-attached prefix *nach* and found 79 cases with 1 error left. This error is due to a missed sentence boundary and a PoS error in a sentence-initial verb. Overall, we observe 47 removed prefix-verb combinations and 16 new prefix-verb combinations. All these changes are correct.

Recall of the re-attachment of separable verb prefixes is more difficult to determine. We see that there are still 1388 particles that are tagged as separated verb prefixes which we were unable to re-attach. We find 590 cases with a combination of prefix + verb which is not licensed through our list of separable prefix verbs, and 798 separated prefixes for which we do not find a full verb in the sentence. Most of these cases are PoS tagging

[2]English also features reduplications in adverbs: *again and again, more and more, neck and neck.*

errors either of the particle or the verb. For example, we have seen some PoS errors where the finite verb is mistakenly tagged as infinitive (the 1st and 3rd plural present tense forms of German verbs are homographic with the infinitive). The presence of a separated prefix indicates that the verb must be finite, and we could use that information to correct the verb's PoS tag if we trust the prefix tag more than the verb tag. This is currently not implemented.

For the unattachable words that are tagged as separated prefixes we found it to be advantageous to automatically correct their PoS tag to adverbs (ADV) for a list of 32 possible prefixes which often function as adverbs such as *empor, nahe, vorbei* (EN: upward, near, past). This correction step solves about half the cases where the PoS tagger assigned the tag "separated prefix" (PTKVZ) but we were unable to re-attach the word to a verb.

5 Related work

Lüdeling (2001) presents an in-depth study of the linguistic and corpus linguistic properties of German particle verbs. Stefan Müller (1999) discusses how to integrate German particle verbs into a comprehensive HPSG grammar whereas Forst et al. (2010) discuss the same for large LFGs. For both grammars it is unclear to what extent they could be used to annotate large corpora.

Hoppermann and Hinrichs (2014) introduce an approach to model particle verbs in their large German WordNet. Versley et al. (2010) have developed an approach for lemma disambiguation in German to serve the TüBa-D/Z treebank. In a recent publication Dewell (2015) investigates the semantics of selected German verb prefixes, both separable and inseparable ones.

Nießen and Ney (2000) report on early experiments to prepend German prefixes to the verbs for statistical machine translation into English. 14 years later Schottmüller (2014) still deals with separated verb prefixes in MT for the same language pair. She suggests to substitute German prefix verbs with synonymous inseparable verbs (e.g. substitute *fängt ... an* with *beginnt* (EN: to begin)) in order to improve translation quality. She demonstrates that current MT systems like Google Translate and Bing Translator still have problems with separated verb prefixes and produce better translations for sentences with synonymous non-separable verbs.

Related to our approach of the annotation of German prefix verbs is (Bott and Schulte im Walde, 2015) who present features to predict the compositionally of German particle verbs. Also similar is (Fritzinger, 2010) who uses parallel texts to detect German verb + prepositional phrase MWEs via automatic word alignment.

However, to the best of our knowledge, there is no literature on the interdependence between the recognition of multi-word adverbs and the analysis of separable prefix verbs. There is also no repository of German multi-word adverbs (unlike in French (Laporte and Voyatzi, 2008) and some other languages).

(Nagy and Vincze, 2014) present a method for the detection of verb-particle constructions in English (e.g. *to eat up, to take off*). They argue that a parser should be trained on a data set that includes specific annotation for verb-particle constructions.

Gereon Müller (1997) presents a detailed study of binomial constructions in German (e.g. *Fug und Recht, samt und sonders*) which includes bi-particle adverbs. He is particularly interested in order constraints (e.g. **Recht und Fug, *sonders und samt*) of the constructions. These constraints also hold for the bi-particle adverbs: **vor wie nach, *wieder und hin, *zu und ab* are not possible. Müller also offers a four level system of semantic opaqueness which would see the bi-particle adverb *hin und wieder* in class 1 (meaning is not compositional) and *auf und ab* in class 4 (meaning is compositional, but ordering constraints hold). He elaborates that end rhyme, alliteration (*ab und an*) and assonances (the repetition of vowel sounds to create internal rhyming) are typical properties of binominal constructions.

6 Conclusion

We have introduced an efficient algorithm for the computation of full lemmas for German verbs with separated prefixes. Checking the algorithm against the relevant verbs in the TüBA-D/Z treebank revealed an accuracy of 96.8%.

We have shown that the correct identification and PoS tagging of German bi-particle adverbs increases the accuracy of the re-attachment of separated prefixes to verb lemmas. Furthermore it

improves the interpretation and analysis of the sentences, both for the multi-word adverbs and the verbs. We also believe that the correct identification of multi-word adverbs and prefix verbs will improve cross-lingual word alignment and subsequently machine translation. This will be our next area of investigation.

Acknowledgments

This research was supported by the Swiss National Science Foundation under grant 105215_146781 for "SPARCLING: Large Scale PARallel Corpora for LINGuistic Investigation" (2013-2017) a joint project with Marianne Hundt and Elena Callegaro at the English Department of the University of Zurich. A first version of this work was presented at the PARSEME COST Action meeting in Struga, Macedonia in March 2016 with support by the European Union. We also thank the anonymous reviewers for helpful comments on an earlier version of this paper.

References

Stefan Bott and Sabine Schulte im Walde. 2015. Exploiting Fine-grained Syntactic Transfer Features to Predict the Compositionality of German Particle Verbs. In *Proceedings of the 11th Conference on Computational Semantics*, pages 34–39, London.

Robert B. Dewell. 2015. *The Semantics of German Verb Prefixes*, volume 49 of *Human Cognitive Processing*. John Benjamins.

Martin Forst, Tracy Holloway King, and Tibor Laczkó. 2010. Particle verbs in computational LFGs: Issues from English, German, and Hungarian. In *Proceedings of the LFG10 Conference*, pages 228–248. CSLI Publications.

Fabienne Fritzinger. 2010. Using parallel text for the extraction of German multiword expressions. *Lexis. E-Journal in English Lexicology*, pages 23–40, April.

Christina Hoppermann and Erhard Hinrichs. 2014. Modeling prefix and particle verbs in GermaNet. In Heili Orav, Christiane Fellbaum, and Piek Vossen, editors, *Proceedings of the Seventh Global Wordnet Conference*, pages 49–54, Tartu, Estonia.

Eric Laporte and Stavroula Voyatzi. 2008. An electronic dictionary of French multiword adverbs. In *Proc. of LREC*, Marrakech, Morocco.

Anke Lüdeling. 2001. *On Particle Verbs and Similar Constructions in German*. CSLI, Stanford.

Gereon Müller. 1997. Beschränkungen für Binomialbildungen im Deutschen. *Zeitschrift für Sprachwissenschaft*, 16(1):25–51.

Stefan Müller. 1999. Syntactic properties of German particle verbs. In *Sixth International Conference on HPSG–Abstracts. 04–06 August 1999*, pages 83–88, Edinburgh.

István Nagy and Veronika Vincze. 2014. VPCTagger: Detecting verb-particle constructions with syntax-based methods. In *Proceedings of Workshop on Multiword Expressions. EACL*, Göteborg.

Sonja Nießen and Hermann Ney. 2000. Improving SMT quality with morpho-syntactic analysis. In *Proc. of COLING*, pages 1081–1085, Saarbrücken.

Nina Schottmüller. 2014. Issues in translating verb-particle constructions from German to English. In *Proceedings of Workshop on Multiword Expressions*, Gothenburg.

Yannick Versley, Kathrin Beck, Erhard Hinrichs, and Heike Telljohann. 2010. A syntax-first approach to high-quality morphological analysis and lemma disambiguation for the TüBa-D/Z treebank. In *Proceedings of the Ninth International Workshop on Treebanks and Linguistic Theories*, pages 233–244, Tartu, Estonia.

Martin Volk, Chantal Amrhein, Noëmi Aepli, Mathias Müller, and Phillip Ströbel. 2016. Building a parallel corpus on the world's oldest banking magazine. In *Proceedings of KONVENS*, Bochum.

TweetNorm: Text Normalization on Italian Twitter Data

Daniel Weber
CIS, LMU Munich, Germany
weber.daniel@campus.lmu.de

Desislava Zhekova
CIS, LMU Munich, Germany
desi@cis.uni-muenchen.de

Abstract

This paper addresses the issue of text normalization on non-standard Italian data. We present TweetNorm[1], a system which normalizes Italian tweets in a way that the amount of microblog slang and distorted text appearance is drastically reduced and the normalized output has a much cleaner and more formal style. The paper shows that with a set of fixed language-independent rules and trained rules for language-dependent abbreviation and acronym expansion good results can be achieved for normalizing Italian Twitter messages.

1 Introduction

In general, the process of normalization of non-standard data is often a necessary preprocessing step to enable natural language processing (NLP) tools which require clean and standardized data as input to perform on their expected quality levels. Differences in the performance of NLP systems when using non-standard data instead of standard data have been early proven with the conclusion that the performance can have a decrease of up to 50% (Poibeau and Kosseim, 2001).

We focused on informal texts from a social media platform, Twitter, which holds massive amounts of user-generated data. Informal text is produced on this platform because the Twitter user's writing style changes over time and is influenced by other users and the conditions of the social service. For example, a condition on Twitter which limits the length of a tweet to 140 characters frequently leads to increased use of abbreviations of common or long words in order not to exceed the size limitation. Previous research on Twitter text normalization is mainly language-dependent and does not cover many different languages, even if generally the methods can be adapted to new languages (often with a large amount of manual effort). So far, there is no work for this task on Italian.

[1] http://www.cis.uni-muenchen.de/desi/tydiady/tweetNorm.zip

The primary goal of our work is to explore text normalization for Italian microblogs and to propose an approach that is as language-independent as possible. By using mainly general and language-independent normalization methods we show that our approach may easily be transferred to other similar, under-resourced languages.

This paper is organized as follows: After reviewing relevant related work in section 2 we describe the normalization system TweetNorm in section 3. Section 4 shows resources used by TweetNorm and provides information about the used Twitter corpus. In section 5, we evaluate the performance of TweetNorm. Section 6 concludes the paper and discusses future work.

2 Related Work

Text normalization on English data e.g. (Han and Baldwin, 2011; Li and Liu, 2012; Xue et al., 2011) is well researched compared to other languages. Especially work on microtext normalization of English and Spanish tweets was contributed as part of two workshops: W-NUT (Baldwin et al., 2015) and TWEET-NORM (Alegria et al., 2013). Most of the text normalization approaches focused on one language, mainly English. Other languages can not directly profit from these approaches as it is too time-consuming to adopt them accordingly.

The multilingual text normalization approach by Bigi (2011) which splits the normalization problem in a set of sub-problems as language-independent as possible targeted only French, English, Spanish, Vietnamese, Khmer and Chinese. Our work also follows the main idea of this approach and splits the normalization problem into various sub-problems where the language independent parts follow previous research on English Tweet normalization (Baldwin et al., 2015) based on the efficient and often used OOV (out of vocabulary) technique (Han and Baldwin, 2011) which comes naturally to mind when only processing unknown tokens.

3 Normalization via TweetNorm

This normalization process consists of three main steps. The first step identifies normalization candidates based on a standard vocabulary and tags tokens which are out of this vocabulary (OOV). The next step normalizes punctuation and OOV-tokens with regard to word contortions (spelling errors), Twitter-tags and removes OOV-tokens which are unnecessary for standardized text. All these methods are generally language-independent. To make some of these methods work in other languages too, one only needs language specific resources, such as a standardized vocabulary. Only the final step is language dependent as it normalizes OOV-tokens concerning abbreviations and acronyms.

3.1 Language-independent System Components

Multiple Character Normalization (MCN)

One part of the first normalization step targets tokens which contain repetitive characters because an author often tries to express its emotion by stretching words or punctuation marks with multiple characters. This was also recognized, described and used by Akhtar et al. (2015). All letters which appear more than two times in a row are reduced to only one occurrence. A subsequent spell checker prevents possible errors due to the reduction (*Capppuccino* → *Capuccino* → *Cappuccino*). Multiple punctuation marks are normalized similarly. There are some special cases: For example, a cascade of multiple exclamation marks and interrogation marks is always normalized to one interrogation mark. Three and more dots are always set to three dots in order not to change the meaning of the sentence.

Non-word Token Removal

The term non-word token summarizes emoticons, hyperlinks, random letter sequences, HTML markup and other special uses. There are two types of emoticon detection – a smiley gazetteer and generic smiley detection rules. In comparison to the system created by Porta and Sancho (2013) which normalizes for example *:DDDD* or *xDDDDD* to canonical emoticon forms, our system removes all detected emoticons because we only considered normalized representations for words. Hyperlinks and HTML markup allow high precision detection and also high precision adaption rules. Based on the fact that a high number of letters per token in relation to a low number of different letters typically does not represent a standard word, random letter sequences (*dfgdfgdkldkglfd*) are removed by a letter frequency algorithm, which calculates the ratio between the length of the token and single letter frequencies. It is considered to be language-independent because this algorithm does not need any training data and long words for example with a length of 16 while containing only 3 different characters are very unusal for the most Indo-European languages (comparable example: senselessnesses length:15, letters:4). The threshold of the algorithm which decides if a token is seen as random may need to be set higher for languages which do not meet these criteria because their alphabet may be smaller or more exceptions exists in this language.

In addition, a bigram language model trained on the standard vocabulary is used to identify tokens which do not represent words in this language (*iruhgcsmiegh*). The sum of letter transition probabilities in a token controls the decision process. All tokens with a low probability to be part of the language described by the model are removed by supporting methods. This is not considered as language-dependent extra work because a standard vocabulary is mandatory for each language TweetNorm is applied. Further, more than three space-separated single (capital) letters in a row are joined to detect and normalize regular words or abbreviations. Additional rules which are used to define tokens reflecting mood states which are not part of the standardized language rely on filtering and observations of the training data and may need discrete investigations for each new language. These rules include for example characteristic multiple letter tuples (*xaxaxax*, *lalala*).

Spelling Correction and String Decomposition

In order to correct words which are spelled wrong, a spelling correction was integrated, following Jin (2015) to measure similarity between two strings. The similarity of two strings is calculated with the Jaccard Index (Levandowsky and Winter, 1971) by comparing differently weighted similarity feature sets which are extracted from both strings. The vocabulary word with the highest similarity score is chosen as the correct version. Computational cost is reduced by only considering the top 150,000 frequent words with precalculated similarity feature sets for each word. In addition, only words from the tweet which have a Levenshtein distance (Levenshtein, 1966) less than three or a ratio greater

than 0.8 are considered as candidates.

To recover missing spaces between words, a string decomposing method was introduced. The method starts at the end of a long token and scans consecutive character by character for the longest match with minimal three letters. Tokens are only decomposed in case each portion of the split token represents a known word. The following example will result in two splits *Questograndeesempio* → *Questo grande esempio* while *Questo**speca**esempio* will remain without any splits because there is no proper split for *Questospeca* concerning only known words. The spelling correction was not combined with the string decomposition method in order not to accidently change the original meaning, because this raises possible splits dramatically if short tokens may be extended or potentially illformed words were corrected. However, the spelling correction was applied on hashtag splits made from capitalization patterns because the tweet author already signalized intended words with uppercase letters which reduced the number of possible splits.

Twitter Tag Normalization

Processing Twitter tags is divided in two main operations – removal and normalization. Based on the position of the tag in a tweet reliable decisions can be met. Tags which appear within the span of a tweet are usually part of the sentence structure and therefore function as a syntactic or semantic element. Starting or closing tags mostly only act as Twitter functions (user address, topic labeling) and their absence does not harm the grammar or sense of the sentence. Tags starting with an @ are resolved to personal names (*@usernameX4* → *Frank Jones*). The first level username alteration uses a dictionary of Twitter usernames mapped with its corresponding cleaned personal names. This dictionary is composed of 18 thousand name pairs seen in the training data and is extended by three top 1,000 Twitter user ranking lists. The users are ranked according to their respective number of followers, following and count of tweets.[2] The optional second level alteration rests upon live profile queries to extract, clean and save personal names. In case none of the previous layers could resolve the username, extra rules try to split the username in capitalized letter chunks (*LauraCaselli123* → *Laura Caselli*).

[2]TwitterCounter. `http://twittercounter.com`

A hashtag followed by a punctuation mark which indicates sentence boundary is always normalized and never removed because it is likely that such a hashtag might be a key element in this sentence. In the following example the search engine *Volunia* is a key element of the tweet "*why don't you switch to #volunia**?** :)*" because the hashtag cannot be removed without loosing important information. Therefore the hashtag must remain and the tweet is normalized to "*... switch to volunia?*". In this case the word *to* also signalizes that the following hashtag is embedded in the sentence and cannot be removed. These indicators can be used by taking the local word context of a hashtag into consideration. For this reason, the context of hashtags is scanned by a list of 600 Italian stop words and verbs for articles, conjunctions, prepositions and specific words which correlate syntactically or semantically with the hashtag. Based on context matches, a removal or normalization action is undertaken. The string decomposition method is also applied to hashtags in order to restore their standardized space-separated form (*#exampletopic* → *example topic*). Plenty hashtags stick to the Twitter recommendation that each new word should start with an uppercase letter. As a result if the hashtag contains a minimum of two uppercase letters it is split on capitalized letter chunks (*#FridayNight* → *friday night*). There is a possibility to feed the spelling correction with OOV chunks, but this is disabled by default because typos in hashtags rarely appeared while running the system. Almost all observed OOV chunks are named entities (organisations or names) like *Pinterest, Sgommati, Driih, Taynara* and the probability to mistakenly correct a named entity to a similiar spelled Italian word was estimated as to high with respect to the low number of necessary corrections and truely corrected words.

3.2 Language-dependent Components

Preprocessing Data

Collection and preprocessing of resources must be done for each language. Parts of the preprocessing steps are automatable but in order to achieve clean data for a new language, human work is obligatory. The preprocessing step entails the biggest effort to patch TweetNorm to a new language. The performance of TweetNorm heavily relies on the quality of the resources therefore the methods themselves do not need any patches, except special language

specific adaptions.

Abbreviation and Acronym Normalization

Each already normalized tweet was POS-tag annotated by the TreeTagger (Schmid, 1999). We used the TreeTagger since we consider the normalized tweets to be very close to standard Italian for which the TreeTagger has been originally trained. Moreover, while good POS taggers for tweets are available for English, this is not the case for Italian. All entries out of the abbreviation collection (section 4) with likely token and POS-tag context information which was extracted from training data (section 4) are initially replaced in the annotated tweet regardless whether the context matched or not. All replaced short forms in which neither the token context nor the POS-tag context matched are flagged as unsafe replacements. During a second POS-tag annotation run an algorithm decides in case of a significant increase of the context POS-tag probability compared to the first run, a POS-tag match of the full form and partial matches in previous and posterior contexts whether the unsafe replacement will be reverted or not. Entries which have no context information are seen as rare and thus they are always replaced by their unambiguous full form. Short forms which require certain conditions and patterns regarding the context like numbers or specific tokens are only replaced if all conditions are fulfilled.

4 Data Acquisition and Preparation

Standardized Vocabulary

A vocabulary which defines words that can be seen as standard is essential in this normalization approach. A good coverage of words which can be seen as standardized allow a better detection rate of normalization candidates. The vocabulary is compiled from different sources with different granularities. For further details see appendix A. All frequency lists together include more than 9,180,000 tokens. After removing smilies, punctuations, dates, numbers, links, misspelled words and other non-standard tokens plus excluding abbreviations the size of the vocabulary was reduced to 4,510,000 entries. TweetNorm supports additional user created lists (containing e.g. named entities or rare domain specific words) which can be treated as whitelists for the system to prevent unwanted token modifications. This is for example relevant if the normalization acts as preprocessing and the normalized text will later be applied to keyword sensitive applications.

Abbreviation and Acronym Collection

A collection of abbreviations and acronyms with their associated full form provides the basis for abbreviation expansion. Further information regarding the sources can be found in appendix B. The collection consists of about 400 abbreviations. Each entry was expanded by its most likely bigram contexts of tokens and bigram contexts of part-of-speech tags based on the full 9 GB POS-tagged Paisà corpus (Lyding et al., 2014). Additional acronyms were extracted from the Twitter corpus by searching for acronyms defined or mentioned within a tweet by the author. One method to find abbreviation definitions scans for keywords and punctuation which indicate mentioned abbreviations within the local context. Another context-free extraction method shrinks a tweet only to the lowercased leading letter of each token and matches sub-sequences from the original tweet:

"*This is a **s**mall **e**xample called **se**!*" → *tia**se**c*!

Twitter Corpus

The data used in this project is mainly self-procured. Periodically crawled microblogs via the Twitter REST API[3] form the biggest part of the corpus consisting of 100 thousand tweets, obtained from September to November 2015. The tweets were crawled by querying messages which contain words out of a predefined most frequent Italian word list which do not occur in any other language. All matches were post-processed with LangID (Lui and Baldwin, 2012) to assure that the messages are Italian only.

Due to limitations with Twitter's free API also mentioned by Weller et al. (2013) relating to accessibility and availability of the Twitter messages beyond a certain time frame it is hard to achieve a diversified Twitter corpus in a fast and efficient way. However, to build a corpus which is not limited to a certain time frame parts of the corpus rely on previous work done by Basile and Nissim (2013) and Basile et al. (2014). In this way, the crawled Twitter corpus was enriched with 75 thousand Italian Twitter messages from different months in 2012 and 2013 obtained by their tweet-ID which was provided by the SENTIPOLC (Basile et al., 2014) and the TWITA corpus (Basile and Nissim, 2013).

[3]Twitter. REST API. `https://dev.twitter.com`

5 Results

Unfortunately, no gold standard dataset was available for this language and task. Thus, we manually evaluated and analysed the system performance on one hundred random tweets from a set aside. Due to lack of time this set is very small and we will approach its extension as soon as possible. For each tweet, all changes done by the normalization system were manually validated. The three main categories of system applied operations cover deletion, transformation and insertion. The correctness of each operation was controlled and the resulting F_1-Score for each operation can be seen in table 1. The

Operation	Accuracy	Precision	Recall	F1
Deletion	98.42	97.31	90.95	94.02
Transformation	98.82	93.04	92.24	92.64
Insertion	99.93	97.82	100.00	98.90
Tokenization	99.37	95.30	98.61	96.69
Total				**95.56**

Table 1: Evaluation of TweetNorm operations.

transformation operation has the lowest F_1-Score with 92.64, but this operation also contains the most complex normalization methods like spelling correction, Twitter tag and abbreviation normalization. The parameters of the spelling correction are set to perform safe transformations in order to maximize precision, but there are still transformation errors. For example neologism like the portmanteau word "*twittatore*" (probably a blend of "***twitt**er*" and "***d**ittatore*") is normalized to "*dittatore*", because the morphological overlap of the involved words is too high for the parameters' sensitivity. Besides this, current errors done by the normalization of Italian abbreviations comprise incorrect expansions of unknown ambiguities like the expansion of "*San val*" to "*San valuta*" instead of "*San Valentino*". Compared to the success of the other normalization methods it still could be improved with more training data or with more specific Italian grammar knowledge and abbreviation creation rules.

After solely applying deletion operations (F_1-Score of 94.02), which includes the removal of smilies, emotionalized tokens, hyperlinks, non-semantic Twitter tags and other non-standard tokens a tweet looks much more structured and is far more readable. In addition, operations like the MCN which achieves excellent results also contribute a big part in improving the readability of a tweet. On an average this method produces 2,400 operations per 20,000 tweets out of a total of 45,000 normalization operation of all methods altogether. The manual validation indicated that the normalization of Twitter mention tags and hashtags is very robust and yields reliable output. Appendix C shows example tweets normalized by TweetNorm.

In conclusion the system performs a suitable normalization with a total F_1-Score of 95.56 on Italian tweets and the output is very similar to handmade changes.

6 Conclusion and Future Work

In this work, an approach to normalize Italian tweets according to their non-standard nature was presented which showed that is possible to achieve clean and accurate outputs with a set of language-independent rules with partial language-specific shapes relying mainly on structured resources while keeping the system itself portable and adaptable to other similar, under-resourced languages.

In order to further increase the usability and customization of TweetNorm it is planed to extend the modular design to allow easy normalization method combinations to fit individual task needs. For example turning off normalization of multiple punctuation marks for opinion mining, because sentence boundaries often indicate strong opinions. In the future this system may be used to generate training data for a statistical machine translation system (SMTS) like Moses[4] which might enhance the normalization process. The normalization task can then be seen as a machine translation problem which processes a parallel corpus of non-standard tweets and normalized tweets to extract generalized normalization rules.

References

Md Shad Akhtar, Utpal Kumar Sikdar, and Asif Ekbal. 2015. IITP: Hybrid Approach for Text Normalization in Twitter. *ACL-IJCNLP 2015*, page 106.

Iaki Alegria, Nora Aranberri, Víctor Fresno, Pablo Gamallo, Lluis Padró, Iaki San Vicente, Jordi Turmo, and Arkaitz Zubiaga. 2013. Introducción a la tarea compartida Tweet-Norm 2013: Normalización léxica de tuits en espaol. In *Workshop on Tweet Normalization at SEPLN (Tweet-Norm)*, pages 36–45.

Timothy Baldwin, Young-Bum Kim, Marie Catherine de Marneffe, Alan Ritter, Bo Han, and Wei Xu. 2015. Shared tasks of the 2015 workshop on noisy user-generated text: Twitter lexical normalization

[4]Moses. `http://www.statmt.org/moses/`

and named entity recognition. *ACL-IJCNLP 2015*, page 126.

Marco Baroni, Silvia Bernardini, Adriano Ferraresi, and Eros Zanchetta. 2009. The wacky wide web: a collection of very large linguistically processed web-crawled corpora. *Language resources and evaluation*, 43(3):209–226.

Valerio Basile and Malvina Nissim. 2013. Sentiment analysis on italian tweets. In *Proceedings of the 4th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 100–107, Atlanta.

Valerio Basile, Andrea Bolioli, Malvina Nissim, Viviana Patti, and Paolo Rosso. 2014. Overview of the evalita 2014 sentiment polarity classification task. *Proceedings of the 4th evaluation campaign of Natural Language Processing and Speech tools for Italian (EVALITA14). Pisa, Italy*.

Brigitte Bigi. 2011. A multilingual text normalization approach. In *5th Language & Technology Conference-The 2nd LRL WORKSHOP*, pages 1–5.

Bo Han and Timothy Baldwin. 2011. Lexical normalisation of short text messages: Makn sens a #twitter. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 368–378, Portland, Oregon, USA, June. Association for Computational Linguistics.

Ning Jin. 2015. Ncsu-sas-ning: Candidate generation and feature engineering for supervised lexical normalization. *ACL-IJCNLP 2015*, page 87.

Michael Levandowsky and David Winter. 1971. Distance between sets. *Nature*, 234(5323):34–35.

VI Levenshtein. 1966. Binary Codes Capable of Correcting Deletions, Insertions and Reversals. *Soviet Physics Doklady*, 10:707.

Chen Li and Yang Liu. 2012. Normalization of text messages using character-and phone-based machine translation approaches. In *INTERSPEECH*, pages 2330–2333.

Marco Lui and Timothy Baldwin. 2012. Langid.py: An off-the-shelf language identification tool. In *Proceedings of the ACL 2012 System Demonstrations*, ACL '12, pages 25–30, Stroudsburg, PA, USA. Association for Computational Linguistics.

Verena Lyding, Egon Stemle, Claudia Borghetti, Marco Brunello, Sara Castagnoli, Felice DellOrletta, Henrik Dittmann, Alessandro Lenci, and Vito Pirrelli. 2014. The paisa corpus of italian web texts. In *Proceedings of the 9th Web as Corpus Workshop (WaC-9)*, pages 36–43.

Thierry Poibeau and Leila Kosseim. 2001. Proper name extraction from non-journalistic texts. In *In Computational Linguistics in the Netherlands*, pages 144–157.

Jordi Porta and José-Luis Sancho. 2013. Word Normalization in Twitter Using Finite-state Transducers. *Tweet-Norm@ SEPLN*, 1086, pages 49–53. Citeseer.

Helmut Schmid. 1999. Improvements in part-of-speech tagging with an applicationto german. In Susan Armstrong, Kenneth Church, Pierre Isabelle, Sandra Manzi, Evelyne Tzoukermann, and David Yarowsky, editors, *Natural Language Processing Using Very Large Corpora*, volume 11 of *Text, Speech and Language Processing*, pages 13–26. Kluwer Academic Publishers, Dordrecht.

Katrin Weller, Axel Bruns, Jean Burgess, Merja Mahrt, and Cornelius Puschmann. 2013. *Twitter and society*, volume 89. Peter Lang New York.

Zhenzhen Xue, Dawei Yin, and Brian D Davison. 2011. Normalizing microtext. In *Proceedings of the AAAI-11 Workshop on Analyzing Microtext*, pages 74–79.

Eros Zanchetta and Marco Baroni. 2005. Morph-it! a free corpus-based morphological resource for the italian language. *Corpus Linguistics 2005*, 1(1).

Appendices

A Vocabulary Resources

- **OpenSubtitle:** An italian frequency list build from "OpenSubtitle" corpus.[5,6]
- **Morph-it!:** A morphological resource with 31,955 Italian lemmas and 506.827 word forms (Zanchetta and Baroni, 2005).
- **itWaC:** A frequency list extracted from the "itWaC" corpus, which contains 2 billions tokens crawled from Italian websites (Baroni et al., 2009).
- **Paisà:** A frequency list based on Paisà corpus which holds 250 millions words extracted from Italian internet documents (Lyding et al., 2014).
- **ItWikiArticles:** Self-produced frequency list of all Italian Wikipedia articles up to and including september 2015.[7]

[5] OPUS. The open parallel corpus. `http://opus.lingfil.uu.se/`

[6] Invoke IT Blog. Frequency Word Lists. `https://invokeit.wordpress.com/frequency-word-lists/`

[7] Wikimedia. Italian Wikidump progress on 20151002. `https://dumps.wikimedia.org/itwiki/20151002/`

B Abbreviation Resources

- Nicla A. Gargano, Corsi ditaliani. Abbreviazioni.
  `http://homes.chass.utoronto.ca/~ngargano/corsi/corrisp/abbreviazioni.html`
- Dr. Ulrich Hondelmann, Italianita. Italian Abbreviations.
  `http://www.italianita.de/files/italienische-abkuerzungen.htm`
- PONS. Italian-German A-Z.
  `http://de.pons.com/bersetzung/italienisch-deutsch/-/A`
- Abbreviations, STANDS4 Network. Italian Abbreviations.
  `http://www.abbreviations.com/acronyms/ITALIAN`
- Michael San Filippo, About Education. Italian Abbreviations and Acronyms.
  `http://italian.about.com/od/gamespuzzles/a/aa082802a.htm`
- Foreign Broadcast Information Service. Abbreviations used in the press of Italy.
  `http://www.ut.ngb.army.mil/clp/linguists/fbis/ita.pdf`
- Andrea Sapuppo, Scuolissima. Abbreviazioni italiane.
  `http://www.scuolissima.com/2012/04/abbreviazioni-italiane.html`
- An abbreviation list created by an Italian native speaker.

C Normalization Examples

Tweets normalized by TweetNorm:
Normalization candidates in the orignal tweets and actual normalizations in the processed tweets are underlined.

- Example 1:

@marie455 Xke 6 triste :-(? tvtb :-* ,all. il n/ video con @LCuccello da nov 2014 https://youtu.be/x5PeQrRsqFo

Perch sei triste ? Ti voglio tanto bene , allegati il nostro video con Laura Cuccello da novembre 2014

- Example 2:

Quella di domaaani sar una luuuuuuunga giooornaaata!!!
Quella di domani sar una lunga giornata!

- Example 3:

Ventura, lei che maestro x i giovan8, un consiglio x Pobb?
Ventura, lei che maestro per i giovanotto, un consiglio per Pobb ?

- Example 4:

Twit della #Buonotte! :)
Twit della buonanotte!

- Example 5:

Mettersi a scaricare plugin per #photoshop alle 4 del mattino ed installarli alla prima. #mammamia!!
Mettersi a scaricare plugin per photoshop alle 4 del mattino ed installarli alla prima. Mamma mia!

- Example 6:

@heyitsflavia13 ti voglio taaaanti bene *___* sogzjlkdkdjaddfghk *attacco di dolcezza* http://t.co/SzUzdEPc
ti voglio tanto bene attacco di dolcezza

Stance-based Argument Mining – Modeling Implicit Argumentation Using Stance

Michael Wojatzki
Language Technology Lab
University of Duisburg-Essen
Duisburg, Germany
michael.wojatzki@uni-due.de

Torsten Zesch
Language Technology Lab
University of Duisburg-Essen
Duisburg, Germany
torsten.zesch@uni-due.de

Abstract

A major remaining challenge in argument mining is implicitness. We propose to model implicit argumentation using explicit stances and the overall stance of a debate. Our evaluation on a social media corpus shows that our model (i) can be reliably annotated even on noisy data and (ii) has the potential to improve the performance of automated argument mining.

1 Introduction

Argument mining aims at an automated analysis of persuasive communication. One yet unsolved problem is that –especially in informal settings – argumentation is often done implicitly. For instance, in a debate on atheism, one may observe an utterance such as *Bible: infidels are going to hell* or even shorter *#JesusOrHell*. In the context of a debate about atheism, both utterances implicitly express the argument that the author is against atheism, because the bible says that this will result in a stay in hell after death. However, both claims are never explicitly mentioned.

Typically, models of argument mining assume that an argument consists of at least an explicit *claim* and a number of optional supporting structures such as *premises* (Palau and Moens, 2009; Peldszus and Stede, 2013). Figure 1a shows an example of the simplest manifestation of these *claim-premise* schemes. However, in implicit argumentation the claim usually needs to be inferred, as it is not explicitly expressed (see figure 1b). We argue, that in the absence of explicit information, the claim always corresponds to the overall stance in the debate in which the utterance is made. *Stance* can be defined as being in favor or against a defined target such as a controversial topic, e.g. *being in favor of atheism* or *being against it* (Mohammad et al., 2016). Thus, one may always

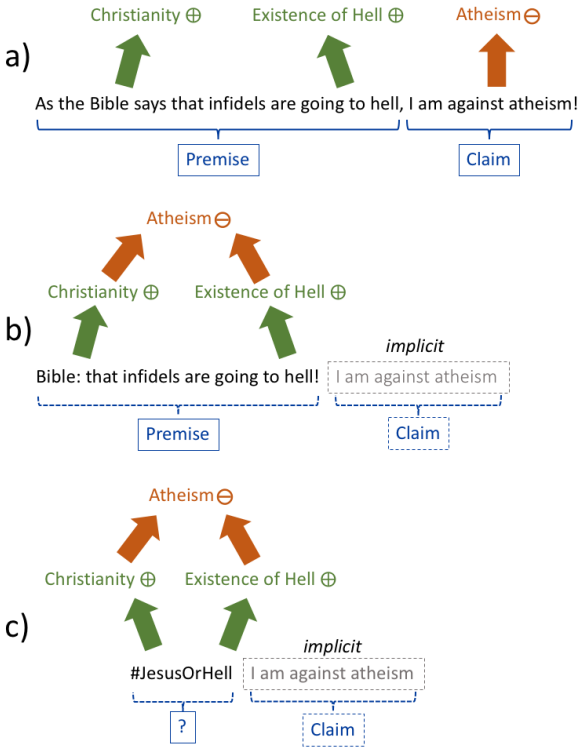

Figure 1: Stance-based vs. *claim-premise* model

transform a stance into a claim of the form *I am* $\{in\ favor|against\}$ [TARGET]. As further illustrated in figure 1c, the claim-premise scheme is also not well suited for fragments like *#JesusOrHell*, while the fragment clearly invokes stances on christianity and the existence of hell.

In this paper, we show how explicitly expressed stances and the (possibly implicit) debate stance can be used as a proxy for argumentation. Compared to the traditional models of argument mining, our model has the advantage that stances are more easily derived and frequent than rich rhetorical structures. As we enforce explicit stances to be backed by direct textual evidence, we ensure a high reliability of the model. We argue that our model is especially useful for argument mining on social media texts, as the informal mode of com-

munication leads to a high proportion of implicit arguments.[1] We annotate a corpus of noisy Twitter messages and show that our model can be reliably annotated and that it has the potential of improving the automated classification of stances as well as of traditional models of argument mining.

2 Related Work

Our model aims at capturing stances as a proxy for implicit arguments. Thus it cannot be directly compared with more complex models that assume typed relations between their components such as the *claim-premise* scheme. Here, we only discuss approaches linking stance and argumentation, and discuss the related work with respect to applicability to different text genres and inter-annotator agreement.

Boltužić and Šnajder (2014) use a set of predefined phrases such as *It is discriminatory to refuse gay couples the right to marry* and align them to stance labeled debate utterances. They report an agreement (Fleiss' Kappa) of 0.46 and 0.51 for the two debates in their corpus. Sobhani et al. (2015) also align predefined phrases with stance labeled comments but only indirectly relate them to the texts by mapping them to extracted statistical topic models. They state that this reduces the annotation effort, but the agreement remains rather low at 0.56 (Cohen's) kappa for tagging the arguments. Conrad et al. (2012) manually model two hierarchies of argumentative phrases with positive and negative stance as root nodes. Each hierarchy consists of more general phrases (such as *bill is politically motivated*) which are refined by phrases in the lower level of the hierarchy. After extensive training of the annotators, they reach a (Cohen's) kappa of 0.68. Hasan and Ng (2014) use argumentative phrases which have been previously extracted from the text. On four different domains they reach a (Cohen's) kappa of 0.78-0.82 on utterance level and of 0.61-0.67 on sentence level.

We thus conclude that enforcing an explicit grounding of annotation decisions in an utterance can be more reliably annotated than annotations that are mainly based on the interpretations of the annotators. Thus, in our model we only annotate stances if they have some explicit anchor in the text. For example, we would annotate a negative stance towards same-sex marriage (abbreviated notation: *Same-Sex Marriage* $\ominus$) only for a sentence like *gay marriage is a sin* where the stance is explicitly expressed, but not for a sentence like *as a true conservative, I trust in every word of the Bible* where *Same-Sex Marriage* $\ominus$ can only be inferred implicitly.

Misra et al. (2015) and Swanson et al. (2015) apply text summarization techniques to extract central propositions and then group them by a similarity measure which incorporates stance. For instance, if two statements relate to the same target, but express different polarities they are considered to be *roughly equivalent*. Consequently, stance is modelled only indirectly but may be inferred from the grouping of statements by the similarity measure. In addition, their approach relies on text summarization which does not make sense for very short texts such as the shown examples.

Another group of approaches deals with detecting agreement or disagreement between consecutive utterances (Ghosh et al., 2014; Clos et al., 2016), which could be interpreted as a stance towards the target that is mentioned in the first utterance. These models require a set of utterances organized in a conversation which limits the applicability. As we have seen from the examples in figure 1, even a single fragment can contain an argument. Our model should thus be applicable to single utterances, and not rely on a minimum text length.

It should be noted that all above mentioned studies have been carried out on data with relatively elaborate discussions, e.g. from dedicated web-based debating portals. We argue that applying those models to social media data like Tweets would result in considerably lower agreement, as the data contains a much higher proportion of implicit and less-elaborated arguments.

3 Modeling Arguments Using Stances

In order to solve the major challenge of implicit arguments that cannot be modeled well with existing approaches, we introduce a new model based on a *debate stance* that will in most cases be implicit, but can be inferred from one or more *explicit stances* that rely on textual evidence from the utterance. We thereby assume that an utterance is always made in the context of a certain debate.

Figure 2 gives on overview of the model which we metaphorically describe as an iceberg. In the

[1] For instance, the annotation of a comparatively elaborate social media corpus by Habernal et al. (2014) shows that almost half of the claims are implicit.

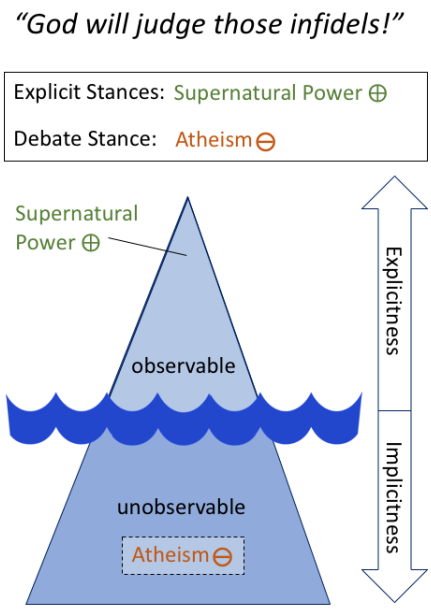

Figure 2: Our model and the iceberg metaphor for capturing implicit argumentation by using the the components (i) explicit stances and (ii) a debate stance.

context of a debate about atheism, an utterance like *God will judge those infidels!* is like the visible (explicit) part of the iceberg. It expresses a stance in favor of a supernatural power (*Supernatural Power* ⊕), while the actual stance on the debate target of atheism (*Atheism* ⊖) is not visible but must be inferred. Note that the debate stance might also be explicitly expressed (see figure 1a), but in implicit argumentation it has to be derived from the explicit stances.

In principle, each utterance evokes a large set of implicit stances (in a similar way as the iceberg contains a lot of invisible ice below the waterline). For instance, one may infer that a person uttering *Bible: infidels are going to hell!* is probably in favor of praying and might have a negative stance towards issues such as abortion, same-sex marriage, etc. However, we argue that being in favor of Christianity already implicitly covers these stances under a common sense interpretation. Depending on the present informational need these targets may be more or less relevant.

For modeling stance, we can build on plenty of research (Anand et al., 2011; Somasundaran and Wiebe, 2009; Sridhar et al., 2014; Hasan and Ng, 2013) and even a shared task on automatic stance detection (Mohammad et al., 2016). These works commonly define stance as being in favor of or against a given target. Consequently, stance is a tuple consisting of a target and a stance expression such as *Atheism* ⊕ or *Atheism* ⊖.

Debates can be categorized in two sided debates in which authors can take a pro or contra stance and more open debates which may contain several other targets (e.g. *What evidence do we have for global warming?*). However, we argue that each of the targets in an open debate – e.g. a certain piece of evidence for global warming – can be considered as a two sided debate. I.e. an authors may agree or disagree on an elevated sea level as evidence of global warming. Moreover, if one acknowledges that the participants in a two sided debate also discuss certain sub-topics, the separation between two sided debates and open debates vanishes.

Debate Stance As described above, we refer to the (frequently implicit) stance towards the target of the whole debate as *debate stance*. For instance, if in the context of an atheism debate someone describes their personal faith, we may assume that they want to communicate the fact that they are against atheism. Note that exactly the same utterance might not communicate a stance against atheism in the context of another debate such as on the importance of charity.

Explicit Stances While the overall debate stance may be implicit, there has to be some explicit information in the utterance that enables this inference. Otherwise the goal of the persuasive utterance (i.e. convincing someone or at least expressing her standpoint) cannot be achieved. As a stance can always be transformed into a claim which can be considered as the minimum constituent of an argument (Habernal et al., 2014; Palau and Moens, 2009), we argue that the minimal information that has to be provided in a persuasive utterance is a stance towards some target.

Given a stance, humans can infer the argument using a common sense interpretation. If one states *God will judge those infidels* (*Believe in God* ⊕) in an atheism debate, one can infer stances such as *being a infidel is a sin* ⊕, *God punishes infidels* ⊕ and the debate stance *Atheism* ⊖. If an author wants to deviate from this interpretation, they need to communicate this explicitly, e.g. by adding *but the constitution grants religious freedom* (*religious freedom* ⊕).

From lexical priming studies it is known that the perception of words can activate knowledge about

associated concepts or real-world events (Jones and Estes, 2012; Hare et al., 2009). Since there is also strong evidence for priming effects of stimuli other than words (Tulving and Schacter, 1990), we conclude that priming should be applicable to stances as well and therefore forms the underlying mechanism of implicit argumentation.

Thereby, our model of implicit argumentation aligns with the *Relevance Theory* proposed by Sperber and Wilson (1986) and the *Cooperative Principle* by Grice (1970) as we also assume that utterances provide hints on the intended meaning to the recipient. Particularly, our model shares the assumption of *Relevance Theory* that the precision of statements is such that a receiver can decode the meaning only by incorporating the context.

Selection of Stance Targets As indicated by our iceberg metaphor (see figure 2), just a small proportion of the argument is observable but the larger part is hidden from sight. The granularity of the stance targets has thereby to be considered with respect to the present informational needs. If one wants to get a more general view on the examples in figure 1, one could fall back to the target *belief in a supernatural power* which is also less explicitly covered. Depending on what degree of explicitness is chosen, an utterance can thereby express more than one explicit stance. Analogously, the unobservable parts of the argument vary in the degree of their implicitness. The degree of implicitness is seen here as the strength to which other stances are primed by the explicit part. For instance, if one claims the existence of hell, one affirms the existence of heaven with a small degree of implicitness but a stance about reincarnation is taken only very implicitly.

What level of granularity should be chosen is an open research question. As demonstrated by Conrad et al. (2012), a too fine grained distinction has the consequence of a sparse distribution which makes it difficult to derive relations between components of their model or to enable automated classification. Thus, selecting the most explicit targets does not appear to be the appropriate level to gain comprehensive insights on how taking a stance in a debate is manifested by explicit stances. However, if a target is too implicit, it might be invoked by authors in favor of the debate target as well as against the target.

4 Corpus Annotation

In order to show that our approach is indeed viable, we conduct an annotation study on social media data from the SemEval 2016 task 6 on stance detection. This enables us (i) to assess how reliably our model can be annotated, (ii) to examine what insights we can get by inspecting usage patterns of explicit stances, and (iii) to estimate how well our model can be assigned automatically. We now describe in more detail the utilized data, the annotation process, and how we derived the targets in a granularity that we found to be appropriate.

4.1 Data

As our argumentation scheme is centered around stance, we rely on data used by the first shared task on automated stance detection (Mohammad et al., 2016) which enables us to consider the present work in this context. A relevant property of the data, as stated by the task organizers, is that it contains a high proportion of tweets that do not explicitly mention the target and therefore can be considered as implicit utterances.

We focus here on Subtask A with tweets about five targets which are annotated for being in favor/against a target or if neither such inference is likely. We limit our study to 733 tweets on *Atheism* (513 from the training set and 220 from the test set), as we found the topic to require less knowledge about specific political events.

4.2 Derivation of Targets

In our model, choosing the right number and granularity of targets is crucial. On the one hand, they have to be expressive enough to capture differences in nuanced argumentation. On the other hand, they should not be too fine grained as this would result in very sparse distributions that cannot be handled by automated methods. Therefore, we utilize a semi-automated, bottom-up approach that focusses on concepts that are mostly explicitly expressed by named entities and nouns. We consider the 50 most frequent concepts. It should be noted that in this corpus of Twitter messages on Atheism, the *atheism* appears exactly once and the *atheist* only 6 times. This indicates that implicit argumentation is prevalent in social media.

As we want to ensure that the targets used enable us to differentiate the authors' positions sufficiently, we also consider the degree of association between nouns and named entities to the stances *Atheism* $\oplus$

and *Atheism* ⊖. In detail, we compute the collocation coefficient *Dice* (Smadja et al., 1996) for each word, and selected the 25 words which are most strongly associated with *Atheism* ⊖ and *Atheism* ⊕.

We found the resulting concepts to be too numerous and too fine-grained to be used in our model. We thus, manually group concepts into more coarse-grained targets. For instance, concepts such as *Bible* and *Jesus* are grouped into the target *Christianity*. A potential criticism of our approach is that at this stage of our work, we can not evaluate whether the set is best possible choice. We plan to shed light on this aspect in future research. The final set of derived, explicit targets is shown in table 1.

4.3 Annotation Process

Using the selected data, we let three annotators (two undergraduate and one graduate student of cognitive science) identify stances towards the derived targets and the debate target. In order to familiarize the annotators with our model, we previously trained them on a small data set that is comparable in its social media character but concerns a different target.

Since the data partly contains utterances which cannot be understood without further context, we give annotators the option to mark them accordingly. Irony is another phenomenon, which influences the interpretability. Therefore, we asked the annotators to annotate the tweets for irony as well.

Since it is still possible that our annotators interpret the tweets differently than in the original annotation, we re-annotated the debate stance using the original questionnaire described in Mohammad et al. (2016). While annotating explicit stances, the annotators had the instruction to only annotate stances towards targets if they have textual evidence for it.

5 Evaluation

In this section, we evaluate the annotated data. For this purpose, we first analyze the reliability of the annotation on different levels of granularity using Fleiss' Kappa (κ). For the analysis, we exclude tweets that are annotated for irony and understandability issues. However, we found that the annotators rarely agree on these phenomena as we get a κ of only 0.06 for understandability and a κ of 0.23 for irony. Therefore, we only exclude 18 tweets in which at least two annotators share the same judgment, which results in 715 tweets for the final corpus.

5.1 Inter Annotator Agreement

Since the explicit targets are annotated on the basis of textual evidence, we expect a high level of agreement. The notation of explicit targets should also result in a strong agreement of the annotation of the debate stance because it enforces a deep analysis of the communicative goal of an utterance. As shown in figure 3, we obtain a Fleiss' κ of 0.72 for the annotation of the debate stance. Unfortunately, we cannot compare our agreement to the originally SemEval data, as the organizers do not report a chance corrected agreement measure for their final decision. Also not directly comparable is the agreement of Sobhani et al. (2015) as they report weighted κ. We argue that their weighted κ of 0.62 is in a range similar to ours.

In figure 3, we also show the agreement for the explicit targets. Since explicit stances have a similar, deriving function like the argumentative phrases proposed by Conrad et al. (2012) and Hasan and Ng (2014), we compare our agreements to theirs which does not exceed a Cohen's κ of 0.68. Two targets (*Christianity* and *Islam*) yield especially high agreement above 0.8, because they are associated with clear signal words such as *Jesus* and *Quran* and other markers such as the numerical reference to biblical passages. Other targets such as *Secularism* and *Freethinking* are rather abstract. They hardly involve special signal words but still gain high agreements of a κ above 0.7, which shows that our annotators did not just learn to recognize certain keywords, but can also reliably annotate more abstract targets. This is further supported by the fact that the agreement for the annotation of *no explicit target* is also in this range. The targets *USA*, *Religious Freedom*, *Same-Sex Marriage*, and *Life After Death* yield only a moderate agreement between 0.4 and 0.6. An error analysis for the target *Same-Sex Marriage* shows that there is disagreement if the tweet contains a stance towards gay rights in general but not to gay marriage. We therefrom see two possibilities here to improve the agreement: On the one hand, we could choose more comprehensive targets such as *gay rights* to cover the combined positions. On the other hand, we could train the annotators to more consistently account for such differences. A rather low κ of 0.31 is obtained for the target *No Evidence*.

Explicit Target	Description	Examples for Textual Evidence
Christianity	belief in the religion Christianity	Jesus, Christ, Bible, Mary Mother of God, Catholic, Gospel
Freethinking	idea that truth should be formed on the basis of logic, reason, and empiricism	#freethinking, #DogmasNeverHelp
Islam	belief in the religion Islam	Quran, Ummah, Hadith, Mohammed, Allah
No Evidence	idea that there is no evidence for religion	there is no evidence for God
Life After Death	believe in an existence after death	paradise, heaven, hell, Dschanna
Supernatural Power	belief in a supernatural being or an abstract supernatural power	God, Lord, Jesus, holy spirit, Allah, Ganesha, destiny, predestination
USA	United States of America	our country, our state, America, US
Conservatism	the conservative movement in the USA	republicans, #tcot, tea party
Same-Sex Marriage	the right of a same-sex couples to marry	gay marriage, same-sex marriage
Religious Freedom	everyone should have the freedom to have and practice any religion	#religiousfreedom, right to choose your religion
Secularism	religion and nation should be strictly separated	separation of church and state, #secularism

Table 1: Explicit targets which are semi-automatically derived for the debate target *Atheism*

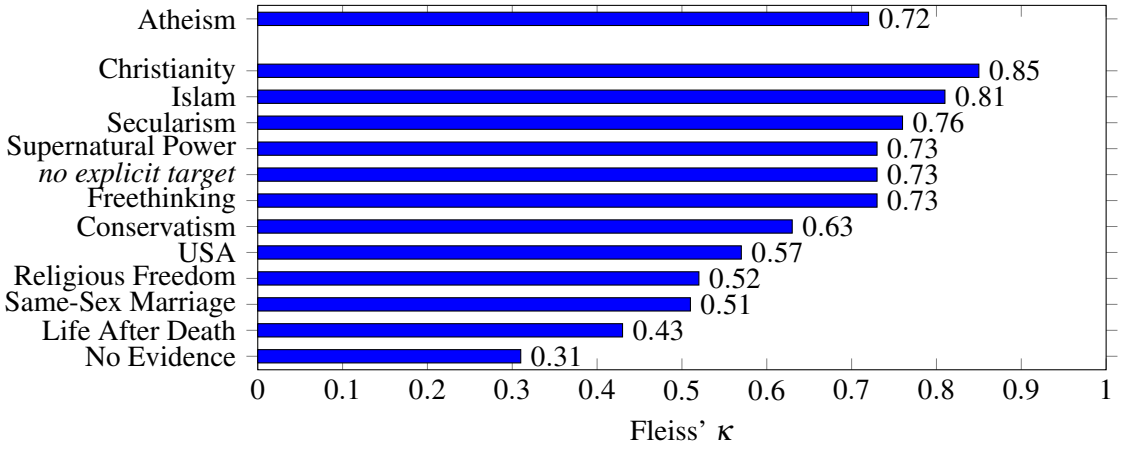

Figure 3: Inter-annotator agreement of the debate stance *Atheism* and explicit stances

Regarding this target, we observe that annotators sometimes deviated from our guidelines and incorporated different degrees of inferred knowledge as they used *Bill Nye* or *Richard Dawkins*[2] as anchors for their decisions, although the utterance contains no explicit stance in favor of *No Evidence*.

Finally, we obtain a κ of 0.63 for the joint decision on both the debate and the explicit targets. Note that this agreement is not directly comparable with the approaches from related work, as they only implicitly model the debate stance, do not report agreements of a joint decision or rely on stances that are determined by the structure of the data. The obtained inter-annotator agreement shows that our model can be annotated reliably and that the recognized difficulties may be compensated by a better training of the annotators and a better selection of targets.

[2] famous supporters of the position that there is no evidence for religion

5.2 Stance Pattern Analysis

In order to inspect usage patterns of explicit stance taking, we must agree on one annotation for each tweet. Since we do not assume that there are differences in the quality of the three annotators, we rely on a majority vote to compile a final annotation.

Figure 4 visualizes the frequency of the explicitly taken stances for Atheism $\oplus$ and Atheism $\ominus$. It shows that there are significant differences in the argumentation patterns between the two camps. As expected, if advocates of atheism are against a target such as *Christianity*, the opponents are mostly in favor of it or do not mention it. This pattern is also observable for the reverse case such as for *Freethinking*. Note that utterances addressing the target *Same-Sex Marriage* are exclusively annotated for expressing no stance towards *Atheism*. Further exceptions are the targets *USA* and *Religious Freedom* that are positively mentioned by both camps. However, a deeper analysis shows that

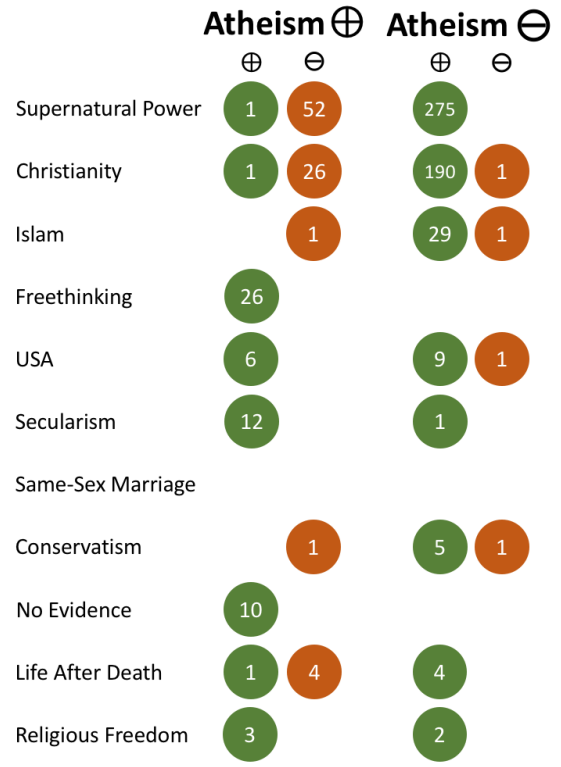

Figure 4: Frequency of explicit stances grouped according to debate stance

these targets always occur together with other targets which seem to be more relevant for the debate stance.

In order to analyze stance patterns in more details, we show which other stances are used together with the target *Supernatural Power* (the most frequent target in both camps) in figure 5. We observe that authors that are against Atheism use *Christianity* ⊕ together with *Supernatural Power* ⊕ in 50% of all cases. In contrast, authors that are in favor of Atheism only combine *Supernatural Power* ⊖ with *Christianity* ⊖ in 13% of all cases. The figure also shows that the other explicit stances only play a subordinate role in the combination with those targets.

From these analyses we can conclude that stable patterns of argumentation using explicit stances other than the debate stance exist. This is a strong indication for the validity of our assumption that the debate stance can be inferred from explicitly expressed stances.

5.3 Automatically Assigning Stances

We now want to examine how well the two main components of our model – the explicit stances and the debate stance – can be automatically assigned.

Target (# instances)	Majority Class Baseline	Our Approach
Supernatural Power (335)	.53	.78
Christianity (223)	.69	.79
Islam (43)	.94	.95

Table 2: Explicit stance classification (only showing targets occurring in at least 5% of all instances)

Feature Set	F_1
majority class baseline	.49
n-gram	.66
explicit stance$_{predicted}$	.65
explicit stance$_{oracle}$	.88

Table 3: Debate stance classification

We re-implement a state-of-the-art classifier (Mohammad et al., 2016) using the DKPro TC framework[3] (Daxenberger et al., 2014) and leave the development of sophisticated classification models to future research. For preprocessing, we rely on the DKPro Core framework[4] (Eckart de Castilho and Gurevych, 2014) and apply a twitter-specific tokenizer (Gimpel et al., 2011). In all experiments, we use ten-fold cross-validation and report micro averaged F_1.

Explicit Stances As the results from the stance detection task in SemEval-2016 (Mohammad et al., 2016) indicate, a support vector machine with a linear kernel equipped with simple word and character n-gram features is the state of the art in automated stance prediction. Table 2 shows the results of our reimplementation of state-of-the-art classifier (using weka's SMO) and the majority class baseline for comparison. The results indicate that the two most frequent targets can be classified with success, if one relates them to the majority class baseline. We observe that each target has its own linguistic markers such as the use of Arabic terms if one is in favor of Islam. Therefore, we argue that these peculiarities can be targeted even better by specialized features.

The analysis in table 2 excludes targets that have a insufficient coverage (less than 5% of all instances) to train a meaningful model. A possibility to deal with this sparsity may be to incorporate unlabelled data such as demonstrated for traditional models by Habernal and Gurevych (2015).

[3] version 0.8.0
[4] version 1.7.0

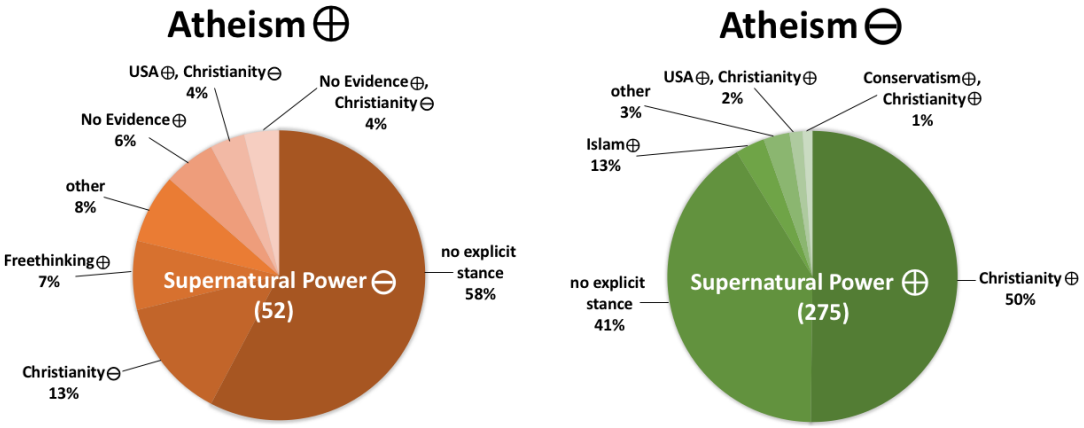

Figure 5: Most frequently used, explicit stances and the percentage shares to which they cooccur with other explicit stances

Debate Stance Table 3 shows the results obtained for automatically assigning the debate stance. Besides the majority class baseline ($F_1 = .49$), we use the same setup as for the explicit stances to train an n-gram based classifier and obtain an F_1 of .66. In order to evaluate the usefulness of explicit stances for inferring the debate stance, we use the predictions from the previous experiment as features for a decision tree classifier (J48). This stacked classifier performs on par (.65) with the n-gram based classifier. It seems that the quality of predicting explicit stances is not yet good enough to improve over the state-of-the-art without incorporating general n-gram features.

In order to estimate the potential of explicit stance features for classifying the debate stance, we add an oracle condition to our experiments in which we assume that the classification of explicit stances is done correctly. This classifier using only the manually annotated explicit stances yields an F_1 score of .88 showing that large improvements over the state of the art are possible if explicit stances can be more reliably classified. We believe that this is indeed possible as explicit stances are always grounded in the text itself, while the debate stance might only be indirectly inferred.

6 Conclusions and Future Work

We have identified implicitness as a major remaining problem in argument mining. Implicit arguments are only poorly supported by textual evidence and need to be inferred. We propose to model implicit argumentation by explicit stances and that cover more implicit stances and – most importantly – the overall stance that is taken in a debate. As we thereby enforce that the explicit stances are assigned with respect to textual evidence, we can ensure that our model is grounded on the actual utterances and less on their interpretation. As we argue that stances can always be interpreted as claims, our approach is interpretable in the form of a *claim-premise* scheme and therefore takes a step in bridging the gap between argument mining and stance detection. We provide evidence that this model can be reliably annotated, even on such a challenging domain as social media. In addition, we demonstrate that the model has the potential to boost performance in the automated detection of debate stance and traditional argument mining. We make the annotated data publicly available[5].

As this is a first attempt on modeling implicit arguments using stances, we see several lines of future research. First, we want to examine how the degree of granularity of explicit targets effects the quality of the model. Furthermore, we want to enhance our approach with an automated derivation of these targets. Finally, we want to improve the automatic assignment of explicit stances to unleash the full potential of explicit stances for argument mining.

Acknowledgments

This work was supported by the Deutsche Forschungsgemeinschaft (DFG) under grant No. GRK 2167, Research Training Group "User-Centred Social Media". We also want to thank our annotators Dominik Lawatsch and Niklas Meyer.

[5]http://www.ltl.uni-due.de/stance-based-am

References

Pranav Anand, Marilyn Walker, Rob Abbott, Jean E Fox Tree, Robeson Bowmani, and Michael Minor. 2011. Cats rule and dogs drool!: Classifying stance in online debate. In *Proceedings of the 2nd workshop on computational approaches to subjectivity and sentiment analysis*, pages 1–9, Stroudsburg, USA.

Filip Boltužić and Jan Šnajder. 2014. Back up your stance: Recognizing arguments in online discussions. In *Proceedings of the First Workshop on Argumentation Mining*, pages 49–58, Baltimore, USA.

Jérémie Clos, Nirmalie Wiratunga, Stewart Massie, and Guillaume Cabanac. 2016. Shallow techniques for argument mining. In *European Conference on Argumentation (to appear)*, Lisbon, Portugal.

Alexander Conrad, Janyce Wiebe, et al. 2012. Recognizing arguing subjectivity and argument tags. In *Proceedings of the Workshop on Extra-Propositional Aspects of Meaning in Computational Linguistics*, pages 80–88, Stroudsburg, USA.

Johannes Daxenberger, Oliver Ferschke, Iryna Gurevych, Torsten Zesch, et al. 2014. Dkpro tc: A java-based framework for supervised learning experiments on textual data. In *ACL (System Demonstrations)*, pages 61–66, Baltimore, USA.

Richard Eckart de Castilho and Iryna Gurevych. 2014. A broad-coverage collection of portable nlp components for building shareable analysis pipelines. In *Proceedings of the Workshop on Open Infrastructures and Analysis Frameworks for HLT*, pages 1–11, Dublin, Ireland.

Debanjan Ghosh, Smaranda Muresan, Nina Wacholder, Mark Aakhus, and Matthew Mitsui. 2014. Analyzing argumentative discourse units in online interactions. In *Proceedings of the First Workshop on Argumentation Mining*, pages 39–48, Baltimore, USA.

Kevin Gimpel, Nathan Schneider, Brendan O'Connor, Dipanjan Das, Daniel Mills, Jacob Eisenstein, Michael Heilman, Dani Yogatama, Jeffrey Flanigan, and Noah A Smith. 2011. Part-of-speech tagging for twitter: Annotation, features, and experiments. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers-Volume 2*, pages 42–47, Portland, USA.

Herbert P. Grice. 1970. *Logic and conversation*, volume 3. Academic Press.

Ivan Habernal and Iryna Gurevych. 2015. Exploiting debate portals for semi-supervised argumentation mining in user-generated web discourse. In *Proceedings of the EMNLP*, pages 2127–2137, Lisbon, Portugal. Association for Computational Linguistics.

Ivan Habernal, Judith Eckle-Kohler, and Iryna Gurevych. 2014. Argumentation mining on the web from information seeking perspective. In *ArgNLP*.

Mary Hare, Michael Jones, Caroline Thomson, Sarah Kelly, and Ken McRae. 2009. Activating event knowledge. *Cognition*, 111(2):151–167.

Kazi Saidul Hasan and Vincent Ng. 2013. Stance classification of ideological debates: Data, models, features, and constraints. In *Proceedings of the IJCNLP*, pages 1348–1356, Nagoya, Japan.

Kazi Saidul Hasan and Vincent Ng. 2014. Why are you taking this stance? identifying and classifying reasons in ideological debates. In *Proceedings of the EMNLP*, pages 751–762, Doha, Qatar.

Lara L. Jones and Zachary Estes. 2012. Lexical priming: Associative, semantic, and thematic influences on word recognition. *Visual word recognition*, 2:44–72.

Amita Misra, Pranav Anand, JEF Tree, and MA Walker. 2015. Using summarization to discover argument facets in online idealogical dialog. In *Proceedings of the NAACL HLT*, pages 430–440, Denver, USA.

Saif M. Mohammad, Svetlana Kiritchenko, Parinaz Sobhani, Xiaodan Zhu, and Colin Cherry. 2016. Semeval-2016 task 6: Detecting stance in tweets. In *Proceedings of the International Workshop on Semantic Evaluation (to appear)*, San Diego, USA.

Raquel Mochales Palau and Marie-Francine Moens. 2009. Argumentation mining: the detection, classification and structure of arguments in text. In *Proceedings of the 12th international conference on artificial intelligence and law*, pages 98–107, New York, USA.

Andreas Peldszus and Manfred Stede. 2013. Ranking the annotators: An agreement study on argumentation structure. In *Proceedings of the 7th linguistic annotation workshop and interoperability with discourse*, pages 196–204, Sofia, Bulgaria.

Frank Smadja, Kathleen R. McKeown, and Vasileios Hatzivassiloglou. 1996. Translating collocations for bilingual lexicons: A statistical approach. *Computational linguistics*, 22(1):1–38.

Parinaz Sobhani, Diana Inkpen, and Stan Matwin. 2015. From argumentation mining to stance classification. In *Proceedings of the NAACL HLT 2015*, pages 67–77, Denver, USA.

Swapna Somasundaran and Janyce Wiebe. 2009. Recognizing stances in online debates. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 1-Volume 1*, pages 226–234, Singapore.

Dan Sperber and Deirdre Wilson. 1986. Relevance: communication and cognition. *Language in Society*, 17(04):604–609.

Dhanya Sridhar, Lise Getoor, and Marilyn Walker. 2014. Collective stance classification of posts in online debate forums. In *Proceedings of the joint Workshop on Social Dynamics and Personal Attributes in Social Media*, pages 109–117, Baltimore, USA.

Reid Swanson, Brian Ecker, and Marilyn Walker. 2015. Argument mining: Extracting arguments from online dialogue. In *Proceedings of the 16th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 217–226, Prague, Czech Republic.

Endel Tulving and Daniel L. Schacter. 1990. Priming and human memory systems. *Science*, 247(4940):301–306.

Sentence Boundary Detection for Transcribed Tunisian Arabic

Inès Zribi
ANLP Research group,
MIRACL Lab.
University of Sfax, Tunisia
ineszribi@gmail.com

Inès Kammoun
ANLP Research group,
MIRACL Lab.
University of Sfax, Tunisia
kammoun91ines@yahoo.fr

Mariem Ellouze
ANLP Research group,
MIRACL Lab.
University of Sfax, Tunisia
mariem.ellouze@planet.tn

Lamia Hadrich Belguith
ANLP Research group,
MIRACL Lab.
University of Sfax, Tunisia
l.belguith@fsegs.rnu.tn

Philippe Blache
Aix-Marseille University
& CNRS LPL, 13100,
Aix-en-Provence, France
philippe.blache@lpl-aix.fr

Abstract

In this paper, we study the problem of detecting sentence boundary in transcribed spoken Tunisian Arabic. We compare and contrast three different methods for detecting sentence boundaries in transcribed speech. The first method uses a set of handmade contextual rules for identifying the limit of sentences. The second method aims to classify words into four classes according to their position in a sentence. Both methods are based only on lexical and some prosodic information such as silent and filled pauses. Finally, we develop two techniques to mix the results of the two proposed methods. We show that sentence boundary detection system can improve the accuracy of a POS tagger system developed for tagging transcribed Tunisian Arabic.

1 Introduction

Automatic or manually transcription, generally, produces a set of texts that represent the contents of a speech. Transcripts need some more structuring or segmentation to be used in different spoken language processing systems (*e.g.,* speech summarization, speech translation, syntactic parsing, etc.), for which sentence is the basic unit. However, it is difficult to find speech sentences because of the absence of punctuation marks in the transcripts, which occur at sentences boundaries in most written languages. Moreover, sentences in spontaneous speech are ill-formed, and sentence boundaries are indistinct (Akita et al., 2006). Therefore, Sentence Boundary Detection (SBD) of transcripts is the preliminary step for multiple Natural Language Processing (NLP) applications.

Dialectal Arabic (DA) poses multiple challenges to SBD task due to the absence of resources. In addition, boundaries of dialectal sentences are related to different lexical cues (connectors such as امّا AmA "but", coordination conjunctions و w "and", etc.), which do not always present borders of sentences (Belguith et al., 2005).

We address, in this paper, the problem of SBD of manually transcribed Tunisian Arabic (TA). We present methods that exploit lexical and some prosodic cues for detecting the boundaries of TA sentences.

This paper is structured as follows: We first review some previous related work (Section 2). In Section 3, we present an overview of TA. We, then, highlight the challenges of SBD for TA (Section 4). Section 5 is devoted to presenting our data. We, then, present our methods (Section 6). In Section 7, we give the evaluation results. Finally, we conclude with a discussion of future work.

2 Related Works

Numerous techniques are used to recognize sentences boundaries for different spoken languages where several are based on statistical approaches using machine-learning techniques.

Jamil et al. (2015) have presented a supervised *Adaboost* classifier for SBD of spontaneous spoken Malay language. Their system is based on seven prosodic features, rate-of-speech and volume.

Beeferman et al. (1998) have developed *CYBERPUNC* that inserts punctuation in the transcripts of an automatic speech recognition system. Their system is solely based on lexical information. It relies on a trigram language model and a straightforward application of the Viterbi algorithm.

Using decision tree and hidden Markov modeling techniques, Shriberg et al. (2000) have combined prosodic cues with word-based approaches, and have evaluated performance on two speech corpora. Obtained results show that the probabilistic combination of prosodic and lexical information give the best result over English's task speech segmentation into sentence and topic units.

Akita et al. (2006) have tested two different techniques: statistical language model (SLM) and support vector machines (SVM) for SBD of spontaneous Japanese. In the SLM-based technique, they have used linguistic likelihoods and occurrence of pause to find sentence boundaries. They have, also, integrated heuristic patterns of end-of-sentence expressions to suppress false alarms. The SBD performed by an SVM-based text chunker (Akita et al., 2006) is based only on lexical and pause information.

Few researchers have investigated SBD of modern standard Arabic (MSA) textual data. Nevertheless, it is still not addressed for DA. These systems are based on lexical information such as conjunctions, punctuation marks, and other lexical items.

Belguith et al. (2005) have used contextual rules for developing the system *STAr* that is able to segment Arabic text in paragraphs and sentences. The rules are mainly based on punctuation marks, conjunctions and other connectors. Belguith et al. (2005) have used some collection of newspaper articles and school books for extracting rules.

Chaibi et al. (2014) and Keskes et al. (2012) have exploited (Belguith et al., 2005)'s method for segmenting Arabic texts in clauses and minimal discursive units.

A statistical approach is tested by (Khalifa et al., 2011) to segment Arabic text into sentences. They have proposed semantic based segmentation method that classifies the connector و w[1] "and" into their rhetoric roles. Khalifa et al. (2011) have trained a SVM classifier using syntactic and semantic features. According to the meaning of the connector, the generated model can segment Arabic texts.

Keskes et al. (2013) have tested a Maximum Entropy (ME) for classifying word in three different classes. Each class represents the position of the word in the minimal discursive unit. They have proved that typographical, lexical and morphological features are enough for detecting minimal discursive unit.

The SBD of transcribed MSA is addressed by (Elshafei et al., 2007). They have developed a system based on hidden Markov models (HMM) that accepts an oral sentence and its orthographic transcription, and generates its phonemic transcription and the segmentation information of sentence. The system is trained using a corpus of Arabic TV news and is validated against manually segmented speech sentences (Elshafei et al., 2007).

3 Tunisian Arabic

Tunisian Arabic (TA) is a dialect of the North African (i.e., the Maghreb) dialects spoken in Tunisia (Zribi et al., 2014). It is considered a low variety given that it is neither codified nor standardized even though it is the mother tongue and the variety spoken by all the population in daily usage (Saidi, 2007). Approximately eleven million people speak at least one of the many regional varieties of TA (Zribi et al., 2014).

There are many differences as well as similarity points between TA and MSA in different levels. In order to compare these two varieties of Arabic language, we focus on four levels (i.e. the phonological level, the morphological level, the lexical level and the syntactic level).

3.1 The phonological level

The vocalic system of TA is reduced (Tilmatine, 1999). Some short vowels are neglected, especially if they are located in the last position of the word (Mejri et al., 2009). The MSA verb شرب /šariba/ "he drank" is pronounced /šrib/ in TA. We note the deletion of the vowels at the first and the last position of the verb. TA has, also, a long vowel /e:/ which does not exist in MSA (Zribi et al., 2014). Moreover, the consonant system includes some phonetic differences (Mejri et al., 2009). In some cases, the Arabic consonant ق /q/ is pronounced /g/. The MSA word بقرة bqrħ /baqara/ "cow" is pronounced in TA /bagra/. In addition, some consonants in TA have multiple pronunciations. For

[1] Transliteration is coded with Buckwalter transliteration. For more details about it, see (Habash et al., 2007).

example, the MSA consonants غ γ /γ/ and ج j /j/ can be pronounced in TA respectively as /x/ or /γ/ and /j/ or /z/.

3.2 The morphological level

The main difference between MSA and TA is on the affix level. We notice the presence of new dialectal affixes and the deletion of others. Dual suffixes ان An and ين yn are generally absent in TA. They are replaced by the numeral زوز[2] zwz "two" located after or before the plural form of the noun. However, some words in TA can be agglutinated to the suffix ين yn to express duality. In verb conjugation, TA is characterized by the absence of the dual (feminine and masculine) and the feminine in the plural. It has seen many simplifications in its affixation system (Ouerhani, 2009). Indeed, new affixes have appeared. The first one is the negation clitic. It is agglutinated to the last position of the verb that must be preceded by the negation particle مَا mA (*e.g.,* مَاكليتش mA klytš "I don't eat") (Mejri et al., 2009). The interrogation prefix of MSA أَ Â is transformed in TA into the suffix شي šy (*e.g.,* خرجشي, xrjšy, "Did he go out?"). Likewise, the future prefix سـ s- is replaced by the particle (بَاش bAš "will"). In addition, we note the absence of the dual clitics in TA.

3.3 The lexical level

TA is distinguished by the presence of words from several other languages. The presence of these languages mainly occurred due to historical facts. We find in Tunisia a significant amount expressions and words from European languages such as Spanish, French, Italian, Turkish and even Maltese (*e.g.,* قطوس qTws "cat" is of Maltese origin; كوجينة kwjynħ "kitchen" is of Italian origin; بلَاصة blASħ "place" and بَاكو bAkw "package" are derived from French language). In addition, TA has several words from the vocabulary of the Berber language (*e.g.,* برنوس, brnws, "traditional clothes") (Zribi et al., 2014).

In addition to all these borrowed terms, which have been integrated in the TA morpho-phonology, Tunisians code switch often in daily conversations, particularly from French (*e.g.,* ça va ? "Okay?", désolé "sorry", rendez-vous "meeting", etc.). All these expressions and words are used without being adapted to TA phonology.

3.4 The syntactic level

The syntactic differences between MSA and TA are minors. The MSA word order is generally VSO (Verb subject Object) especially in verbal sentences. But in TA, the preferred word order is SVO (Mahfoudhi, 2002). The VSO and VOS orders are also used in TA.

4 Challenges in TA Sentence Boundary Detection

Arabic language characteristics.

SBD is a challenging task for Arabic language that is characterized by the absence of capital letters and the boundaries of sentences are not generally marked with punctuation marks. We often find a paragraph in Arabic language, which has only one full stop. Boundaries of Arabic sentences are strongly related to conjunctions and other lexical expressions. These lexical cues are not necessarily present sentence limits. They have other discursive functions. For example, the interjection (بَاهي, bAhy, "OK") can be used as an adjective that means "good".

Spoken language characteristics.

The spoken form of the Arabic language presents other challenges for the task of SBD. Firstly, the transcripts are usually not punctuated. Similarly, linguists interested in speech quickly deserted the notion of sentence (Tellier et al., 2010). We have to define, first, the term *sentence*. In TA oral, we can detect several types of sentences: well-formed sentences, incomplete sentences, and sentences containing disfluent segments. The incomplete sentences are very frequent in oral. The disfluency, also, affects the structure of the sentences by involving several elements of different nature in a sentence. Truncated words, filled pauses, silent breaks, repetitions, etc. affect the syntactic structure of the sentence. So, it is necessary to define the units of statement that we suggest detecting its boundary.

Tunisian Arabic characteristics.

TA is a spoken variety of Arabic that Tunisians code switch between MSA and French language.

[2] We follow the CODA-TUN convention (Zribi et al., 2014) when writing examples of words in TA.

The massive use of words from foreign languages and code switching engender in certain cases a loss of the syntactic structure of sentences. Indeed, TA is characterized by an irregularity in the word order in the sentence. We can express a single sentence with several syntactical structures: Subject-Verb-Object (SVO), Verb-Subject-Object (VSO) and Object-Verb-Subject (OVS) (Mahfoudhi, 2002). The mix of language (MSA, TA, and French) and the free word order for TA increase the difficulty of SBD.

Consider the English sentences: "*It is true that we are today... It is a day of celebration, but we have to work...*". These sentences can be translated into the following sentence:
(c'est vrai امَا c'est le jour de la fête اليوم احنَا اللّي نخدموَا يلزمنَا, c'est vrai Allly AHnA Alywm c'est le jour de la fête AmA ylzmnA nxdmwA).

The translated sentence is composed of the French phrase (*c'est vrai*, "it is true"), the French sentence (*c'est le jour de la fête*, "it is a day of celebration") and a set of TA words. SBD of such a sentence, which is very frequent in daily speech of Tunisians, is very difficult. Indeed, in French grammar, the expression (*c'est*, "it is") always marks the beginning of a new sentence. However, this expression can be used anywhere in TA sentence. The first occurrence of the expression (*c'est*) introduces the start of a new sentence, but it is not the case for the second occurrence.

To conclude, the presence of many foreign words in the TA speech and the code switching phenomena improve the difficulties of SBD of TA.

5 Data

5.1 Presentation

In this work, we used a manually transcribed TA corpus, created by (Zribi et al., 2015), and labeled as "STAC". The corpus consists of about 42,388 words, and follows the CODA-TUN (Zribi et al., 2014) convention for writing TA words and OTTA guideline (Zribi et al., 2013) for annotating the phenomena of the oral. The corpus is morphosyntatic annotated and segmented into sentences. Speech text for each speaker is divided into many speech turns. Zribi et al. (2015) gathered the speech turn for each speaker in a unique text. They, then, segmented it in utterances. They, considered a sentence a semantically meaningful unit.

5.2 Preparation

In STAC Corpus, the experts have performed the segmentation manually. We have redone the segmentation of the corpus with two experts to validate the segmentation of sentences. We have calculated the inter-annotator agreement. The two experts achieved a Kappa coefficient rate of 0.86% indicating almost perfect agreement.

All types of annotations are removed from the corpus. We kept only annotations that mark incomplete words, filled pauses and named entities. We eliminated, also, all specific symbols from the corpus.

The STAC corpus is divided into three parts. The first part of the STAC corpus was used for training our methods. It is composed of 32,012 words and 6,133 sentences. The second part comprised 7,201 words and 1,215 sentences to test the different proposed approaches. The remaining part of the STAC corpus (440 sentences and 3,175 words) is used for development.

6 Our Methods

In this section, we describe three methods for SBD of TA. Our proposed methods belong to three approaches: rule-based, statistical and hybrid.

6.1 Rule-based method

Rule-based techniques are proposed for developing MSA SBD systems. The handmade rules are essentially based on punctuation marks, conjunctions and other connectors. We propose to apply this technique for segmenting TA transcripts. We have used *lexical items* (such as conjunctions and other markers) and *two simple prosodic features* (silent and filled pauses) for designing our SBD rules.

The *lexical markers* are in certain cases specific to oral. In others, they can be used in the written form of the dialect. We have classified our rules following this criterion. The role of our segmentation rules is to detect a word (or an expression) at the beginning of a sentence. The rules are, also, based on words belonging to the right and/or the left context. We call them contextual rules (CR).

Contextual rules follow the same structure as defined by (Belguith et al., 2005). They have the following form:

Left Context	Marker	Right Context
G	X	D

G, *X* and *D* present lexical items which can be the beginning of a sentence. X is a trigger marker. If the left context *G* and/or the right context *D* are present, then *X* or *D* can be the beginning of a sentence. The window size of right and left context is variable according to the number of words that compose the lexical markers.

We have extracted two sets of rules. The *first set* groups rules that detect sentence boundaries of the oral form of TA. These rules are based on oral *specific lexical items* and *prosodic features*. Indeed, silent pauses are located in 57.25% of the cases at the first position of sentences. In this case, the silent pause can be compared to a full stop in writing texts. However, in 42.75% of cases, silent pauses are in the right or the left context of the first position of the sentence. Filled pauses are also located in the last or the first position of sentences. Based on these two prosodic features, we have extracted six contextual rules.

Below (See Table 1) is an example of contextual rule based on a silent pause and some lexical features. If the trigger marker is equal to a silent break "#" and the left context belongs to this list of words, then, the break is a mark of the beginning of a sentence.

Left Context	Marker	Right Context
Interrogative Adverb : عَلاش ʕlAš "why", قدَاش qdAš "how much", etc. **Expression that marks time:** عَام كل kl ʕAm "every year", غدوة γdwħ "tomorrow", etc.	#	∅

Table 1: Contextual rule based on the silent pause.

The *second set* of rules is more generic. It can be applied to the written form of TA. Rules conception is based on connectors, personal and relative pronouns, verbs, etc. Indeed, the syntactic structure of TA is very complex. Thus, we had difficulty in identifying patterns to detect the boundaries of sentences since the STAC corpus is an oral corpus with a high degree of spontaneity (95.65%). Sentences with simple structure present only 15.86% of our corpus. Below (See Table 2) is an example of rule that detects boundaries of sentences based on verbs.

Left Context	Marker	Right Context
∅	**Verb**	الَاقل علَى ʕlỳ AlAql "at least", أَيَا ÂyA "come on", لَا lA "no", وَاِلَا wAilA "otherwise", معنَاتهَا mʕnAthA "that is to say", لهنَا lhnA "here", etc.

Table 2: Contextual rule based on a verb.

This rule allows the detection of sentence that begins with a verb proceeded by an expression belonging to this list.

At the end, we have extracted in total 23 contextual rules. During the design of our rules, we have kept only rules that their precision is superior to 50%.

6.2 Statistical method

We have experimented with another approach for the SBD of TA. The task of SBD is converted into a word classification. We have proposed to classify words into four classes:

- "B-S" for marking the first word of the sentence,
- "I-S" for marking the word in the sentence,
- "E-S" for marking the last word of the sentence,
- and finally "S" for marking sentence composed of a single word.

We have built a classifier based on the rule-based classifier PART (Mohamed et al., 2012). Part is a partial decision tree algorithm, which is the development version of C4.5 and RIPPER algorithms (Mohamed et al., 2012). The main specialty of the PART algorithm is that does not need to do global optimization like C4.5 and RIPPER to generate exact rules, but it practiced separately and-conquer

strategy. For example, it builds a rule, and removes instances. It covers, and continues to create a recursive rules for the remaining of instances until there are no instances. PART builds a partial C4.5 decision tree in every iterative and makes the "best" leaf into a rule (Mohamed et al., 2012).

We have experimented with other classification methods included in the WEKA machine-learning tool[3]. However, PART gives the best results for our task.

The result of a classifier is strongly influenced by the set of defined features. In literature, the SBD task for spoken language is mainly related to two types of features: linguistic and prosodic features. The prosodic information (such as intonation, rhythm, etc.) is absent in our work. Thus, we have used two simple prosodic features that are silent and filled pauses. In the design of our features, we rely on linguistic features like adverbs, adjectives, verbs, etc. We note that we use lexicon lookup for determining words part-of-speech .

We have also used contextual features. To fix the window size, *n*, we have tested several contexts. We have experimented with *n=0*, *n=1*, and *n=2*. We show that *n=2* is the best configuration for our task.

Finally, we have used dynamic feature. It uses the class that is dynamically assigned to the two preceding words. Features given to PART are presented in Table 3. We note that the features take two possible values: *true* or *false*. They specify whether a word in the context belonging to the possible values set.

Features	Examples of the possible value
Silent Pause	#
Filled Pause	آ Ā "euh"
Expression marking the beginning of sentence	لكن lkn "but"
Conditional particle	ولأَن wlÂn "and because"
Discursive marker	معنَاتهَا mʕnAthA "that is to say"
Expression marking place	ثمة θmħ "there is"
The verb "want"	حبيت Hbyt "I want"
The verb "say"	يقول yqwl "he says"
Verb	TA verbs
Personal pronoun	آنَا ĀnA "I"
Verb "to be"	كَان kAn "he was"
Relative pronoun	اللّي Ally "that"
Demonstrative pronoun	هذية hðyħ "this"
Expression marking the time	كل عَام kl ʕAm "every year"
Interrogative adverb	علَاه ʕlAh "why"
Special expression	بصرَاحة bSrAHħ "honestly"
Greeting expression	عَالسلَامة ʕAlslAmħ "hello"

Table 3: Features for PART classifier.

6.3 Hybrid method

We have proposed to combine the result of the rule-based method and the statistical method. We have tested three different methods for combining the results of the two previous methods.

The *first method* consists of analyzing the transcripts using the contextual rules. The output of this step is a set of sentences. We have reanalyzed the longer sentences with the statistical model. We consider that a sentence is long only if the number of words is higher than 9. Nine words was chosen because it is the average number of words per TA sentence and nine gives us the best development results.

The *second method* is the opposite of the first method. It consists of applying, in the first step, the model generated by the statistical method. Then, we apply for the longer sentences contextual rules for segmenting them.

The *third method* consists of using the generated rules from the PART algorithm. We have suggested using simultaneously contextual rules and the generated rules by the algorithm PART for segmenting TA transcripts.

PART algorithm extracts a set of rules from the training corpus that classify words to four classes (B-S, I-S, E-S and S). These rules have the following form: "*if condition(s), then conclusion*".

We have chosen rules that classify words into "B-S" and "S". These rules can detect words at the first position of the sentence and word that presents a whole sentence. Only no redundant rules are

[3]http://www.cs.waikato.ac.nz/ml/weka/

selected. We have attributed to each rule a score. We have calculated it by applying the rule to a validation corpus composed of 440 sentences. This corpus is not used for generating rules. We have calculated the success rate for each rule. If its rate exceeds 75% we kept the rule. We remark that 40 rules attribute incorrectly class. All remaining rules are equal to handcrafted contextual rules.

Therefore, this method fails to integrate automatic generated and handcrafted rules. However, it shows that the automatic rule extraction can generate rules equal to handmade rules.

7 Evaluation

We look first at the performance of the three SBD methods proposed in this paper. We compare these methods against the baseline. Then, we test the effect of SBD methods on POS tagging of transcribed spoken TA.

7.1 Results and discussion

The evaluation metrics we use are recall, precision and F-measure. We have evaluated how well we could correctly segment TA transcripts. In this evaluation, we have compared our proposed methods: rule-based method (CR), statistical method (PART) and two hybrid methods (Hyb1 and Hyb2) against the baseline.

We have used STAr system (Belguith et al., 2005) as our baseline. STAr is SBD system designed for written form of MSA and it is based on a set of contextual rules. We have chosen STAr since some of its contextual rules are shared with TA. These rules are based primarily on the coordination conjunction (و, w, "and").

Table 4 lists the results of the different methods. We see that running statistical method alone gives us the best SBD results. We reported improvements up to 27.35% compared to the baseline. We see that the STAr system performs poorly on TA input. However, the precision value of the baseline is good (82.45%). This is due to the high number of TA sentences that begin with the coordination conjunction (و, w, "and"). The results given by rule-based method are lower than those of statistical method. Indeed, some of lexical markers are located far from sentences limits. This is due to the relative free order of some TA sentences. As well, some markers have other discursive functions that falsified the output of the application some contextual rules.

We turn now to analyze the hybrid methods results. The application of the first hybrid method has improved the recall value of the contextual rules. We notice an improvement of 4.11%. By against, it decreases the precision value (5.65%) compared to the method PART. We see that the application of PART algorithm followed by contextual rules downgraded the recall value. The value fell down from 72.42 to 66.00 (a decrease of 6.42%).

The second step of the two hybrid methods divides the long sentences into very small segments. This segmentation increases the number of sentences, but it decreases the accuracy of the SBD.

In conclusion, we note that the rule-based method and statistical method are powerful for the task of SBD. However, the higher increase (gain) has been observed in statistical method.

	Recall	**Precision**	**F-measure**
Baseline	40.98	82.45	54.75
CR	68.31	90.841	77.98
PART	**72.5**	**94.8**	**82.1**
Hyb1	72.42	89.15	79.92
Hyb2	66.00	73.91	69.73

Table 4: Comparison of the performance of the different SBD methods.

7.2 Extrinsic Evaluation: POS tagging of Tunisian Arabic

Part-of-speech tagging task (POS tagging or POST), is the process of marking up a word in a text (corpus) as corresponding to a particular part of speech, based on both its definition and its context — *i.e.*, its relationship with adjacent and related words in a phrase, sentence, or paragraph[4]. Thus, the detection of sentence (in writing language) and utterance boundaries (in spoken language) is considered one of the necessaries preliminary steps. Indeed, SBD for written languages is trivial due to the presence of punctuation marks and capital letters notably on Indo-European languages. Contrariwise, it is not trivial for spoken languages, specifically for spoken Arabic dialect.

We present in this section the effect of sentence boundary detection on POS tagging of transcribed spoken TA. Here, we are evaluating a POS tagger for TA trained on the STAC corpus (Zribi et al., 2015). The proposed tagger is tested with three different training methods: the statistical method

[4]https://en.wikipedia.org/wiki/Part-of-speech_tagging

SVM (Vapnik, 1995) and two rule-based classifiers (Ripper (Cohen, 1995) and PART (Collins and Singer, 1999)). We compare the performance of this tagger when it is trained on a manually (HandSeg), automatically (AutSeg) and non-segmented (NoSeg) version of the STAC corpus. In order to make the best use of our corpus, we tested our POS tagger using a 10-fold cross-validation procedure. Table 5 shows the result of the evaluation.

We remark that the SBD system helps the TA POS tagger to improve its accuracy. We note that SVM and RIPPER performed better when the SBD system detects short sentences. The value of accuracy of our POS tagger trained on SVM has decreased from 61.78% (non-segmented corpus) to 63.66% (corpus segmented with the second method of hybridization). Likewise, the accuracy increases from 62.53% to 64.84% when Ripper is used for training the tagger. However, the PART algorithm works best with long sentences. We show that the best value is given by using non-segmented corpus.

		Ripper	**PART**	**SVM**
NoSeg		62.53	**71.88**	61.87
HandSeg		63.92	70.55	63.02
AutSeg	**PART**	61.69	66.58	61.04
	CR	**64.84**	70.65	63.04
	Hyb1	64.20	70.21	63.39
	Hyb2	63.92	68.22	**63.66**

Table 5: The accuracy values of the POS tagger trained and tested with a manually (HandSeg), automatically (AutSeg) and non-segmented (NoSeg) corpus.

8 Conclusion

In this paper, we have proposed three different methods for detecting Tunisian Arabic sentence boundaries. We have experimented a rule-based, statistical, and hybrid method. These different methods are based on linguistic and two simple prosodic cues. The proposed method has shown encouraging results.

As future work, we intend to add more prosodic features to improve the efficiency of our system. We also intend to realize an extrinsic evaluation of our system in some NLP applications dealing with the spoken form of Tunisian Arabic. Finally, we aim to expand the training and the test corpora to cover other types of TA sentences.

References

Yuya Akita, Masahiro Saikou, Hiroaki Nanjo, and Tatsuya Kawahara. 2006. Sentence boundary detection of spontaneous japanese using statistical language model and support vector machines. In *INTERSPEECH 2006 - ICSLP, Ninth International Conference on Spoken Language Processing, Pittsburgh, PA, USA, September 17-21, 2006*.

Doug Beeferman, Adam L. Berger, and John D. Lafferty. 1998. Cyberpunc: a lightweight punctuation annotation system for speech. In *Proceedings of the 1998 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP '98, Seattle, Washington, USA, May 12-15, 1998*, pages 689–692.

Lamia Hadrich Belguith, Leila Baccour, and Ghassan Mourad. 2005. Segmentation de textes arabes basée sur l'analyse contextuelle des signes de ponctuations et de certaines particules. In *TALN 2005*.

Anja Habacha Chaibi, Marwa Naili, and Samia Sammoud. 2014. Topic segmentation for textual document written in arabic language. *Procedia Computer Science*, 35:437 – 446. Knowledge-Based and Intelligent Information ; Engineering Systems 18th Annual Conference, KES-2014 Gdynia, Poland, September 2014 Proceedings.

William W. Cohen. 1995. Fast effective rule induction. In Morgan Kaufmann, editor, *In Proceedings of the Twelfth International Conference on Machine Learning*, pages 115–123.

M. Collins and Y. Singer. 1999. A simple, fast and effective rule learner. *Proceedings of the 16th National Conference on Artificial Intelligence (AAAI-99)*, pages 335–342.

Moustafa Elshafei, Mohammad Ali, Husni Al-Muhtaseb, and Mansour Al-Ghamdi. 2007. Automatic segmentation of Arabic speech. In *Workshop on Information Technology and Islamic Sciences.*

Nizar Habash, Abdelhadi Soudi, and Timothy Buckwalter. 2007. *Arabic Computational Morphology: Knowledge-based and Empirical Methods*. Springer Netherlands.

Nursuriati Jamil, M.I. Ramli, and N. Seman. 2015. Sentence boundary detection without speech recognition: A case of an underresourced language. *Journal of Electrical Systems*.

Iskandar Keskes, Farah Benamara, and Lamia Hadrich Belguith. 2012. Clause-based discourse segmentation of arabic texts. In *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey, may. European Language Resources Association (ELRA).

Iskander Keskes, Farah Benamara Zitoune, and Lamia Hadrich Belguith. 2013. Segmentation de textes arabes en unités discursives minimales. In *Conférence du Traitement Automatique des Langues Naturelles - TALN 2013*, pages pp. 435–449, Sables d'Olonne, FR. LINA - Laboratoire d'Informatique de Nantes Atlantique.

Iraky Khalifa, Zakareya Al Feki, and Abdelfatah Farawila. 2011. Arabic Discourse Segmentation Based on Rhetorical Methods. *International*

Journal of Electric and Computer Sciences IJECS-IJENS, 11(01):10–15, February.

Abdessatar Mahfoudhi. 2002. Agreement lost, agreement regained: A minimalist account of word order and agreement variation in arabic. *California Linguistic Notes*, XXVII(2).

Salah Mejri, Mosbah Said, and Inès Sfar. 2009. Pluringuisme et diglossie en tunisie. *Synergies Tunisie*, 1:53–74.

W. N. H. W. Mohamed, M. N. M. Salleh, and A. H. Omar. 2012. A comparative study of reduced error pruning method in decision tree algorithms. In *IEEE International Conference on Control System, Computing and Engineering (ICCSCE)*, pages 392–397, Nov.

Béchir Ouerhani. 2009. Interference entre le dialectal et le littéral en tunisie: Le cas de la morphologie verbale. In *Synergies Tunisie n1*, pages 75–84.

Darine Saidi. 2007. Typology of Motion Event in Tunisian Arabic. In *LingO*, pages 196–203.

Elizabeth Shriberg, Andreas Stolcke, Dilek Z. Hakkani-Tür, and Gökhan Tür. 2000. Prosody-based automatic segmentation of speech into sentences and topics. *Speech Communication*, 32(1-2):127–154.

Isabelle Tellier, Iris Eshkol, Samer Taalab, and Jean-Philippe Prost. 2010. Pos-tagging for oral text with crf and category decomposition. *Research in Computing Science*, 46:79–90.

Mohamed Tilmatine. 1999. Substrat Et Convergences: Le Berbère Et L'arabe Nord-Africain. *Estudios de dialectología norteafricana y andalusí*, pages 1–3.

Vladimir N. Vapnik. 1995. *The Nature of Statistical Learning Theory*. Springer-Verlag New York, Inc., New York, NY, USA.

Inès Zribi, Marwa Graja, Mariem Ellouze Khmekhem, Maher Jaoua, and Lamia Belguith Hadrich. 2013. Orthographic Transcription for Spoken Tunisian Arabic. In *14th International Conference CICLing 2013, Proceedings, Part I, Samos, Greece, March 24-30*, volume 7816 of *LNCS*, pages 153–163. Springer.

Inès Zribi, Rahma Boujelbane, Abir Masmoudi, Mariem Ellouze, Lamia Belguith Hadrich, and Nizar Habash. 2014. A Conventional Orthography for Tunisian Arabic. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'2014), Reykjavik, Iceland, May 26-31*, pages 2355–2361. ELRA.

Inès Zribi, Mariem Ellouze, Lamia Hadrich Belguith, and Philippe Blache. 2015. Spoken Tunisian Arabic Corpus " STAC": Transcription and Annotation. *Research in computing science*, 90.