An Out-of-Domain Test Suite for Dependency Parsing of German

Wolfgang Seeker, Jonas Kuhn
Institut für Maschinelle Sprachverarbeitung
University of Stuttgart
{seeker,jonas}@ims.uni-stuttgart.de

Abstract

We present a dependency conversion of five German test sets from five different genres. The dependency representation is made as similar as possible to the dependency representation of TiGer, one of the two big syntactic treebanks of German. The purpose of these test sets is to enable researchers to test dependency parsing models on several different data sets from different text genres. We discuss some easy to compute statistics to demonstrate the variation and differences in the test sets and provide some baseline experiments where we test the effect of additional lexical knowledge on the out-of-domain performance of two state-of-the-art dependency parsers. Finally, we demonstrate with three small experiments that text normalization may be an important step in the standard processing pipeline when applied in an out-of-domain setting.

Keywords: dependency conversion; out-of-domain evaluation; dependency parsing

1. Introduction

Most of today's publicly available tools for doing natural language processing use statistical models that have been trained on manually annotated newspaper text. However, when people use these tools, they apply them to any kind of language data that they find interesting.

Statistical models trained in one domain but applied to another will normally yield considerably worse results compared to an in-domain setting (see e. g., Daume III (2007), Gildea (2001) for parsing). Since a statistical model learns a statistical distribution, its performance naturally decreases when it is applied to data from a different distribution because of the mismatch between training and test distribution. When statistical NLP tools trained on newspaper text are applied to weblogs, cooking recipes, manuals, microblogs, or whatever else people would like to run their tools on, this mismatch in distribution is the standard case rather than the exception.

We could of course simply create large manually annotated corpora for each of the domains, so that we can train different statistical models for different domains. That way, there would be an appropriately trained model for each domain. However, manual annotation is time-consuming and expensive and might not even be worth it, because it is not clear how to define precisely what a particular domain is anyway. A better option might therefore be to develop techniques for adapting statistical models to foreign domains. Research in domain adaptation specifically targets these issues and strives to develop models that generalize better to new domains, thus handling domain shifts more robustly. In order to evaluate and subsequently improve domain-adaptation techniques, however, we need manually annotated test data for many different domains that provide us with a better and more comprehensive understanding of the performance of our tools and models. While this still involves manual evaluation, we would need considerably less data for testing our models than for training them.

In this paper, we present a dependency conversion of two publicly available German data sets, namely the SMULTRON corpus (Volk et al., 2010) and a small part of the EuroParl corpus (Pado and Lapata, 2005). Both data sets together provide five different out-of-domain test sets for dependency parsing of German. We use the same conversion style that we used in Seeker and Kuhn (2012) to convert the German TiGer treebank (Brants et al., 2002) to dependency format. Having the TiGer corpus as training data with the same dependency annotation minimizes effects due to systematic differences in the annotation schemes, allowing the researcher to focus solely on the effects that stem from systematic differences in the distribution of language phenomena. The conversion tool is open-source and freely available.[1]

With the exception of English, there are few languages for which we have several different test sets available to measure the performance of dependency parsing models. The conversion that we present here is intended to provide researchers with the means to evaluate their dependency parsers on a broader set of domains for German in the hope that this will lead to better parsers. Performing well on several different genres will require parsers to learn something fundamental about the language rather than the data set that they happened to be trained on.

2. Related Work

There is a wide variety of different syntactically annotated corpora available for English. Although most parsing research for English is still done using the WSJ-section of the Penn Treebank (Marcus et al., 1993), there are many other corpora that were created to provide data for phenomena that are underrepresented in the PTB, e. g. Question Bank (Judge et al., 2006). Most of the English treebanks use constituency as their syntactic formalism. For research on dependency parsing, these annotations are commonly converted to dependencies by using head-finding rules (Magerman, 1994). The Ontonotes corpus (Weischedel et al., 2011) comprises several different domains, for example telephone conversations, magazines, etc. Recently, the English Web Treebank (Bies et al., 2012) was made available, which includes domains like email, blog posts, etc.

[1] http://www.ims.uni-stuttgart.de/˜seeker

The English Web Treebank uses an extended version of the PTB annotation guidelines to make sure that annotations between PTB and the Web Treebank are compatible, thus making it possible to evaluate models without worrying about differences in annotation. The number of different corpora for English is too large to list them all here, but generally, the situation for a researcher looking for different test sets to test her models on English is not so bad.

In this respect, English is quite the exception though. And even though German is a language with many resources compared to most other languages, the situation is far from being comparable to English. For parsing, there are two major treebanks available for German, the TiGer Treebank (Brants et al., 2002) and the TüBa/DZ (Hinrichs et al., 2004). Both have originally been annotated for constituent structure, but can be turned into dependency treebanks by using head-finding heuristics as is done in English (Daum et al., 2004; Kübler, 2008; Hajič et al., 2009; Seeker and Kuhn, 2012). Both treebanks are based on newspaper text. The SMULTRON corpus, which we use in this work, was annotated as a parallel treebank and comprises four additional domains (see next section). Recently, the NoSta-D data set (Dipper et al., 2013) was annotated for dependency syntax, which adds another five domains to the set of available treebanks, however, using an annotation scheme that is not directly compatible with TiGer or TüBa/DZ.

3. The Data Sets

The SMULTRON corpus[2] (Volk et al., 2010) was originally created to facilitate research on multilingual treebanks. It consists of four small parallel texts in five different languages (English, French, German, Spanish, Swedish; not all texts are available in all languages). All four texts are available in German and comprise the domains *novel (Sophie's World)*, *DVD manual*, *economy news*, and *alpine hiking stories*. It was annotated for part-of-speech and phrase-structures following the STTS annotation guidelines (Schiller et al., 1999) and the TiGer annotation guidelines (Albert et al., 2003), respectively. Since the SMULTRON corpus is a parallel treebank, the phrase structure annotation contains links to the annotations for the other languages.

The EuroParl data set consists of 707 sentences of the German part of the EuroParl corpus (Koehn, 2005). It was annotated by Pado and Lapata (2005) to evaluate annotation transfer systems. It was also annotated for part-of-speech and phrase structures following the STTS and TiGer guidelines. In addition, it contains annotations for semantic roles. This data set has previously been converted to dependency format and used in the CoNLL 2009 Shared Task (Hajič et al., 2009) as an out-of-domain test set.

Table 1 shows the different genres available in the two source data sets together with their respective sizes (by number of sentences).

corpus	domain	# sentences
SMULTRON	novel	529
	DVD manual	547
	economy news	518
	alpine hiking stories	1060
EuroParl700	political debates	707

Table 1: The domains in SMULTRON and EuroParl700.

3.1. Conversion

The conversion process takes a constituent tree and turns it into a dependency by traversing it bottom-up, thereby selecting for each phrase a head from the set of children of that phrase. The conversion is performed using a set of hand-written rules that specify preferences for function label and part-of-speech tag of the head word as well as the direction from where to start the search, i.e. whether to go through the children from left to right or from right to left. Table 2 displays the set of rules that are used to convert the SMULTRON data sets. Rules higher up in the table will be applied before lower ones.

The rules are very similar to the ones that were used to convert the TiGer corpus (Seeker and Kuhn, 2012). They are not exactly the same due to some small differences in the annotation: in the original TiGer annotation guidelines, prepositional and noun phrases do not have their head marked explicitly. In SMULTRON, the heads are explicitly marked with label HD. The conversion rules can rely on the explicit head marking. Remaining HD labels are replaced by the standard TiGer label NK after the conversion has finished. Another difference is that the annotators of SMULTRON decided to annotate noun phrases in prepositional phrases, which is not part of the original TiGer annotation. When converting TiGer, we used a special rule to introduce a noun phrase level for each prepositional phrase (cf. Seeker and Kuhn (2012), which is not necessary when converting SMULTRON. The SMULTRON annotation also uses a special syntactic category MPN for multi-token proper nouns, which is not part of TiGer. As can be seen in Table 2, we use the same rule to convert standard proper nouns (PN) and MPNs. Finally, the part-of-speech annotation of SMULTRON uses the PIDAT tag, which is part of the STTS tag set, but was not used when TiGer was annotated. We automatically map all PIDAT tokens to PDAT, which is the part-of-speech tag for these words in TiGer.

We convert coordination to Mel'čuk style, which puts the first conjunct as the head of the coordination and then creates a chain with the following conjuncts and coordinating conjunctions.

In order to convert the EuroParl data set, we kept the original rules used to convert TiGer and added one rule to handle the root node, which is called TOP in the EuroParl annotation (see Table 3).

As a final remark, we would like to point out that this conversion is indeed a conversion and not a manual annotation. These rules are quite specific, but they are not infallible, so that it is perfectly possible to create a linguistically undesirable annotation. This may be caused either because the rule is not exact or because the original annotation was faulty in the first place.

An important goal of this work is that the conversion of the out-of-domain data sets should be as similar as possible to

[2] http://kitt.ifi.uzh.ch/kitt/smultron

phrase	label	pos	dir
s	hd	*	*
vp	hd	*	*
vz	hd	*	*
avp	hd, ph	*	*
avp	avc	adv	right
avp	avc	fm	left
ap	hd, ph	*	*
dl	dh	*	*
aa	hd	*	*
isu	uc	*	*
pn	pnc	ne, nn, fm, trunc, appr, apprart, card, vvfin, vafin, adja, adjd, xy	right
mpn	pnc	ne, nn, fm, trunc, appr, apprart, card, vvfin, vafin, adja, adjd, xy	right
nm	nmc	nn, card	right
mta	adc	adja	right
pp	hd, ac, ph	apprart, appr, appo, proav, ne, apzr, pwav, trunc	right
pp	nk	proav	left
ch	hd	*	left
ch	uc	fm, ne, xy, card, itj	left
np	hd	nn	left
np	nk	nn	left
np	hd, nk	ne, pper, pis, pds, prels, prf, pws, pposs, fm, trunc, adja, card, piat, pwav, proav, adjd, adv, apprart, pdat	right
np	ph	nn, ne, pper, pis, pds, prels, prf, pws, pposs, fm, trunc, adja, card, piat, pwav, proav, adjd, adv, apprart, pdat	right
cac	cj	*	left
cap	cj	*	left
cavp	cj	*	left
ccp	cj	*	left
cnp	cj	*	left
co	cj	*	left
cpp	cj	*	left
cs	cj	*	left
cvp	cj	*	left
cvz	cj	*	left

Table 2: Head-finding rules for the SMULTRON data sets. First column gives the phrase category for which the head is selected, second column gives the label preference of the head (highest preference on the left), third column gives the part-of-speech tag preference (highest preference on the left), and last column gives the search direction in which to search through the daughters of the phrase (left: start with left-most daughter, right: start with right-most daughter).

phrase	label	POS	dir
top	–	vvfin, vafin, vmfin, vvimp	left

Table 3: Additional head-finding rule for the EuroParl data.

the TiGer conversion. The similarity of both conversions depends directly on the similarity between the original annotations. For all three sources (TiGer, SMULTRON, EuroParl), the original TiGer annotation guidelines were used. As we pointed out earlier, there are small differences, especially between SMULTRON and TiGer, which are mostly due to the original requirements on SMULTRON as a parallel treebank. For example, the introduction of internal NPs in PPs, which is not part of standard TiGer, allowed the original annotators to introduce more links between the different languages in SMULTRON. However, we spent time and effort to find these deviations and deal with them appropriately. Systematic differences are rather easy to remedy once identified, which is reflected in the slightly different set of rules used for the conversion (see Table 2). There are of course some remaining differences, but most of them are more likely due to inconsistencies in the original annotations in both TiGer and the out-of-domain data sets.

3.2. Error Correction

The head-finding rules of the converter are deliberately designed in a way such that the conversion will fail if there is no matching rule available. This means that contrary to many converters, there is no catch-all rule that fires if none of the other rules is applicable. Having such a last-resort rule ensures that there will be a formally correct dependency tree for each sentence, but it also effectively hides any problem in the annotation. So rather than just running the converter and forcing each structure into a dependency tree, we used a semi-automatic process: we first ran the converter on the entire data set, and afterwards manually inspected the sentences where it failed to produce a proper dependency tree. In most of these cases, we found annotation inconsistencies in the original annotation, which we corrected.

Table 4 shows a break-down of single changes that we applied to the data sets during the conversion process. Note that this does not guarantee an error-free annotation, since not all errors in the original annotation may lead to conversion failures. However, the problems that do make the converter fail could be found and corrected. In very few cases, there was no obvious annotation error but it also did not make sense to change the rule system. In these cases, we selected the head manually.

edit	count
change PoS	161
change function label	103
reattach edge	62
introduce node	10
remove node	18
new root	5

Table 4: Break-down of single edits applied to the original annotation in order to correct annotation errors.

3.3. Variation

The purpose of the following section is to illustrate differences between the in-domain data (TiGer) and the out-of-domain test sets, as well as among each other. We consider

TiGer the in-domain data because this data would normally be used to train a statistical model. We will briefly discuss unknown words, 1st and 2nd person vs. 3rd person, imperative mood and questions. These phenomena are relatively easy to show and only constitute the tip of the proverbial iceberg. They are meant to demonstrate the variation among the different text genres as well as the limitations of our standard training data sets in terms of linguistic phenomena.

Having too close a look at one's test data can be dangerous, since these data sets are meant to be used as fair test sets for evaluation. Normally, one would like to avoid knowing the test set in order to prevent any chance of overfitting. However, for improving parsing models on out-of-domain data, we need to know what makes them out-of-domain with respect to the training set.[3]

3.3.1. Unknown Word Forms

The first property we look at is the amount of unknown tokens (by their word form) that we find with respect to the training portion of the TiGer corpus. We split the corpus as follows: train/dev/test = 40,472/5,000/5,000 sentences. We use this split also later in Section 4. to train and evaluate our models.

Table 5 gives a breakdown of the fraction of unknown word forms found in the different data sets. For comparison, we include the same fractions for the in-domain test set from TiGer.

	oov %	avg. oov / sent
TiGer test set	9.96	1.83
EuroParl	13.26	3.08
novel	7.64	1.22
DVD manual	23.89	3.91
economy news	12.30	2.53
alpine stories	14.84	2.72

Table 5: Ratios of out-of-vocabulary (oov) tokens and the average number of unknown tokens per sentence.

We would normally expect the ratio of unknown tokens to be higher in out-of-domain data sets than in in-domain data sets. We find this only for three out of the five out-of-domain data sets. For the novel data set, the ratio is even lower than for the in-domain test set. The DVD manual shows the highest ratio of unknown tokens, more than 20%. The high ratio of unknown tokens in the EuroParl data set stems mainly from the fact that there is a systematic spelling difference between the EuroParl data set and the TiGer corpus: all umlauts (*ä*, *ö*, *ü*) are written as *ae*, *oe*, and *ue*. Normalizing the spelling of umlauts across both training and test data pushes the ratio of unknown tokens for the EuroParl data set down to 4.49%. As we will see later, this also has positive effects on the performance of a part-of-speech tagger and a dependency parser.

3.3.2. 1st & 2nd Person vs. 3rd Person

A typical property of newspaper text is the low frequency of 1st and 2nd person inflection compared to 3rd person, which is due to the reporting style of newspaper articles. This means that statistical models that are trained on newspaper text will generally assume a lower probability for 1st and 2nd person. These assumptions become a problem when the models are applied in an out-of-domain setting, for example to domains like conversation, online chat, or stories told from a first-person point-of-view where 1st person is much more frequent than 3rd.

Table 6 shows the frequencies and ratios for 1st and 2nd person personal pronouns compared to 3rd person personal pronouns. We counted personal pronouns (gold part-of-speech tag: PPER) rather than verbal inflection, because the data sets do not provide gold morphological annotations (yet). Personal pronouns are a closed word class, which can be enumerated. We counted capitalized *Sie*[4] as 2nd person when it did not occur at the beginning of a sentence.

	1st & 2nd vs. 3rd	% of pers. pron.
TiGer train set	1961 / 9129	17.7 / 82.3
TiGer test set	327 / 1343	19.6 / 80.4
EuroParl	554 / 325	63.0 / 37.0
novel	185 / 322	36.5 / 63.5
DVD manual	3 / 269	1.1 / 98.9
economy news	7 / 31	18.4 / 81.6
alpine stories	562 / 244	69.7 / 30.3

Table 6: Frequencies of personal pronouns by grammatical person and their fraction of all personal pronouns (PPER).

The ratios for the different data sets show that economy news has the most similar distribution of 1st & 2nd vs. 3rd pronouns compared to the in-domain test set, followed by novel. The EuroParl and alpine hiking stories data sets have the opposite distribution with about twice as many 1st & 2nd person pronouns than 3rd person pronouns. DVD manual almost exclusively uses 3rd person pronouns.

In this brief evaluation, we only look at pronouns to approximate the distribution of the different person values. However, these ratios should roughly represent also the distribution of verb forms with different person inflection. Many of these verb forms do not occur in TiGer, even though many of them may occur with 3rd person inflection. Training statistical models on newspaper text may therefore make it difficult to learn e. g. core morphosyntactic properties of German like person agreement between subjects and verbs.

3.3.3. Imperatives and Questions

Another typical feature of newspaper text, which is also due to its reporting style, is the low frequency of imperatives. Table 7 shows the frequency of verbs in imperative mood in the individual data sets, as identified by their part-of-speech tag VVIMP/VAIMP.

Imperatives are rare in newspaper, and also in all of our out-of-domain data sets except for the DVD manual. It is not surprising that manuals use imperatives to instruct

[3] Normally, one would split the data sets into development and test data for this purpose. However, these data sets are quite small and we do not think it makes much sense to split them any further.

[4] The German pronoun for politely addressing another person.

	# imperatives	% of verb forms
TiGer train set	114	0.1
TiGer test set	24	0.2
EuroParl	4	0.1
novel	5	0.4
DVD manual	183	15.2
economy news	0	0.0
alpine stories	0	0.0

Table 7: Frequencies of imperatives (VVIMP) and their fraction of all verb forms (V*).

the reader how to do something. It shows that a low frequency phenomenon in one domain may be a high frequency phenomenon in another. A second example are questions, which are of low frequency in newspapers too. The frequencies in Table 8 approximate the distribution of questions by showing the frequencies of question marks. The novel data set has a significantly higher proportion of questions compared to all other data sets.

	# questions	% of all sentences
TiGer train set	658	1.6
TiGer test set	54	1.1
EuroParl	26	3.7
novel	93	17.6
DVD manual	0	0.0
economy news	0	0.0
alpine stories	26	2.5

Table 8: Frequencies of questions and their fraction of all sentences.

Both imperatives and questions are marked in German (among other things) by a different word order compared to the standard assertional sentence. These word orders are underrepresented in newspaper text and will therefore get assigned low probabilities/scores by statistical models.

4. Experiments

In this section, we provide some baseline experiments and three small experiments to test very simple methods for avoiding most obvious mismatches. We first present experiments where we apply a standard NLP pipeline trained on TiGer to the out-of-domain test sets. For lemmatization and parsing, we use the mate-tools,[5] specifically the graph-based dependency parser described in Bohnet (2010). As a second parser, we use turboparser[6] (Martins et al., 2010), a parser based on linear programming relaxations. For jointly predicting part-of-speech and morphological features, we use MarMoT[7] (Mueller et al., 2013). Since the out-of-domain data sets do not provide gold-standard lemmata and morphological annotation, we cannot evaluate the quality of the automatic predictions, but instead only report on part-of-speech and syntax.

As the baseline experiment, we train the processing pipeline using 10-fold jackknifing on the training data and apply them to the out-of-domain data sets. We compare this baseline to an experiment where we provide MarMoT with additional lexical information from SMOR[8] (Schmid et al., 2004), a finite-state morphological analyzer for German. Morphological disambiguation has used morphological lexicons for a long time (see e. g. Hajič (2000)) since they restrict the space of possible analyses and provide information on unknown words that cannot be learned by a statistical model (e. g. the grammatical gender of nouns in German). Since out-of-domain data may come along with a higher number of unknown words (or with new usages of known words), morphological lexicons can be a valuable resource because they are not restricted to a particular training set.

	baseline	+ smor	Δ
TiGer test set	97.22	97.72	+0.50
EuroParl	93.46	94.35	+0.89
novel	94.13	95.20	+1.07
DVD manual	88.96	90.81	+1.85
economy news	91.83	91.77	−0.06
alpine stories	93.32	94.42	+1.10

Table 9: Part-of-speech accuracies for MarMoT with and without using a lexicon.

Table 9 presents the part-of-speech tagging accuracies achieved on the different data sets once without and once with the additional information provided by SMOR. In this experiment and all following ones we also provide results on the in-domain TiGer test set. As we show in Table 9, the general performance of the part-of-speech prediction is considerably lower on the out-of-domain data sets compared to the in-domain data set. This is the general pattern that occurs in out-of-domain application scenarios. Among the out-of-domain test sets, the models perform worst on DVD manual and best on novel, which is however still considerably behind the in-domain test set.

Table 9 also shows that adding additional information using a morphological lexicon does improve performance on all test sets (including in-domain) except on the economy news. Closer inspection reveals that the main problem in the economy news data is the large amount of foreign words (part-of-speech tag: FM), which are most of the time predicted as proper nouns (part-of-speech tag: NE). Most of these words are English company names or English names for operation areas like *process automation*, which had not been translated to German. However, German morphological lexicons would not cover these terms and therefore cannot help with these problems. For all other test sets, the lexicon improves results, generally to a larger extent for the out-of-domain test sets.

In Table 10, we present the parsing results in terms of labeled (las) and unlabeled accuracy (uas). Again, we compare the baseline system and the system with access to a morphological lexicon. We always show the difference in

[5] code.google.com/p/mate-tools

[6] https://www.ark.cs.cmu.edu/TurboParser

[7] http://code.google.com/p/cistern/wiki/marmot

[8] http://www.cis.uni-muenchen.de/ schmid/tools/SMOR

	TiGer test set			EuroParl			novel			DVD manual			economy news			alpine stories		
	Δpos	las	uas	Δpos	las	uas	Δpos	las	uas	Δpos	las	uas	Δpos	las	uas	Δpos	las	uas
mate		88.16	90.32		79.31	83.56		83.66	88.34		79.19	83.30		79.83	83.64		82.68	86.00
+ smor	+0.50	88.55	90.66	+0.89	80.48	84.74	+1.07	84.12	88.46	+1.85	79.82	83.53	−0.06	80.01	83.73	+1.10	83.43	86.63
turbo		88.07	90.56		79.95	84.42		84.03	88.75		78.97	83.65		79.50	83.57		82.04	85.78
+ smor	+0.50	88.51	90.95	+0.89	80.85	85.22	+1.07	84.72	89.13	+1.85	79.86	84.15	−0.06	79.58	83.41	+1.10	83.42	87.01

Table 10: Parsing performance for mate and turboparser with and without using a lexicon for predicting part-of-speech and morphology.

part-of-speech accuracy as comparison to make it easier to see the relationship between improvements in part-of-speech tagging and parsing. Keep in mind though that the lexicon also influences morphological prediction, which we cannot measure but which influences the parsing performance as well.

The two different parsers that we use for our experiments seem generally to perform equally well, with maybe a small advantage for turboparser. Both parsers give state-of-the-art results on the in-domain test set. Across the test sets, we get the same picture as for part-of-speech tagging, with the out-of-domain test sets clearly behind the performance on the in-domain test set. The differences of up to 7 percentage points are however much bigger than for part-of-speech tagging. The DVD manual data and the economy news data again seem to be more difficult test sets than the novel data and the alpine stories. The results on the EuroParl data set are on the lower end too. We have a closer look at these three data sets in the next section, where we find that simple orthographic normalization is an important step with significant impact when doing out-of-domain parsing.

4.1. Orthographic Normalization

In this section, we normalize three different orthographic phenomena in order to make the test data more similar to the training data. As we will show here in three small experiments, orthographic normalization suddenly becomes very important once we drop the assumption of consistent orthography that holds in newspaper treebanks. Orthographic normalization has already been studied intensively for different domains, e. g. historical text (Baron and Rayson, 2008) or microblogging (Han et al., 2013).

4.1.1. Umlauts

In the previous experiment, parsing results on the EuroParl data resided on the lower end of the spectrum. As we already mentioned in Section 3.3.1., there is a systematic spelling difference between TiGer and the EuroParl data, namely the spelling of German umlauts *ä*, *ö*, and *ü*, which data set are spelled *ae*, *oe*, and *ue* in the EuroParl. The same applies to German *ß*, which is spelled *ss* in the EuroParl data set. The EuroParl spelling is the standard way of encoding German umlauts and *ß* in ASCII, which is not done in TiGer. For the following experiment, we normalized the spelling in the EuroParl test set to use the standard German umlauts and *ß*.[9] We then applied the models from the previous experiment to the normalized data.

	original			normalized		
	pos	las	uas	pos	las	uas
mate	93.46	79.31	83.56	96.47	82.92	86.92
+ smor	94.35	80.48	84.74	96.63	82.95	86.79
turbo	93.46	79.95	84.42	96.47	83.21	87.32
+ smor	94.35	80.85	85.22	96.63	83.22	87.22

Table 11: EuroParl: comparison between original and normalized spelling of umlauts (ä, ö, ü) and sharp-s (ß).

Table 11 shows the differences in part-of-speech tagging and parsing when comparing the original (from Table 10) and the normalized version. Improvements of about 3 percentage points for both tasks are very high and demonstrate convincingly the importance of normalization. In the unnormalized version, most of the words containing umlauts are unknown to the statistical models. As stated in Section 3.3.1., normalizing the spelling changes the percentage of unknown words from 13.26% to 4.49%, which is only half the amount that we find in the in-domain test set.

4.1.2. Quotation Marks

We use the alpine stories to demonstrate the effect of normalizing punctuation, in this particular case quotation marks. The alpine stories use French quotation marks, namely ≪and ≫. Since these symbols never occur in TiGer, they are standardly treated as unknown words by the part-of-speech tagger and the parsers. This means that the models rely on context to decide the correct annotation, which leads to quotation marks being tagged as nouns, adjectives, foreign words etc. Out of 147 such tokens, only 7 were tagged correctly as quotation mark. Since these tags provide information to parsing, it is naturally desirable to avoid such tagging errors. In the experiment, we replaced these quotation marks with the standard symbols used in TiGer and applied the models from the first experiment.

Table 12 shows the part-of-speech tagging and parsing results on the original and the normalized alpine stories data set. Again, by normalizing the test set, we can achieve an improvement in performance for both tasks. Even though punctuation may not seem an important linguistic phenomenon it deserves attention because it provides context to other decisions.

4.1.3. Capitalization

Our third normalization experiment is concerned with the DVD manual test set. In this data set, there is a num-

[9] This cannot be done completely loss-free by automatic means. Given the size of the test set, it was not very time-consuming to decide the ambiguous cases manually.

	original			normalized		
	pos	las	uas	pos	las	uas
mate	93.32	82.68	86.00	94.06	83.47	86.71
+ smor	94.42	83.43	86.63	95.12	84.03	87.15
turbo	93.32	82.04	85.78	94.06	82.81	86.46
+ smor	94.42	83.42	87.01	95.12	84.08	87.59

Table 12: Alpine stories: comparison between original and normalized spelling of quotes (≪,≫).

ber of sentences which are written using exclusively capital letters. These are sometimes headlines, but also warnings are marked that way. As with umlauts and quotation marks, these words are treated as unknown by the statistical models since the actual word forms were never seen during training. In order to test the influence of normalizing capitalization, we use an artificial setting, where we use the gold part-of-speech tags to decide how to lowercase a word form. The reason for this is that in German, common nouns are written with a capital first letter. The heuristic that we use to normalize the data is that we lowercase every word that does not have one of the part-of-speech tags NE, XY, or FM. We keep the first letter capitalized if the word starts a sentence or if it is a common noun (part-of-speech tag: NN). This is an artificial setting since normally we would not know the part-of-speech tag of a word at this stage of processing. The reported results should therefore be interpreted as a kind of upper bound that can be achieved rather than an actual performance estimate.

	original			normalized		
	pos	las	uas	pos	las	uas
mate	88.96	79.19	83.30	92.30	82.20	85.65
+ smor	90.81	79.82	83.53	92.72	82.05	85.24
turbo	88.96	78.97	83.65	92.30	82.06	86.08
+ smor	90.81	79.86	84.15	92.72	82.05	85.91

Table 13: DVD manual: comparison between original and normalized spelling of all-caps words.

Table 13 shows the results for the third experiment. Again, improvements between 2 and 3 percentage points for all tasks demonstrate the influence of proper normalization.

One additional effect that can be observed in the first and the third experiment in this section (Tables 11 and 13) is that the improvements due to the morphological lexicon disappear after the normalization. The settings with and without the morphological lexicon perform roughly similarly, this time with a small advantage for mate parser. A possible explanation for this is the fact that in the unnormalized setting, the morphological lexicon was doing some of the work that is done by the normalization. For example, the morphological lexicon can compensate the systematic encoding of umlauts by applying the appropriate rules to the word form. However, the lexicon cannot completely take care of the problem since the information is only encoded in the lexicon features given to the part-of-speech tagger, whereas the word form stays unknown to the statistical model. An additional processing step to take care of normalization is therefore desirable.

5. Conclusion

In this work, we presented a dependency conversion of five different test sets for German, namely the SMULTRON corpus and a small part of the EuroParl corpus. The conversion was performed semi-automatically and was designed to be as close as possible to the conversion of the TiGer corpus presented in Seeker and Kuhn (2012). We described the data sets and discussed their variability among each other by means of small case studies targeting some easy to find phenomena. We also presented a baseline experiment using a standard processing pipeline and compared this to an approach where we give additional lexical information to the part-of-speech tagger. Further experiments revealed that while additional lexical resources can improve out-of-domain parsing, a standard way of normalizing text may be an important addition to the standard NLP pipeline when parsing out-of-domain data.

The presented conversion is intended to make it easy to test parsing models in an out-of-domain setting rather than always testing on the same in-domain data. Our hope is that this will support researchers in developing robust parsing models that perform equally well on different domains.

Acknowledgements

This work was funded by the Deutsche Forschungsgemeinschaft (DFG) via SFB 732 ”Incremental Specification in Context”, project D8.

6. References

Albert, S., Anderssen, J., Bader, R., Becker, S., Bracht, T., Brants, S., Brants, T., Demberg, V., Dipper, S., Eisenberg, P., Hansen, S., Hirschmann, H., Janitzek, J., Kirstein, C., Langner, R., Michelbacher, L., Plaehn, O., Preis, C., Pußel, M., Rower, M., Schrader, B., Schwartz, A., Smith, G., and Uskoreit, H. (2003). TIGER Annotationsschema. Technical Report Juli, Universität Stuttgart.

Baron, A. and Rayson, P. (2008). VARD2:a tool for dealing with spelling variation in historical corpora. In *Postgraduate Conference in Corpus Linguistics*, Birmingham, UK. Aston University, Birmingham.

Bies, A., Mott, J., Warner, C., and Kulick, S. (2012). English Web Treebank. http://www.ldc.upenn.edu.

Bohnet, B. (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.

Brants, S., Dipper, S., Hansen-Shirra, S., Lezius, W., and Smith, G. (2002). The TIGER treebank. In *Proceedings of the 1st Workshop on Treebanks and Linguistic Theories*, pages 24–41, Sozopol, Bulgaria.

Daum, M., Foth, K., and Menzel, W. (2004). Automatic transformation of phrase treebanks to dependency trees. In *Proc. 4th Int. Conf. on Language Resources*.

Daume III, H. (2007). Frustratingly easy domain adaptation. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 256–263, Prague, Czech Republic, June. Association for Computational Linguistics.

Dipper, S., Lüdeling, A., and Reznicek, M. (2013). NoSta-D: A corpus of German non-standard varieties. In Zampieri, M., editor, *Non-Standard Data Sources in Corpus-Based Research*, pages 69–76. Shaker.

Gildea, D. (2001). Corpus variation and parser performance. In Lee, L. and Harman, D., editors, *Proceedings of EMNLP*, pages 167–202, Pittsburgh, PA, USA.

Hajič, J. (2000). Morphological Tagging: Data vs. Dictionaries. In *Proceedings of the 6th ANLP Conference / 1st NAACL Meeting*, pages 94—101, Seattle, Washington. Association for Computational Linguistics.

Hajič, J., Ciaramita, M., Johansson, R., Kawahara, D., Martí, M. A., Màrquez, L., Meyers, A., Nivre, J., Padó, S., Stepánek, J., Stranák, P., Surdeanu, M., Xue, N., and Zhang, Y. (2009). The CoNLL-2009 shared task: Syntactic and Semantic dependencies in multiple languages. In *Proceedings of the 13th CoNLL Shared Task*, number June, pages 1–18, Boulder, Colorado.

Han, B., Cook, P., and Baldwin, T. (2013). Lexical normalization for social media text. *ACM Trans. Intell. Syst. Technol.*, 4(1):5:1–5:27, February.

Hinrichs, E., Kübler, S., Naumann, K., Telljohann, H., and Trushkina, J. (2004). Recent developments in linguistic annotations of the TüBa-D/Z treebank. In *Proceedings of the Third Workshop on Treebanks and Linguistic Theories*, pages 51–62, Tübingen, Germany.

Judge, J., Cahill, A., and van Genabith, J. (2006). Questionbank: Creating a corpus of parse-annotated questions. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 497–504, Sydney, Australia, July. Association for Computational Linguistics.

Koehn, P. (2005). Europarl: A Parallel Corpus for Statistical Machine Translation. In *Conference Proceedings: the tenth Machine Translation Summit*, pages 79–86, Phuket, Thailand. AAMT, AAMT.

Kübler, S. (2008). The PaGe 2008 shared task on parsing German. In *Proceedings of the Workshop on Parsing German*, pages 55–63, Morristown, NJ, USA. Association for Computational Linguistics.

Magerman, D. M. (1994). *Natural language parsing as statistical pattern recognition*. Ph.D. thesis, Stanford.

Marcus, M. P., Marcinkiewicz, M. A., and Santorini, B. (1993). Building a large annotated corpus of English: The Penn Treebank. *Computational Linguistics*, 19(2):313–330.

Martins, A., Smith, N., Xing, E., Aguiar, P., and Figueiredo, M. (2010). Turbo Parsers: Dependency Parsing by Approximate Variational Inference. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 34–44, Cambridge, MA, October. Association for Computational Linguistics.

Mueller, T., Schmid, H., and Schütze, H. (2013). Efficient higher-order CRFs for morphological tagging. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 322–332, Seattle, Washington, USA, October. Association for Computational Linguistics.

Pado, S. and Lapata, M. (2005). Cross-linguistic projection of role-semantic information. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pages 859–866, Vancouver, British Columbia, Canada, October. Association for Computational Linguistics.

Schiller, A., Teufel, S., and Stöckert, C. (1999). Guidelines für das Tagging deutscher Textcorpora mit STTS (Kleines und großes Tagset). Technical Report August, Universität Stuttgart.

Schmid, H., Fitschen, A., and Heid, U. (2004). SMOR: A German Computational Morphology Covering Derivation, Composition and Inflection. In *Proceedings of the 4th International Conference on Language Resources and Evaluation.*

Seeker, W. and Kuhn, J. (2012). Making ellipses explicit in dependency conversion for a german treebank. In Calzolari, N., Choukri, K., Declerck, T., Doğan, M. U., Maegaard, B., Mariani, J., Odijk, J., and Piperidis, S., editors, *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC-2012)*, pages 3132–3139, Istanbul, Turkey, May. European Language Resources Association (ELRA). ACL Anthology Identifier: L12-1088.

Volk, M., Göhring, A., Marek, T., and Samuelsson, Y. (2010). SMULTRON (version 3.0) — The Stockholm MULtilingual parallel TReebank. http://www.cl.uzh.ch/research/paralleltreebanks_en.html. An English-French-German-Spanish-Swedish parallel treebank with sub-sentential alignments.

Weischedel, R., Palmer, M., Marcus, M., Hovy, E., Pradhan, S., Ramshaw, L., Xue, N., Taylor, A., Kaufman, J., Franchini, M., El-Bachouti, M., Belvin, R., and Houston, A. (2011). OntoNotes release 4.0.