Transferring PoS-tagging and lemmatization tools from spoken to written Dutch corpus development

Antal van den Bosch*, Ineke Schuurman†, Vincent Vandeghinste†

*ILK / Dept. of Language and Information Science
Tilburg University
P.O. Box 90153, NL-5000 LE Tilburg, The Netherlands
Antal.vdnBosch@uvt.nl
† Centre for Computational Linguistics
Katholieke Universiteit Leuven
Maria-Theresiastraat 21, B-3000 Leuven, Belgium
{Ineke.Schuurman,Vincent.Vandeghinste}@ccl.kuleuven.ac.be

Abstract

We describe a case study in the reuse and transfer of tools in language resource development, from a corpus of spoken Dutch to a corpus of written Dutch. Once tools for a particular language have been developed, it is logical, but not trivial to reuse them for other types or registers of the language than the tools were originally designed for. This paper reviews the decisions and adaptations necessary to make this particular transfer from spoken to written language, focusing on a part-of-speech tagger and a lemmatizer. While the lemmatizer can be transferred fairly straightforwardly, the tagger needs to be adaptated considerably. We show how it can be adapted without starting from scratch. We describe how the part-of-speech tagset was adapted and how the tagger was retrained to deal with written-text phenomena it had not been trained on earlier.

1. Introduction

Since 1998, the transnational Dutch Language Union[1] has sponsored the development of corpora of spoken and written Dutch. The Spoken Dutch Corpus project (CGN, 1998–2003)[2] has produced a corpus of contemporary standard Dutch as spoken by adults in The Netherlands and Flanders, totaling about 9 million words. After being fully orthographically transcribed, the 9-million-word corpus was PoS-tagged and lemmatized, and manually corrected. The new follow-up project Dutch Language Corpus Initiative (D-Coi, from 2005, funded by the Dutch Language Union's STEVIN program)[3] aims at creating a 50-million-word corpus of contemporary written Dutch. The project has been set up with the intention to reuse as much as possible from the Spoken Dutch Corpus, such as the part-of-speech tag set, but also annotation tools and protocols for human correctors. The question that arises then, is how to cope with the differences between handling (symbolic representations of) spoken language, and written text.

Spoken and written language differ in at least the following aspects:

- Spoken language is essentially produced as sound waves consisting of connected speech sounds occasionally halted by stops and pauses, and can be transcribed into symbolic alphabets representing phonemic and prosodic structures. It can also be transcribed into an orthographical representation, using regular letters, ways to segment words, and adopting some rules for dealing with, e.g., fragmented words. Written language, in contrast, uses orthography per definition; it segments words by spaces, using the orthographic alphabet in both the capitalized and lower-case register meaningfully.
- Written language contains *sentences*, while spoken language can be more arbitrarily segmented into *utterances*, for example at the onset and end of a speaker's turn, or when a speaker takes a pause. Both units may contain virtually anything, but in general sentences tend to contain one main clause and an optional number of subordinate clauses, while spoken utterances can contain fragmented sentences. Related to this difference, the general tendency of written text is that it contains grammatical sentences, while many utterances in spoken language are not grammatical in the strict sense (e.g. of having to contain at least one main verb, verbs and subjects agreeing in number, etc.).
- Spoken utterances are devoid of punctuation, which are present in written text. One group of punctuation markers in written text, including the period, the comma, the colon, and the semicolon are actually correlated with speaking phenomena such as prosodic breaks in Dutch (Marsi et al., 2003), which have been partially annotated in the Spoken Dutch Corpus. Another group of punctuation markers, the quotes, denote meta-information on citations, quotes, and turn-taking, which are typically not available in speech, except sometimes in the literal sense of actual turn taking.
- Written sentences are typically devoid of disfluencies, while spoken utterances are full of them (Lendvai et al., 2003). Disfluencies include stutters, repetitions, fragmented words, filled pauses ("uhm..."), and elongated vowels.

The logical consequence of transferring tools, protocols, and tag sets from spoken to written Dutch is that all of them

[1] URL: http://taalunieversum.org/en/
[2] URL: http://lands.let.kun.nl/cgn/ehome.htm
[3] URL: http://lands.let.ru.nl/projects/d-coi/

are changed to reflect the absence of all above-mentioned phenomena that exclusively occur in spoken utterances, and the presence of certain elements that exclusively occur in written text. In this paper we outline the changes made to a combined part-of-speech tagger and lemmatizer. In Section 2 we describe how the tagger and lemmatizer were developed originally for the Spoken Dutch Corpus. Subsequently we describe how they are adapted to the new written Dutch corpus in Section 3. The memory-based tagger-lemmatizer is used in both corpus annotation projects to automatically process raw material, producing part-of-speech tagged and lemmatized data, which is subsequently checked for correctness by human annotators. We outline how we optimize this process in terms of time spent by the annotators in Section 4. We summarize our conclusions and recommendations in Section 5.

2. CGN: The Spoken Dutch Corpus

The part-of-speech tagset developed for CGN[4] consists of 316 tags. It closely follows the Algemene Nederlandse Spraakkunst (ANS), and conforms to the EAGLES guidelines.

Part-of-speech tagging of CGN was performed automatically by an ensemble of taggers trained on different tagsets and corpora, and employing different machine learning algorithms, with a meta-tagger on top that learned to integrate the different tagger outputs (Van Halteren et al., 2001). A subset of the combined sub-taggers was trained (and retrained at regular intervals) on the growing CGN corpus itself, while the other taggers were static existing taggers for written Dutch (Zavrel and Daelemans, 2000) using different tag sets. Performance (average tagging accuracy) on random 10% held-out test data taken from the new corpus grew from an initial 94.2% to 97.1% at the last training. Figure 1 visualizes the learning curve of the meta-tagger along with those of the retrained sub-taggers participating in the combination. Two of the sub-taggers, the Brill tagger (Brill, 1995) and a maximum-entropy-based tagger, MXPOST (Ratnaparkhi, 1996), were not used throughout the entire period since it became to time-intensive to retrain them. The other two sub-taggers, a memory-based tagger (Daelemans et al., 1996) and a hidden-markov-based approach, Trigrams 'n' Tags (Brants, 2000), were retrained until the end of the project.

As can be seen in Figure 1, meta-learning yielded a major performance boost over a set of heterogeneous taggers in the early stages of the project, when between 10,000 and 100,000 tagged words were available for training. While some of the sub-taggers still performed at under 90% tagging accuracy on unseen data when 100,000 manually corrected tagged words were available, the meta-learning tagger had a fairly stable performance above 94% even at the early stage when only 10,000 training words were available. Later in training, meta-learning continued to contribute mildly over the scores of the best sub-taggers, but clearly the scores of the taggers almost converged.

[4]See Van Eynde (2004), Part of Speech Tagging and Lemmatization (CGN), available from URL http://lands.let.kun.nl/cgn/ .

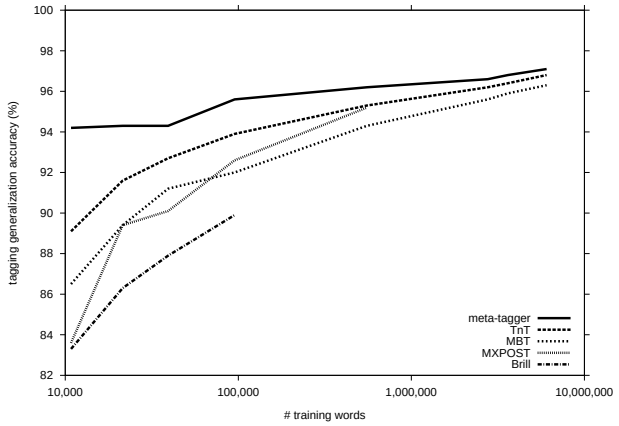

Figure 1: Learning curves of the CGN meta tagger (top solid line) and its component sub-taggers (bottom dashed lines) with increasing amounts of training material. The x-axis has a logarithmic scale.

Parallel to tagging, a separate module was developed that computed the lemmas of the words in all transcribed texts. The CGN lemmatizer was based on the lexicon developed for the corpus, but did not yet cover the new data to be annotated. For words already stored in the lexicon, the lemmatizer simply retrieved all possible known lemmatizations. For unknown words, it followed a lemmatization procedure using memory-based learning, based on Van den Bosch and Daelemans (1999).

Finally, a post-processing module integrated the output of the tagger and the lemmatizer, listing for each word the most probable tag, its corresponding lemma, and a likelihood estimation of this tag extracted from the metatagger's output. For use in the interactive annotation tool used by the human correctors, all other less likely tag-lemma tuples with their respective likelihood were output as well. The task of the correctors was to select the contextually appropriate tag-lemma combination.

3. D-Coi: Dutch Corpus Initiative

In D-Coi (2005–2006), aimed at delivering a 50-million word corpus of contemporary written Dutch, PoS-tagging and lemmatization will be provided for the whole 50-million-word corpus, but only a subcorpus of 500,000 words will be manually verified and corrected. Manually established accuracies of the tagger-lemmatizer on this subcorpus will give a reasonable estimate of the error residue in the remaining unchecked 49,500,000 words, and will pinpoint potential "hot spots" of typical recurring hard cases in Dutch tagging, which may also be the basis of a targeted manual correction phase focusing on these cases in the larger corpus. The tagset used in D-Coi is the same as the one used in CGN, be it that a few tags have been added to handle phenomena not occuring in spoken language (Van Eynde, 2006): abbreviations and symbols.

Certain tags that already existed in the original tag set now cover new phenomena that differ from their original use. One example is the tag for punctuation markers, which was only used in CGN to tag the artificial end-of-utterance markers (the period, the question mark, and ". . ." for grammatically unfinished sentences), but is now used for regular

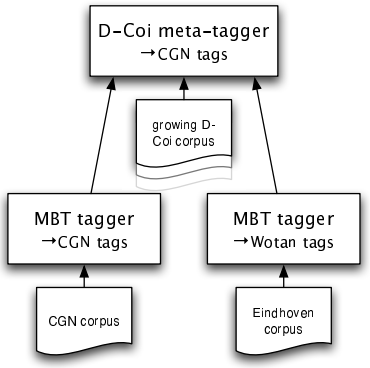

Figure 2: Schematic visualization of the architecture of the meta-tagger for the D-Coi project. Two static sub-taggers (bottom) feed their predictions to a retrainable meta-tagger (top) which also uses the growing D-Coi corpus as training material. The arrows indicate the type of tag set produced by each tagger.

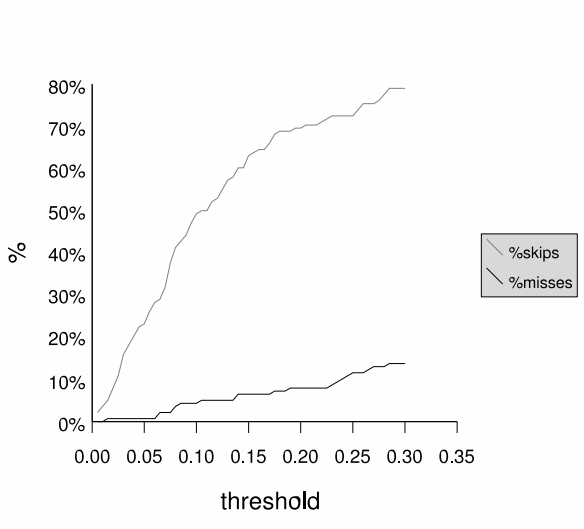

Figure 3: Effects of thresholding the cases presented to the human annotator for correction, on the percentage of cases skipped, and the percentage of PoS-tagging errors missed.

punctuation markers. Another example is the set of tags for written out numbers, as they always appear in CGN, which will be reused for numbers represented by digits. As the new D-Coi project unfolds, there may be the additional necessity to include tags in case new types of language are included (SMS, chat sessions).

The D-Coi tagger, visualized in Figure 2, borrows its overall structure roughly from the meta-tagger used in the original CGN project (Van Halteren et al., 2001), in the sense that it is an ensemble of taggers: two sub-taggers produce output for one meta-tagger. The two sub-taggers are (1) a memory-based tagger (Daelemans et al., 1996) trained on the full 9-million-word CGN corpus, estimated to be about 97% correct when tested on unseen spoken data, and (2) a memory-based tagger trained on the Eindhoven corpus (thus far the largest tagged corpus of written Dutch) tagged with the Wotan tagset (Berghmans, 1995), which was already one of the static sub-taggers in the CGN meta-tagger, and is estimated to be about 95% correct on unseen written data (Daelemans et al., 1996). The new D-Coi meta-tagger is initially trained on a small bootstrap sample of newspaper text, containing 42,912 words in 2,896 sentences, tagged (automatically, and manually corrected) with the new D-Coi tagset. Aside from being trained on this bootstrap sample of text, and being retrained in the future on growing chunks of the D-Coi corpus, in predicting tags on new text the meta-tagger also takes into account the predictions of the two sub-taggers on that text. In other words, the meta-tagger bases its final prediction on a rich contextual representation of a word in context, plus the predictions of two already quite accurate sub-taggers.
While the written-Dutch sub-tagger can be expected to produce around 95% correct tags on new written text, the CGN tagger trained on spoken data will probably not be 97% correct, since it is not familiar to punctuation and capitalization. As a first estimate of its actual performance, we in fact measured a 88.7% tagging accuracy (93.3% on known words, and a low 35.8% on unseen words) on the first batch of 38,934 manually-corrected tagged words for the D-Coi corpus.

4. High-volume manual correction: Focusing on suspect tags

The quality of the tagger–lemmatizer makes it hard to find the few mistakes left, when looking through them one by one. We are therefore deploying tools that focus on suspect tags only, identified by a low confidence value. This procedure works along the following path.
The output of the tagger consists of PoS-tagged files, containing all possible tags for each token, together with the probability of that tag. We developed a tool for the manual correction of these automatically generated PoS-tagged files. This tool takes a PoS-tagged file as input, together with a threshold value. It presents the human annotator only with those cases where more than one possible tag has an above-threshold probability. All other cases where more than one tag is generated by the tagger, or those cases where only one tag is generated, are not presented to the annotator, resulting in a markedly lower workload.
We performed a small experiment to determine at which value we best set the threshold. As Figure 3 shows, a threshold value of 0.06 results in a reduction of the number of decisions to be made by the human annotator with 28%, while skipping a mere 1% of errors which are not presented to the annotator.

This shows that using a well trained tagger that is continuously retrained, we can manually check increasingly higher amounts of data in the same time, missing hardly any errors.
Besides this tuned thresholding method, we use the following methods to correct errors in a post-hoc phase:

Checking against a blacklist. We regularly check all manually corrected material to a blacklist of typi-

cal errors made by the tagger, particularly on multi-word named entities (the tagger uses different tags for single-word proper nouns and multi-word named entities, so many words can be tagged as both), and high-frequency ambiguous function words such as *dat* (*that*, having the same ambiguity as in English) which the tagger sometimes tags incorrectly yet with high confidence.

Feeding back errors from shallow parsing modules. Tagging errors are known to cause further errors in automatic shallow and full parsing. Applying a phrase chunker, for example, and correcting that, typically reveals PoS-tagging errors, which can be fed back as bug reports for manual correction.

5. Conclusions

There is no principled problem in transferring a tag set, an accompanying tag set annotation protocol, and an automatic tagger-lemmatizer from being targeted at spoken utterances of a particular language to written texts in the same language. However, the following changes need to be applied with care:

- The tag set should be adapted to capture phenomena exclusive to written text: punctuation, abbreviations, and symbols.
- Automatic taggers used to aid human annotators, or used for large-scale automatic tagging of new text, should be retrained. A tagger trained on transcribed spoken data will function rather badly on written text not only because it does not recognize new types of tokens, but also because it is not familiar with capitalization of sentence-initial words, if that was not part of the transcription protocol.

To retrain the tagger, it will be necessary to collect a small bootstrap corpus of annotated written text (for which we would recommend a minimum size of 10,000 tagged words, as used on the outset of CGN), but this modest start is backed up by rich information from existing taggers. The same reasoning proposed and followed by Van Halteren et al. (2001), to bootstrap a tagger by combining several heterogenous taggers for the same language and a small bootstrap of the new type of data, can be applied here; we have described a combined tagger that merges the predictions of (1) an existing part-of-speech tagger for written Dutch that uses a different tag set and (2) the part-of-speech tagger trained on the entire Spoken Dutch corpus, and which uses these two predictions as features in a meta-tagger that weighs in the votes of these two taggers to make its own decision. The new second-stage tagger will become increasingly less dependent on the votes of the two sub-taggers, as it is continuously retrained on increasing amounts of human-corrected data of the new written-text corpus in the new tag set.

Acknowledgements

This research is funded by STEVIN, a Dutch Language Union (Taalunie) programme[5]), as part of the D-Coi (Dutch Corpus Initiative) project. We thank the D-Coi project partners for useful discussions and feedback, and to Jakub Zavrel, Walter Daelemans, and Frank van Eynde for the groundwork on the original part-of-speech tag set and tagger for the Spoken Dutch Corpus.

[5]URL: http://taalunieversum.org/taal/technologie/stevin/

6. References

J. Berghmans. 1995. Wotan - een probabilistische grammatikale tagger voor het Nederlands. Master's thesis, TOSCA Research Group, University of Nijmegen, Nijmegen, The Netherlands.

T. Brants. 2000. TnT – a statistical part-of-speech tagger. In *Proceedings of the 6th Applied NLP Conference, ANLP-2000, April 29 – May 3, 2000, Seattle, WA*.

E. Brill. 1995. Transformation-based error-driven learning and natural language processing: A case study in part of speech tagging. *Computational Linguistics*, 21(4):543–565.

W. Daelemans, J. Zavrel, P. Berck, and S. Gillis. 1996. MBT: A memory-based part of speech tagger generator. In E. Ejerhed and I. Dagan, editors, *Proceedings of the Fourth Workshop on Very Large Corpora*, pages 14–27. ACL SIGDAT.

P. Lendvai, A. Van den Bosch, and E. Krahmer. 2003. Memory-based disfluency chunking. In *Proceedings of Disfluency in Spontaneous Speech Workshop (DISS'03)*, pages 63–66.

E. Marsi, M. Reynaert, A. Van den Bosch, W. Daelemans, and V. Hoste. 2003. Learning to predict pitch accents and prosodic boundaries in Dutch. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, pages 489–496, New Brunswick, NJ. ACL.

A. Ratnaparkhi. 1996. A maximum entropy part-of-speech tagger. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing, May 17-18, 1996, University of Pennsylvania*.

A. Van den Bosch and W. Daelemans. 1999. Memory-based morphological analysis. In *Proceedings of the 37th Annual Meeting of the ACL*, pages 285–292, San Francisco, CA. Morgan Kaufmann.

F. Van Eynde. 2004. *Part of Speech Tagging en Lemmatisering.* Protocol for the Annotators in the Spoken Dutch Corpus. http://www.ccl.kuleuven.be/Papers/POSmanual_febr2004.pdf

F. Van Eynde. 2006. *Part of Speech Tagging en Lemmatisering.* Protocol for the Annotators in D-Coi. Project internal document

H. Van Halteren, J. Zavrel, and W. Daelemans. 2001. Improving accuracy in word class tagging through combination of machine learning systems. *Computational Linguistics*, 27(2):199–230.

J. Zavrel and W. Daelemans. 2000. Bootstrapping a tagged corpus through combination of existing heterogeneous taggers. In *Proceedings of the Second International Conference on Language Resources and Evaluation (LREC 2000)*, pages 17–20.