Unsupervised Induction of Part-of-Speech Information for OOV Words in German Internet Forum Posts

Jakob Prange and **Stefan Thater** and **Andrea Horbach**
Department of Computational Linguistics
Saarland University
Saarbrücken, Germany
{jprange,stth,andrea}@coli.uni-saarland.de

Abstract

We show that the accuracy of part-of-speech (POS) tagging of German Internet forum posts can be improved substantially by exploiting distributional similarity information about out-of-vocabulary (OOV) words. Our best method increases the accuracy by +15.5% for OOV words compared to a standard tagger trained on newspaper texts, and by +12.7% if we use an already adapted tagger.

1 Introduction

A major challenge in the automatic linguistic processing of data from computer-mediated communication (CMC) is often the lack of appropriate training material. Tools like part-of-speech (POS) taggers are usually trained on and optimized for edited texts like newspaper articles, and their performance decreases substantially when applied to out-of-domain CMC data. The tagger used in our study, for instance, achieves an accuracy of 97.2% when trained on and applied to German newspaper text; when applied to posts from an Internet forum, performance goes down to 85.0%.

One important reason for this decrease in performance is that CMC texts often contain out-of-vocabulary (OOV) words which the tagger has not seen during training. Consider the following example from the Internet forum *www.chefkoch.de*:

(1) Bei mir gab ***kabeljau ihh*** also manche ***fische*** mag ich ***irklich*** nicht ***aba rollmops*** mit **gebackene** ***kartoffeln*** und das ist **leckerer**!

The words in boldface are unknown to the tagger. They range from misspellings (*[w]irklich*), action words or interjections (*ihh*) to creative new word formations or deliberate orthographical variation (*aba* instead of *aber*) up to words that are perfectly acceptable but were not covered in the training material (*leckerer*) due to domain differences between test and training data. Words that are mis-tagged by an out-of-the box tagger model are printed in italics. We can see that, in this case, the mis-tagged words are a subset of the unknown words. Apart from this example, the frequency of mis-tagging is generally high and the percentage of mis-taggings is dramatically higher within the unknown words.

In this paper, we explore different methods to automatically induce possible POS tags for OOV words and compare different ways to exploit this information in a POS tagger. More precisely, we explore the idea that distributionally similar words tend to belong to the same lexical class and thus their POS tags can be used to induce possible POS tags of OOV words. We evaluate several ways of integrating this information into a POS-tagger: As a post-processing step, as an additional lexicon of a HMM-based tagger and as features in a CRF-based tagger. Our best approach increases the accuracy for OOV words by +15.5% for a tagger trained on standard newspaper text, and by +12.7% for an already adapted tagger.

2 Related Work

The problem that CMC texts usually contain many OOV words can be addressed in several ways. One can normalize the input text by mapping OOV words to known words in a preprocessing step, correct the POS tags of OOV words after tagging in a post-processing step, or adapt the tagger itself so that additional knowledge about possible POS tags of OOV words can be used directly during tagging.

The first two options have been explored *e.g.* by Gadde et al. (2011), who use word clusters based on string similarity to relate OOV words to known words and obtain an improvement of 4.5% over the baseline tagger on a small SMS corpus.

The third option has been investigated, amongst others, by Rehbein (2013), who trains a CRF-based

Proceedings of the 2nd Workshop on Natural Language Processing for Computer-Mediated Communication / Social Media
pages 38–43, University of Duisburg-Essen, September 28, 2015. ©2015 GSCL

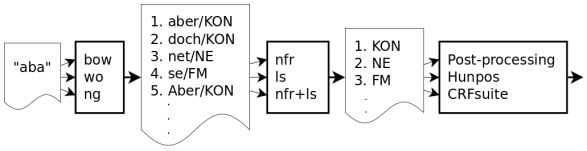

Figure 1: Example run of our pipeline with the OOV word "aba" ("aber").

tagger for German Twitter tweets on features derived from word clusters, an automatically created dictionary for OOV words and additional out-of-domain training data. The tagger achieves an accuracy of 89% on a corpus of 506 German tweets. (See Owoputi et al. (2013) for using cluster features for English data.)

While we also use a CRF-based tagger in our experiments, our approach is more closely related to the work of Han et al. (2012), who use a combination of distributional and string similarity to induce a normalization dictionary for microtexts from Twitter. The main difference is that we use the normalization dictionary only indirectly to learn possible POS tags for OOV words.

3 Our Approach

The key idea underlying our approach is that distributionally similar words tend to belong to the same lexical class and thus their POS tags can be used to induce possible POS tags of OOV words. Figure 1 describes the workflow of our approach in more detail: Given an OOV word such as *aba*, we compute the list of 20 distributionally most similar known words together with their POS tags. Based on this list of similar words we then create a lexicon that lists possible POS tags of OOV words, which we use to increase tagging accuracy of OOV words in different ways.

Distributional models. We consider three different distributional models to compute similarity scores, which we train using the *chefkoch* dataset described in Section 4 below. We tag the dataset using the *hunpos* POS tagger (Halácsy et al., 2007) trained on the *Tiger* corpus (Brants et al., 2004) and use a sliding window approach to count frequencies of context words, using a fixed window size of ± 2 words around the target word. We restrict ourselves to contexts where all context words are known to the tagger; the target word itself can be OOV, in which case we replace the POS tag assigned by the tagger by the pseudo tag *X*.

We consider (i) a standard bag-of-words model (*bow*), (ii) a variant of the bow model where context words are indexed by their relative position to the target word (*wo*), and (iii) a model where we use 5-grams of the form $\langle t_1, t_2, *, t_3, t_4 \rangle$, where the t_i are the POS-tags of the context words (*ng*). In all cases, we use PMI scores derived from the frequency counts as weights in the word vectors.

POS-Lexicon. In order to induce a ranked list of possible POS tags of OOV words, we first compute a *candidate list* containing the 20 known words with the highest similarity scores to the OOV word, taking scalar product between the word-vectors of the respective model (*bow*, *wo*, *ng*) as similarity measure. Then, we extract all POS tags that occur in the candidate list and rank the tags using different methods. We report results for the following approaches:

n-first-ratio (nfr): POS tags are ranked based on the ratio of their frequency in the candidate list and the index at which they first occur.

Levenshtein distance (ls): POS tags are ranked based on the Levenshtein distance of the corresponding word in the candidate list to the OOV word; if a POS tag occurs several times in the candidate list, we take the value for the word with minimal distance.

nfr+ls: The two weights assigned to POS labels by the algorithms above are normalized and combined linearly.

We use this ranking to induce a lexicon that lists possible POS tags of OOV words. In the experiments, we consider two variants, one which lists only the highest ranked POS tag and one which lists the three best POS tags.

Taggers. We consider two taggers in our experiments: The *hunpos* tagger already mentioned above, which is based on Hidden Markov Models, and a re-implementation of Rehbein (2013)'s CRF tagger using the CRFsuite package (Okazaki, 2007). The list of possible POS tags for OOV words can be used directly as a "morphological lexicon" in the *hunpos* tagger; the tagger uses the POS tags in this lexicon to limit the search space when emission probabilities for OOV words are estimated. In order to give the distributional information to the CRF tagger, we expand a baseline feature set (Rehbein, 2013) by the top 1 and top 3 suggested POS labels, respectively, for OOV words

feature	description	example
wrd	word form	mann
len	word length	4
cap	word capitalized?	false
upper	number upper case	0
digit	number digits	0
sym	number other non-chars	0
pre 1	first char	m
⋮	⋮	⋮
pre n	first n chars	
suf 1	last char	n
⋮	⋮	⋮
suf n	last n chars	
simpos	top n POS suggestions	⟨NN, PIS, PPER⟩

Table 1: Feature set used for experiments with CRF.

(see Table 1); for known words we take the most frequent POS label(s) of the word in the training set.

4 Experiments and Results

We train the distributional models using forum articles downloaded from the German online cooking platform *www.chefkoch.de*. This dataset has been used in previous work by Horbach et al. (2014) and consists of about half a billion tokens from forum posts about a variety of daily-life topics. A small subset of $12,337$ tokens comes with manually annotated POS information. Following previous work, we use two thirds ($8,675$ tokens) of the annotated subset as gold standard for the evaluation and one third as additional training material to re-train the tagger (see Experiment 4). The gold standard contains $1,500$ OOV tokens.

The manual annotations use a CMC-specific extension of the STTS tagset (Schiller et al., 1999) proposed by Bartz et al. (2014), covering CMC specific phenomena such as contractions, emoticons or action words. About 4% of the OOV tokens in the gold standard use tags from the extended tagset, which cannot be predicted correctly in our first three experiments.

Experiment 1. Our first experiment compares the three distributional model variants against each other. We tag the test set using the *hunpos* tagger trained on standard newspaper text (*Tiger* corpus) and then replace the POS tags of all OOV words by the POS tag of the word in the candidate list with the highest distributional similarity (*hs*) according to the respective model in a postprocessing step (*pp*).

	all	IV	OOV
baseline	85.0	93.1	46.6
bow	85.3	93.1	48.9
wo	86.6	93.1	56.7
n-gram	**87.2**	93.1	**59.9**

Table 2: Accuracy of the baseline tagger and combinations with different distributional models.

As Table 2 shows, all three distributional models achieve an improvement over the *hunpos* tagger (baseline). The difference to the baseline is small for the *bow* model, but both the *wo* and the *n-gram* model achieve substantial improvements of +10.1% and +13.3%, respectively, for OOV words. The good performance of the *n-gram* model might be surprising as n-gram information is also used directly by the tagger. The added value from the distributional model is, however, that it is trained on a much larger corpus, and abstracts away from the individual context of an OOV word and considers all contexts of this word in the complete training corpus.

Experiment 2. Next, we evaluate the effect of the methods used to rank the POS tags in the induced POS lexicon. Again, we replace the POS tags of OOV words predicted by the tagger in a postprocessing step, but this time using the tag that is ranked highest by each of the three methods considered here, instead of just the distributionally most similar one.

Table 3 shows the results. Levenshtein distance does not improve tagging performance over the *hs* result in our first experiment. However, the *n-first ratio* produces a substantial improvement, and the combination of both methods gives an additional small improvement, showing that these two methods complement each other. The approaches which use the *n-gram* model give the best results, with an improvement of +15.5% on OOV words compared to the baseline. *Upper bound* shows how often the correct POS tag occurs at least once in the candidate list in the first place. We can see that the *nfr+ls* ranking method performs quite well wrt. this upper bound; at the same time, we see that in around one third of the cases the candidate list does not contain the correct POS tag, which obviously

model	hs	nfr	ls	nfr+ls	upper bound
bow	48.9	53.5	48.1	55.3	67.0
wo	56.7	59.7	55.9	60.5	68.7
n-gram	59.9	61.4	57.7	**62.1**	67.7

Table 3: Accuracy of different ranking methods for OOV words.

leaves room for future improvements.

Experiment 3. The results obtained in our first two experiments are quite encouraging, but the method of replacing the POS tag of an OOV word with the highest ranking alternative in a post-processing step is somewhat unsatisfactory, as it does not use potentially helpful information of the context in which the target OOV word occurs. An OOV word will always get the same new POS label, even if the word is ambiguous, and known words in the context cannot benefit from context effects of a correct tag for an OOV word.

To overcome this problem, we use the induced POS lexicon as a "morphological lexicon" for the *hunpos* tagger considering the 3 highest ranked POS tags as ranked by *nfr+ls*. When the tagger sees an OOV word, it uses one of the tags listed in this lexicon. We also consider a re-implementation of the CRF-tagger used by Rehbein (2013) in this experiment, where we add the suggested POS labels to a standard CRF feature set.

Surprisingly, neither *hunpos* nor our CRF-tagger profit from this additional information (see Table 4). To the contrary, the performance decreases substantially. However, if we consider only the highest ranked POS tag (*top 1*), we do get a small improvement for *hunpos* over the *pp* baseline(s), ranging between $+0.2\%$ and $+0.3\%$. These results show that the context does not help in picking the correct POS tag among the three candidates listed in the *top 3* lexicon, but forcing the tagger to use the highest-ranked POS tag for OOV words (*top 1*) has a positive effect on the tagging accuracy of the words in the OOV word's context.

Experiment 4. Our final experiment investigates whether a similar performance gain can be achieved when we use a tagger model that has already been adapted to CMC data. We follow Horbach et al. (2015) and use one third of the manually annotated subset of the *chefkoch* corpus in addition to the *Tiger* corpus to train the *hunpos* tagger, reaching a new baseline accuracy of 91.5% . We tag the complete *chefkoch* corpus using this adapted tagger model and train our distributional models on this dataset. Thereby we gain the ability to retrieve also POS tags that only occur in the extended STTS tagset.

	pp	hunpos top1	crfsuite top3
baseline	91.5 (69.4)	91.5 (69.4)	90.8 (72.1)
bow	92.1 (75.2)	92.2 (75.2)	92.7 (78.4)
wo	92.8 (81.3)	93.0 (81.3)	93.1 (81.4)
n-gram	92.9 (**82.1**)	93.1 (**82.1**)	**93.2** (81.9)

Table 5: Results of experiments with already adapted training data. In parenthesis accuracy on unknwon words.

Table 5 shows the results. We observe similar tendencies in performance compared to the previous experiment. The overall best performance is achieved by the *n-gram* model, followed by *wo* and *bow*. Interestingly, the CRF tagger achieves with 93.2% the best overall result ($+1.7\%$ over the *hunpos* baseline and $+2.4\%$ over the CRF baseline) although it does not reach the performance of *hunpos* on OOV words.

The relative improvements over the baseline(s) are a bit smaller. One reason for this is that the adapted tagger model covers some of the most frequent OOV words in the whole *chefkoch* corpus so that these frequent and presumably easier cases for the distributional model do not need to be handled any more. Another reason is that some tags from the extended STTS tagset, specifically emoticons, often appear in syntactically not integrated positions and show high distributional similarity with punctuation, which makes the prediction of POS tags of OOV punctuation much harder.

Error analysis. Having shown that using our system does have a positive effect on the POS tagging of OOV words, it is still interesting to known, what kind of errors are made by the baseline tagger in the first place and which of these can be handled by our system.

The confusion matrix in Table 6 shows the classifier's performance and different classification errors made by the baseline tagger as well as the effects of our best system compared to the baseline in parentheses. We collapse POS tags into five groups for nouns, adjectives, verbs, other standard

	pp	hunpos top1	hunpos top3	crfsuite top1	crfsuite top3
baseline	85.0 (46.6)	85.0 (46.6)	85.0 (46.6)	85.0 (50.1)	85.0 (50.1)
bow	86.4 (55.3)	86.7 (55.3)	86.3 (53.7)	87.0 (57.3)	86.9 (56.5)
wo	87.4 (60.5)	87.6 (60.5)	86.8 (56.5)	87.5 (60.0)	87.6 (60.4)
n-gram	87.7 (62.1)	**87.9** (**62.1**)	86.7 (55.9)	87.6 (61.1)	87.6 (60.4)

Table 4: Accuracy for different ways of integrating the information into the taggers. *top 3* gives the results when the three highest ranked POS tags are considered; *top 1* gives the results when only the highest ranked POS tag is used. In parenthesis is the accuracy on only the unknown words.

		baseline tagger (effect of best configuration)				
		N	A	V	other	new
gold	N	87.2 (+8.4)	7.4 (-5.8)	2.1 (-1.1)	3.3 (-1.5)	0.0 (+0.1)
	A	2.1 (+0.2)	92.6 (±0)	3.1 (-1.6)	2.1 (+1.2)	0.0 (+0.2)
	V	2.1 (-1.7)	1.1 (-0.5)	96.6 (+1.9)	0.2 (+0.2)	0.0 (+0.1)
	other	3.0 (-2.6)	2.1 (-1.5)	0.9 (-0.9)	94.0 (+4.7)	0.0 (+0.4)
	new	13.2 (-3.4)	8.3 (-6.0)	9.4 (-5.7)	69.1 (-57.4)	0.0 (+72.5)

Table 6: Confusion matrix between our baseline tagging model and the gold standard. In parentheses is the absolute difference to this baseline for our best-performing model. POS categories are collapsed into nouns, adjectives, verbs, other standard STTS tags and the new STTS 2.0 tags.

STTS tags and the new STTS 2.0 tags.

We can observe three interesting phenomena: Firstly, due to a lot of lower-cased – and thus unknown – noun forms, there is a high rate of nouns getting erroneously tagged as adjectives (7.4%). In fact, out of the 111 nouns tagged as adjectives by the baseline tagger, 94 are lower-case. This problem is mostly solved by our system (-5.8%).

Another frequent mistake is the tagging of interjections (included in *other*) as proper nouns. This is also handled quite well ($3.0\% \rightarrow 0.4\%$).

Finally, the baseline tagging model is of course not able to cope with new tags from the extended STTS tagset. The adapted model leads to an accuracy of 72.5% for these tags, which – while not quite reaching the per-class accuracy of the other classes – is a reasonable result, given the limited amount of training data.

5 Conclusions

We have shown that distributional similarity information can be used to learn possible POS tags of out-of-vocabulary words and thereby improve the performance of POS taggers on CMC data. Our best performing approach increases the overall tagging accuracy on German internet forum posts by $+2.9\%$ compared to a tagger that has been trained on standard newspaper text; for a tagger that has already been adapted to CMC data, our approach increases accuracy by $+1.7\%$ / $+2.4\%$ to 93.2%.

We use two different taggers in our experiments, a HMM-based tagger and one based on CRF. One interesting observation is that the HMM-tagger generally performs better on OOV words, while the CRF tagger gives the overall best results when we use an already adapted tagger. This observation suggests that information about OOV words is not encoded optimally in the CRF-based tagger, and that we can improve our approach in future work.

Our approach is completely unsupervised in the sense that it does not rely on any additional manually annotated data, so it can be applied to other kinds of CMC data as well.

Acknowledgements. This work is part of the BMBF-funded project "Analyse und Instrumentarien zur Beobachtung des Schreibgebrauchs im Deutschen." We would like to thank Ines Rehbein for helpful discussion and support with the implementation of the CRF-tagger.

References

Thomas Bartz, Michael Beißwenger, and Angelika Storrer. 2014. Optimierung des Stuttgart-Tübingen-Tagset für die linguistische Annotation von Korpora zur internetbasierten Kommunikation: Phänomene,

Herausforderungen, Erweiterungsvorschläge. *Journal for Language Technology and Computational Linguistics*, 28(1):157–198.

Sabine Brants, Stefanie Dipper, Peter Eisenberg, Silvia Hansen, Esther Koenig, Wolfgang Lezius, Christian Rohrer, George Smith, and Hans Uszkoreit. 2004. Tiger: Linguistic interpretation of a german corpus. *Journal of Language and Computation, Special Issue*, 2(4):597–620.

Phani Gadde, L. Venkata Subramaniam, and Tanveer A. Faruquie. 2011. Adapting a WSJ trained part-of-speech tagger to noisy text: preliminary results. In *Proceedings of the 2011 Joint Workshop on Multilingual OCR and Analytics for Noisy Unstructured Text Data*, page 5. ACM.

Péter Halácsy, András Kornai, and Csaba Oravecz. 2007. Hunpos: An open source trigram tagger. In *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*, ACL '07, pages 209–212, Stroudsburg, PA, USA. Association for Computational Linguistics.

Bo Han, Paul Cook, and Timothy Baldwin. 2012. Automatically constructing a normalisation dictionary for microblogs. In *Proceedings of the 2012 joint conference on empirical methods in natural language processing and computational natural language learning*, pages 421–432. Association for Computational Linguistics.

Andrea Horbach, Diana Steffen, Stefan Thater, and Manfred Pinkal. 2014. Improving the performance of standard part-of-speech taggers for computer-mediated communication. In Josef Ruppenhofer and Gertrud Faaß, editors, *Proceedings of the 12th Edition of the Konvens Conference, Hildesheim, Germany, October 8-10, 2014*, pages 171–177. Universitätsbibliothek Hildesheim.

Andrea Horbach, Stefan Thater, Diana Steffen, Peter M. Fischer, Andreas Witt, and Manfred Pinkal. 2015. Internet corpora: A challenge for linguistic processing. *Datenbank-Spektrum*, 15(1):41–47.

Naoaki Okazaki. 2007. CRFsuite: a fast implementation of Conditional Random Fields (CRFs).

Olutobi Owoputi, Brendan O'Connor, Chris Dyer, Kevin Gimpel, Nathan Schneider, and Noah A. Smith. 2013. Improved part-of-speech tagging for online conversational text with word clusters. In *Proceedings of NAACL-HLT*, pages 380–390.

Ines Rehbein. 2013. Fine-grained pos tagging of german tweets. In *Language Processing and Knowledge in the Web*, pages 162–175. Springer.

Anne Schiller, Simone Teufel, Christine Stöckert, and Christine Thielen. 1999. Guidelines für das Tagging deutscher Textcorpora mit STTS. Technical report, IMS-CL, University Stuttgart.