UdS-(retrain|distributional|surface): Improving POS Tagging for OOV Words in German CMC and Web Data

Jakob Prange, Andrea Horbach, Stefan Thater
Department of Computational Linguistics
Saarland University
Saarbrücken Germany
(jprange|andrea|stth)@coli.uni-saarland.de

Abstract

We present in this paper our three system submissions for the POS tagging subtask of the Empirist Shared Task: Our baseline system *UdS-retrain* extends a standard training dataset with in-domain training data; *UdS-distributional* and *UdS-surface* add two different ways of handling OOV words on top of the baseline system by using either distributional information or a combination of surface similarity and language model information. We reach the best performance using the distributional model.

1 Introduction

Part-of-speech (POS) tagging is a fundamental subtask in many linguistic tool-chains that provides necessary information for subsequent analysis steps such as lemmatization or syntactic parsing. Most recent approaches to POS tagging use statistical techniques and can provide excellent results – as long as the tagger is applied to the same kind of text it has been trained on. When applied out-of-domain, results tend to be significantly worse. This problem is particularly pronounced in the case of data from the domain of computer-mediated communication (CMC) such as posts in Internet fora or micro-posts from Twitter. POS taggers are usually trained on newspaper articles or other edited texts from professional writers, while CMC data often deviates on the lexical, orthographic (e.g., spelling errors, non-capitalization of German nouns) and grammatical level (e.g., sentences without subjects) and contains phenomena such as emoticons or action words that are not covered by standard POS tagsets (Bartz et al., 2014).

This paper describes our contribution to the EmpiriST 2015 Shared Task "Automatic Linguistic Annotation of Computer-Mediated Communication/Social Media" where we participated in the subtask of adapting POS taggers to German CMC and Web data. All three of our submitted systems are at least partially based on a previous tagging system, that we developed in the BMBF funded project "Analyse und Instrumentarien zur Beobachtung des Schreibgebrauchs im Deutschen."[1] We have shown that out-of-vocabulary (OOV) words are particularly problematic when a standard tagger is applied to out-of-domain CMC data. Therefore, our previous system focuses on OOV words in two ways: First, tagger accuracy can be improved substantially by adding relatively small amounts of manually annotated in-domain (CMC) data to a standard training set (Horbach et al., 2014). This method is used in our *retrain* system that we consider as a baseline. A further, smaller but still significant improvement can be obtained by using an additional component based on distributional models (Prange et al., 2015) that predicts possible POS tags of words which are still OOV under the retrained model.

For the shared task, we modify our system in two ways: First, the annotation guidelines underlying the training data used in our previous work differ in some details from the guidelines of the shared task. We re-annotate our previous training data to match the new annotation guidelines and use it in addition to the training data provided by the shared task. Second, we experiment with two different components for predicting POS tags of OOV words.

These experiments resulted in three individual systems: *UdS-retrain* uses different versions of additional in-domain training data to retrain a POS tagger and constitutes the basis tagger for the other two systems. *UdS-distributional* adds a component to predict the POS tag for OOV words based

[1] www.schreibgebrauch.de

Proceedings of the 10th Web as Corpus Workshop (WAC-X) and the EmpiriST Shared Task, pages 63–71,
Berlin, Germany, August 7-12, 2016.

on distributional information similar to (Prange et al., 2015); *UdS-surface* uses a combination of surface similarity and language model perplexity to normalize OOV words in a preprocessing step.

Almost all of our system configurations outperform a baseline trained on the TIGER corpus (Brants et al., 2004) alone on both datasets (with the exception of surface run 1 on Web); the improvement is especially pronounced on the CMC subcorpus. We achieve the best results on both corpora with the distributional system (87.33% on CMC and 93.55% on Web). An oracle experiment shows that the different models do not subsume each other and perform differently so that there might be room for further benefits through model combinations.

The plan for the paper continues as follows: We give a short overview of our previous work in Section 2 and describe the various data and tagsets used in our experiments in Section 3. We describe the architecture of our three systems in Section 4 and provide our results in Section 5. Section 6 provides additional analyses and experiments to better understand our results. We conclude in Section 7.

2 Our Previous Work

In previous work, we experimented with various ways to adapt statistical POS taggers to German CMC data. This section briefly summarizes the approach by Prange et al. (2015), as it was the basis for our distributional system and conceptually also inspired the surface system. It uses the *HunPos* tagger (Halácsy et al., 2007) and combines two approaches to adapt it to German CMC data.

In a first step, the tagger is (re-)trained on data which combines the standard TIGER corpus (Brants et al., 2004) with manually annotated in-domain CMC data, the *Schreibgebrauch* dataset (Horbach et al., 2015). This in-domain data was collected from forum posts of a German online cooking community (*www.chefkoch.de*), the *Dortmunder Chat-Korpus* (Beißwenger, 2013) and microposts from Twitter.[2] In total, the dataset contains approx. 34 000 tokens and has been independently annotated by three trained undergraduate students of computational linguistics using an extension of a preliminary version of the "STTS 2.0" tagset proposed by Bartz et al. (2014): Our original motivation for adapting POS taggers was to support the monitoring of German orthography; therefore, we added two additional POS tags for cases where the author incorrectly wrote two words as a single token (ERRTOK) or incorrectly separated a single word into two tokens (ERRAW).

Tagging accurracy is increased substantially (+11% on chat data) when using the annotated in-domain data as additional training data (Horbach et al., 2015). A major reason for this is that the original tagger performs relatively poorly on OOV words, and adding in-domain data to the training set decreases the amount of OOV tokens. Yet, a substantial amount of OOV tokens remains even after re-training the tagger.

Prange et al. (2015) therefore use a second component that aims at learning candidate POS tags for OOV tokens. The two key observations underlying this second component are that (i) in-vocabulary (IV) words are tagged with high accurracy and (ii) distributionally similar words tend to belong to the same lexical class and thus have the same POS label. We tagged the complete *chefkoch* dataset and trained a distributional model on the automatically annotated dataset. For each OOV word, we compute the 20 most similar in-vocabulary words, which by assumption carry reliable POS information. This candidate set is then ranked using a combination of different string similarity measures and the POS tags of the words in the candidate set are propagated to the OOV word. This results in a POS lexicon for OOV tokens, which can be directly applied to the *HunPos* tagger to guide the search process during tagging.

3 Data and Tagset

As do potentially most other participating systems we use the TIGER corpus (Brants et al., 2004) as one of the standard corpora for the task of German POS tagging as a basis, and make use of the training data provided by the shared task (*EmpiriST train*); additionally, we also use the Schreibgebrauch dataset. In contrast to previous approaches on this dataset, we use both the training and the test section for training. Table 1 shows the size and composition of all datasets.

The standard tagset for German POS tagging (here referred to as STTS 1.0) (Schiller et al., 1999) has been extended recently to account for phenomena not present in standard newswire text. The EmpiriST Shared Task datasets are annotated with a version of the STTS 2.0 tagset (Beißwenger et

[2]The dataset is available at http://www.coli.uni-saarland.de/projects/schreibgebrauch/

Dataset	Appr. size (in tokens)	Domain	Tagset
TIGER	900 000	newspaper text	STTS 1.0
EmpiriST-train CMC	5 000	Chat, Twitter, Wikipedia talk, blog comments, whatsapp	STTS 2.0
EmpiriST-train Web	5 000	monologic Internet texts	STTS 2.0
Schreibgebrauch	34 000	Forum, Chat, Twitter	STTS 2.0* & STTS 2.0

Table 1: Datasets used in our models

al., 2015) that differs slightly from the tagset used in our previous studies to annotate the *Schreibgebrauch* corpus (we call it STTS 2.0* here to distinguish it from the shared task tagset). Since we want to use both datasets to re-train the tagger, we re-annotated (in part automatically) our Schreibgebrauch corpus as follows:

- Certain particles in conceptually oral utterances that had been tagged as adverbs ADV in our data received their own tags in the EmpiriST datasets as 1) intensifier, focus and gradation particles (PTKIFG), 2) modal and downtoner particles (PTKMA) or 3) particles as part of multi-word lexemes (PTKMWL). We manually re-examined the Schreibgebrauch annotations of adverbs and adapted the tag where necessary.

- Action words like **freu** are annoted with the tag AKW in the EmpiriST data, while the Schreibgebrauch corpus uses AW. Also, the "*" which is often used to indicate an action word is taken to be part of the action word in the EmpiriST datasets, while it is a separate token in the Schreibgebrauch corpus (**/AWIND breit/ADJD grins/AW */AWIND*). We automatically changed AW to AKW and replaced AWIND by $((**/$(freu/AKW */$(*).

- The EmpiriST datasets distinguish between ASCII emoticons and emoticons represented as images, while the Schreibgebrauch corpus tags all emoticons as EMOASC even if they are represented as images. Also, the dataset uses the standard PAV instead of the tag PROAV as used in the TIGER corpus and our annotations. We used a simple regular expression to automatically identify image emoticons in the Schreibgebrauch corpus and re-annotated them as EMOIMG, and replaced PROAV by PAV.

- The Schreibgebrauch corpus uses two tags to annotate tokens which are incorrectly tokenized by the author. In cases where a word like "Umkleidekabinen" is incorrectly split into two tokens by the author ("Umkleide Kabinen"), the first token is tagged as ERRAW. In cases where two separate words are incorrectly written as a single token ("alldas"), the token is annotated as ERRTOK. Instances of ERRTOK are automatically re-tagged as XY and all tokens tagged as ERRAW were removed following the observation that these tokens are mainly premodifiers.

Since tokens which need to be re-annotated as a disourse marker DM cannot be identified systematically using simple regular expressions, we checked and re-annotated only occurrences of ADV and KOUS. We did not (re-)annotate EMLs; we conjecture that they do not occur in our data.

4 Our Systems

We entered three different systems into the competition that tackle the tagging problem in different ways: a simple retraining approach (UdS-retrain), which enriches a standard training set with additional in-domain training data and is used as a baseline; and two systems that additionally target specifically OOV words: a distributional approach that exploits the observation that similar words tend to have the same POS tag (UdS-distributional) and an approach based on surface similarity that aims at detecting and correcting potential spelling mistakes (UdS-surface). We also compare our models against another baseline that is trained on the TIGER corpus only. In all of our systems, we use the *HunPos* tagger (Halácsy et al., 2007).

4.1 UdS-retrain

Following previous work (Horbach et al., 2014; Kübler and Baucom, 2011), we adapt the tagger by retraining it on a dataset that combines the standard

corpus	run1	run2	run3	run4
TIGER	✓	✓	✓	✓
EMPIRIST - same domain	✓	✓	✓	✓
EMPIRIST - other domain			✓	✓
Schreibgebrauch - original	✓			
Schreibgebrauch - adapted		✓	✓	

Table 2: Training corpora for each of our system runs

TIGER corpus with additional in-domain data: the *Schreibgebrauch* corpus and the shared task training sets.

Since the annotated in-domain training data is very small compared to the size of the TIGER corpus, we boost the in-domain data by adding it 5 times to give it more weight. Furthermore, we duplicated the TIGER corpus and used both the original version as well as a version obtained by automatically converting it to the new German orthography, to account for the fact that writers in German CMC data might be using both the old and the new German orthography.

We submitted runs for three different configurations of the UdS-retrain system, depending on the corpora used to train the model:

- **run 1** uses a model trained on TIGER, the EmpiriST training data for the specific subcorpus (CMC and Web) and the original *Schreibgebrauch* training data without any tagset adaptations.

- **run 2** is like run 1, but uses a version of the *Schreibgebrauch* training data adapted to the STTS 2.0 version used in the shared task datasets.

- **run 3** is like run 2, but uses both the CMC and web training data sets, independent of the text type the model is applied to.

4.2 UdS-distributional

This system closely follows Prange et al. (2015). As described above in Section 2, the system induces a POS lexicon that lists suitable POS tags for OOV words, i.e., words that do not occur in the training data. This POS lexicon is used by the *HunPos* tagger to limit the search space when the tagger sees an OOV word.

We use the UdS-retrain model (run-2) to tag about half a billion tokens from the German online cooking platform *www.chefkoch.de* and train a distributional model that uses POS 5-grams as features, weighted using pointwise mutual information (PMI). This distributional model is used to find, for each OOV word in the test set, the 20 distributionally most similar IV words. From this candidate set, we extract one or more POS tags and store them in the POS lexicon as possible tags of the OOV word.

We submitted three system runs, that differ in how the POS tags to be added to the POS lexicon are selected:

- **run 1:** The distributional model returns a list of the distributionally most similar words together with their POS tag. The tags are then ranked using different ranking algorithms based on surface similarity between the original words and its distributional neighbours (Levenshtein and 2 variants of Jaro-Winkler distance) and the position and frequency of each POS tag in the list (ranking by frequency, ratio between frequency and first position in the list, sum of inverse ranks at which a tag occurs). Each ranker contributes one top-ranked POS tag, among which we take a majority vote.

- **run 2:** This setting is a variant of the one above, where we use up to three POS tags from the list of top-ranked tags proposed by the different rankers: If the list contains at least three tags and the most frequent tag occurs less than 4 times in the candidate list, we also include the second most frequent tag in the POS lexikon. If the list contains 4 or more entries, we also include the third best entry. In doing so, we treat the frequency of each tag in the list as a confidence threshold and include more candidates if our confidence in the best one is low.

- **run 3:** the best-performing configuration from (Prange et al., 2015), where we linearly combine the two best-performing rankers from run-1: Levenhstein distance and the frequency-position-ratio.

4.3 UdS-surface

This approach explores an alternative to the distributional model; like the former, it explicitely

addresses OOV words. In contrast to the former, however, we rely here on the assumption that many OOV words are spelling errors (or voluntary misspellings) that are on the surface very similar to the word they stand for, similar to approaches by Han et al. (2012) and Gadde et al. (2011). In this approach, we first filter the OOV words that are likely to be typos and then rank their potential replacements using language models. We thus construct a normalized version of the sentence and feed it to the tagger.

In order to make sure that we select primarily such candidates for normalization that are indeed misspellings and not just words unknown to the tagger, we use the spellchecker *aspell* in its standard configuration to identify words that are likely misspellings (in contrast to known words or words for which aspell has no suggestions for corrections). For these words we collected lists of potential replacements candidates in three different ways (described below). We then use a language model using the SRILM toolkit (Stolcke, 2002) built on raw texts from *www.chefkoch.de*, rank the different versions for each sentence and select the one with the lowest perplexity.

We tested the following three configurations of the system.

- **run 1:** We use a variant of Jaro-Winkler similarity[3] and consider only replacement candidates from the annotated training data with a surface similarity above a certain threshold. In the first run we set the threshold to 0.8.
- **run 2:** In the second run, we use a more restrictive threshold and only select tokens with a similarity above 0.95.
- **run 3:** In this setting we only select the candidates with the highest similarity (several if they have the same similarity score).

5 Shared Task Results

This section presents the results for our submitted runs.

5.1 Shared task runs

Table 3 shows that all of our systems' configurations clearly outperform the baseline for both CMC and Web corpora ($\alpha < 0.001$ according to a McNemar test), except for surface run 1; the distributional model works best for both subcorpora. This is plausible, given that the model builds on UdS-retrain as its baseline and has – compared to UdS-surface – a more unbiased approach towards OOV words; it does not expect them to be necessarily typos. The model can find replacements whenever an OOV word is frequent enough in the large background corpus for the model. Within the three variants of our distributional models, we see very little variance in performance.

Run	CMC	Web
TIGER baseline	71.15	91.19
UdS-retrain 1	85.48	92.71
UdS-retrain 2	86.40	92.79
UdS-retrain 3	86.43	92.71
UdS-distributional 1	87.26	93.51
UdS-distributional 2	87.33	93.55
UdS-distributional 3	87.29	93.01
UdS-surface 1	84.58	91.19
UdS-surface 2	86.45	92.43
UdS-surface 3	85.36	92.01

Table 3: Evaluation results of our system runs

For the retraining approach, we can see that the adaptation of our project corpus to the new tagset gives a performance boost of about 1 percent for the CMC dataset (statistically significant, $\alpha < 0.001$), but not for the Web corpora. This is not surprising as the CMC dataset contains much more phenomena covered by new tags, some of which have systematically different tags in our original version of our own training data: 479 CMC test tokens (out of 5234) received a gold tag from STTS 2.0 (285 tokens from the subset that would have been tagged differently in our STTS 2.0* version compared to our adapted version), compared to 94 tokens (out of 7568) from the Web dataset (87 tokens that differ between tagset versions).

The UdS-surface system outperforms the retrain approach only slightly for the CMC dataset (statistically not significant), and not for the Web Corpora. We suspect a higher frequency of typos in the CMC dataset. The Web corpora dataset seems much more well-formed, so that we might have there a higher percentage of OOV words that are erroneously replaced, although the word is not a

[3]Standard Jaro-Winkler uses the length of common prefixes to compute a similarity score; we also consider a variant that uses common suffixes instead, with the idea that a shared suffix might indicate the same POS tag

typo but just a lexical gap, i.e. does not occur in the tagger lexicon.

5.2 Performance on OOV Words

All of our systems focus on improving the performance of words that are OOV for a standard tagger. We therefore evaluate the performance on OOV and IV words separately. Table 4 shows the performance on these words, if we take the TIGER baseline as a reference as to whether a word is known or not. Consequently, all retrain, distributional and surface runs have the same OOV words as TIGER and thus the numbers for the performance on OOV are directly comparable.

We can see that we reach the vast majority of our improvements over the TIGER baseline on OOV words; the performance on IV words also improves by 5 to 6%, due to both a better context that helps to disambiguate words with several possible POS tags (e.g. ART vs. PREL) and additional lexicon entries for words that were already known in TIGER but with different or fewer POS tags. For instance, a word like *essen* (verb – to eat) might also occur in in-domain training data as the erroneously not-capitalized version of the noun *Essen* (meal).

Adding a component for handling OOV words reduces the number of words for which our model has no additional information about the POS tag. For the distributional models, there are only about 4% of tokens for which we do not have any predictions about distributional neighbours. For the surface models, between 3 and 10 percent of all tokens are not replaced by a similar word and thus are treated as OOV by the tagger.

6 Discussion and Analysis

This section presents additional experiments and analyses that aim at shedding light on the differences between the individual systems.

6.1 Experiment 1: How different are our systems?

One interesting question is how different our individual systems really are: Do they subsume each other, or are there opportunities for improvements by combining them? To address this question, we evaluate as an oracle condition how good a combined tagger would be. To this end, we evaluate a condition where we take everything as correct that is correctly done by at least one configuration of one of our systems. This evaluation is thus an upper bound of what an optimal combination of all our apporaches might be able to reach. We do that within individual systems and across all three systems (see Table 5). We also evaluate for how many tokens all systems get it right (*all correct* in the table). We can see that we only profit slightly from combining different variations for a single system, and – as expected – more substantially from combining the three models corresponding to three different approaches.

The *all correct* evaluation shows that even the system with the worst performance (surface-1) is better than only those cases that all systems have correct, i.e. even this system contributes something and is not subsumed by the others.

In order to understand the remaining problems better, we looked at the remaining hard cases, i.e., tokens that none of our system configurations were able to tag correctly. Tables 6 and 7 show the most frequent mistaggings and the confusions for those POS tags that occur at least 10 times in a dataset.

We can see that we especially struggle with the new adverb derivates; we assume that to be because of their low frequencies, and because the lexical items appear often with the ADV tag in TIGER. Other hard cases are more typical POS confusion phenomena such as NN vs. NE, ADJD vs. ADV, VVINF vs. VVFIN etc.

6.2 Experiment 2: The influence of our manually annotated data

All of our submitted systems use the Schreibgebrauch data in some way. We have observed in previous work that adding this data improved performance, compared to a model trained on newspaper data, by a large amount Therefore, we want to check, in the next experiment, what our results would be if we had used only the in-domain training data provided by the shared task for each subcorpus.

We see in table 8 that the CMC subcorpus profited substantially from the additional *Schreibgebrauch* corpus (up to +2.96%); for Web, however, the performance did not change. We attribute that to the domain differences between Web and CMC.

7 Conclusion

In this paper we described our contributions to the EmpiriST 2015 Shared Task on automatic linguistic annotation of computer-mediated communication/social media. We entered three systems into

Run	CMC IV	CMC OOV	CMC %OOV	Web IV	Web OOV	Web %OOV
TIGER baseline	83.39	28.95	22.83	94.44	71.07	14.20
UdS-retrain 1	88.88	73.97	22.83	95.16	77.86	14.20
UdS-retrain 2	89.82	74.81	22.83	95.23	78.05	14.20
UdS-retrain 3	89.90	74.73	22.83	95.10	78.23	14.20
UdS-distributional 1	89.73	78.91	22.83	95.27	82.88	14.20
UdS-distributional 2	89.75	79.16	22.83	95.29	83.07	14.20
UdS-distributional 3	89.80	78.83	22.83	95.26	79.44	14.20
UdS-surface 1	88.66	70.79	22.83	94.93	68.56	14.20
UdS-surface 2	89.40	76.49	22.83	95.12	76.19	14.20
UdS-surface 3	88.66	74.23	22.83	94.99	73.95	14.20

Table 4: Evaluation results split into OOV and IV words according to the TIGER baseline.

	CMC	Web
oracle - retrain	87.03 (86.43)	93.05 (92.79)
oracle - distributional	87.62 (87.33)	93.70 (93.55)
oracle - surface	87.52 (86.45)	93.59 (92.43)
oracle - all	89.78 (87.33)	94.94 (93.55)
all correct - all	81.14 (84.58)	89.14 (91.19)

Table 5: Results for an oracle condition experiment. In parentheses is the performance of the best run that contributed to the oracle experiment and the worst run for the *all correct* condition.

tag	freq	out of	3 most frequent confusions
PTKIFG	59	72	ADV (413), ADJD (71), PIS (21),
$(	43	343	$. (306), XY (45), KON (36),
PTKMA	42	74	ADV (325), ADJD (23), PTKIFG (20),
NE	33	230	NN (121), ADR (89), FM (19),
$.	32	358	$((282), NN (3), ITJ (2),
NN	32	696	NE (90), ADJA (69), ADJD (30),
ADJD	30	187	ADV (152), VVPP (63), ADJA (27),
ITJ	17	99	ONO (45), AKW (31), NN (25),
URL	16	16	NE (37), CARD (27), XY (18),
AKW	15	60	VVFIN (45), NN (28), NE (12),
VVFIN	14	183	VVINF (73), NN (18), ADJD (13),
PTKVZ	12	40	APPR (54), ADV (27), ADJD (18),
ADR	12	48	NE (36), NN (32), ADV (18),
ADV	12	268	ADJD (36), PTKVZ (18), PIAT (14),
ADJA	11	149	NE (38), NN (25), FM (15),
VVIMP	11	20	VVFIN (27), NE (24), ADV (18),
KOKOM	11	21	APPR (45), KOUS (27), FM (16),
PDS	10	51	ART (45), PRELS (19), PDAT (9),

Table 6: Most frequent mistagged gold standard tags for CMC. We show the frequency of the mistagged word compared to the overall occurrence of that word. Misstagging numbers are higher, as they refer to the sum of misstaggings by all nine tagging models.

tag	freq	out of	3 most frequent confusions
PTKIFG	53	61	ADV (424), ADJD (53),
VVFIN	36	250	VVINF (172), VVPP (116), NN (11),
NN	27	1661	NE (121), ADJA (57), FM (18),
NE	26	252	NN (190), ADJD (9), URL (8),
$(	23	263	NN (68), XY (45), $. (30),
FM	18	43	NE (76), NN (20), VAFIN (18),
ADJD	17	223	ADV (101), ADJA (17), NE (14),
ADJA	14	498	FM (27), NN (25), ADJD (18),
APPR	13	583	ADV (36), KOKOM (36), KON (36),
VVINF	13	125	VVFIN (81), NN (36),
PTKMWL	13	14	ADV (108), ADJD (9),
VVIMP	10	12	VVFIN (59), VVPP (13), ADJD (9),
VAFIN	10	208	VAINF (90),

Table 7: Most frequent mistagged gold standard tags for Web

	CMC	Web
retrain run 1/2	83.44	92.71
retrain - run 3	83.65	92.84
distrib - run 1	84.89	93.38
distrib - run 2	85.00	93.39
distrib - run 3	84.94	92.88
surface - run 1	82.25	91.05
surface - run 2	84.05	92.36
surface - run 3	82.98	91.66

Table 8: Results for versions of our systems that have been trained without our additionally annotated training data.

the competition: *UdS-retrain* uses manually annotated in-domain CMC data in addition to a standard newspaper corpus (TIGER) to train the tagger, *UdS-distributional* additionally learns possible POS tags of OOV words not covered by the training set and *UdS-surface* normalizes OOV words prior to tagging using surface similarity measures and a language model.

Our results confirm findings made in previous work: A big improvement over a standard tagger trained on newspaper texts is obtained by UdS-retrain (+15% on CMC); a further improvement is obtained by UdS-distributional (+1.8% on CMC), while UdS-surface does not lead to significantly better results (+0.05% on CMC; -0.4% on Web).

The distributional system is closely based on previous work by Prange et al. (2015). This previous system learns only one possible POS tag for OOV words. Here, our attempt was to learn several possible POS tags and let the tagger decide which of these candidate tags is most appropriate in a certain context (run 2). However, the differences from runs 1 and 3 are very small and statistically not significant.

While UdS-surface improves tagging accuracy of OOV words (compared to UdS-retrain on CMC), the accuracy on IV words decreases, which suggests that this approach is not accurate enough to improve tagging results. More specifically, we often erroneously correct words that are OOV but not spelling errors.

From our oracle experiments, we see that the combination of our taggers has the potential to be better than each tagger individually. None of our systems explores "low hanging fruits" such as using regular expressions to identify addressing terms, email addresses or emoticons, which might also be integrated in future work.

Acknowledgments

This work is part of the BMBF-funded project "Analyse und Instrumentarien zur Beobachtung des Schreibgebrauchs im Deutschen". We thank our student assistants Maximilian Wolf and Sophie Henning for the annotations used in this study as well as the anonymous reviewers for their helpful comments on this paper.

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.

Michael Beißwenger, Thomas Bartz, Angelika Storrer, and Swantje Westpfahl. 2015. 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 / Social Media" (EmpiriST2015). Technical report.

Michael Beißwenger. 2013. Das Dortmunder Chat-Korpus. *Zeitschrift für germanistische Linguistik*, 41(1):161–164.

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.

Sandra Kübler and Eric Baucom. 2011. Fast domain adaptation for part of speech tagging for dialogues. In Galia Angelova, Kalina Bontcheva, Ruslan Mitkov, and Nicolas Nicolov, editors, *RANLP*, pages 41–48. RANLP 2011 Organising Committee.

Jakob Prange, Stefan Thater, and Andrea Horbach. 2015. Unsupervised Induction of Part-of-Speech Information for OOV Words in German Internet Forum Posts. In *Proceedings of the 2nd Workshop on Natural Language Processing for Computer-Mediated Communication / Social Media (NLP4CMC 2015)*.

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

Andreas Stolcke. 2002. SRILM - An Extensible Language Modeling Toolkit. pages 901–904.