Efficient Training Data Enrichment and Unknown Token Handling for POS Tagging of Non-standardized Texts

Melanie Neunerdt, Michael Reyer, Rudolf Mathar
Institute for Theoretical Information Technology
RWTH Aachen University, Germany
{neunerdt, reyer, mathar}@ti.rwth-aachen.de

Abstract

In this work we consider the problem of social media text Part-of-Speech tagging as fundamental task for Natural Language Processing. We present improvements to a social media Markov model tagger, by adapting parameter estimation methods for unknown tokens. In addition, we propose to enrich the social media text corpus by a linear combination with a newspaper training corpus. Applying our tagger to a social media text corpus results in accuracies of around 94.8%, which comes close to accuracies for standardized texts. [1]

1 Introduction

Part-of-Speech (POS) tag information can be achieved by automatic taggers with accuracies up to 98% for standardized texts. However, when applying state-of-the-art taggers to non-standardized texts such as social media texts or spoken language, tagging accuracies drop significantly. Social media texts suffer from informal writing style such as misspelled or shortened words, which leads to a high number of unknown (out-of-vocabulary) tokens. Thus, some special challenges are given for developing methods for automatic social media text POS taggers. In this work we propose some adapted parameter estimation methods to our social media Markov model tagger, *WebTagger* (Neunerdt et al., 2013a). We improve the parameter estimation for unknown tokens in several ways. Beside different combination methods for tokens' prefix and suffix tag distributions, we propose a semi-supervised verb auxiliary lexicon. Furthermore, we consider the different grammatical structure of social media and newspaper texts leading to diverse distributions of POS tag sequences. In contrast to existing POS tagging approaches, we propose a linear combination of a social media training corpus and a newspaper corpus by an efficient oversampling of the in-domain training data. We experimentally evaluate the proposed methods for a German social media text corpus and different social media text types. Results are compared to the underlying *WebTagger* and state-of-the art widely used POS taggers. We show that by applying our adapted Markov model tagger to an existing social media text corpus we are able to obtain accuracies close to 95%.

The paper is organized as follows: Section 2 summarizes the related work to provide an overview of POS tagging, particularly for non-standardized texts. In Section 3 and 4 we introduce the basic tagger model and propose our adapted parameter estimation methods. Section 5 reports experimental results. In Section 6 we conclude our work.

2 Related Work

Performance investigations of state-of-the art taggers (Toutanova et al., 2003; Schmid, 1995) show that automatic POS tagging of non-standardized social media texts results in significant accuracy drops, see (Giesbrecht and Evert, 2009; Ne-

unerdt et al., 2013b) . Therefore, recent publications (Gadde et al., 2011; Owoputi et al., 2012; Owoputi et al., 2013; Rehbein, 2013; Neunerdt et al., 2013a) particularly deal with the task of tagging non-standardized texts, such as twitter messages or Web comments. (Gadde et al., 2011) introduce feature adaptions to the Stanford maximum entropy tagger (Toutanova et al., 2003), to handle noisy English text. Results are evaluated based on an SMS dataset. In (Gimpel et al., 2011) a twitter tagger based on a conditional random field (CRF) with features adapted to twitter characteristics is proposed. They propose some additional word clustering and further improvement to their method in (Owoputi et al., 2013) and evaluate their approach on different English twitter data, where a maximal accuracy of 92.8% is achieved. (Rehbein, 2013; Neunerdt et al., 2013a) propose POS taggers for German social media texts. In (Rehbein, 2013) a CRF POS tagger for German Twitter microtexts is presented. Applying word clustering with features extracted from an automatically created dictionary leads, to 89% accuracy, which is slightly lower then results achieved for English twitter data. In (Neunerdt et al., 2013a) a Markov model tagger, called *WebTagger*, for the application to Web comments is proposed. Improvements are particularly achieved by the mapping of unknown tokens to known training tokens or some regular expressions. Furthermore, a semi-supervised auxiliary lexicon is proposed. Tagging accuracies of about 94% are achieved on a Web comment corpus. The proposed *WebTagger* serves as a basis for the methods introduced in this work.

3 Tagger Model

As a basic tagger model we use the Markov model proposed in (Neunerdt et al., 2013a). In this section we shortly explain this basic model. The aim of the tagger is to predict the associated POS tag sequence $t_1, \ldots, t_n, \ldots, t_N$ with $t_n \in \mathcal{T}$ (STTS) for a given sequence of tokens $w_1, \ldots, w_n, \ldots, w_N$ with $w_n \in \mathcal{W}$, where $\mathcal{W}$ contains all possible tokens. The optimization task is given as

$$\hat{t}_1^N = \arg\max_{t_1^N} P(t_1^N, w_1^N)$$

with a sequence of POS tags t_l^n

$$t_l^n = \begin{cases} (t_l, \ldots, t_n) & 1 \leq l \leq n \leq N \\ (t_1, \ldots, t_n) & l \leq 0 \end{cases}$$

where $l \in \mathbb{Z}$, $n \in \mathbb{N}$, and $l \leq n \leq N$. The sequence of tokens w_l^n is defined analogously. By applying the probability chain rule and some simplifying assumptions the optimization problem is solved by:

$$\hat{t}_1^N = \arg\max_{t_1^N} \prod_{n=1}^{N} \overbrace{\frac{P(t_n \mid w_n)}{P(t_n)}}^{LexicalProb.} \overbrace{P(t_n \mid t_{n-k}^{n-1})}^{TransitionProb.}$$

where $k \in \mathbb{N}$ describes the dependency depth of transition probabilities. Before the tagger can be used to predict the associated POS tag sequence $\hat{t}_1^N$, lexical und transition probabilities have to be estimated. Estimation of transition probabilities are inherited from (Neunerdt et al., 2013a). Lexical probability estimation methods are adapted and complemented, by our proposed methods described in the following section.

4 Lexical Probability Estimation

Lexical probability estimation differs significantly depending on wether a token is known or unknown from the training corpus. Whereas for known tokens the empirical distributions is accessible from the training, in the unknown case it is a more challenging task. However, we still know some characteristics of the word, e.g. the prefix and suffix of a word or some knowledge from an unsupervised or semi-supervised corpus.

In the following section we propose adaptions to a social media text tagger based on such characteristics and knowledge. In order to describe our estimation methods we first introduce a manually annotated social media text corpus

$$\mathcal{TR}_{ID} = \left\{ (\hat{w}_i, \hat{t}_i) \mid 1 \leq i \leq I \right\} \tag{1}$$

which is used for training. For each word $\hat{w}_n$ the correct tag $\hat{t}_n$ is known. Furthermore, we treat lexical probabilities as position independent and hence replace $P(t_n \mid w_n) = P(t \mid w)$ in the following notation.

4.1 Prefix/Suffix Combination

Previous work has shown that a words' prefix and suffix can successfully be used to determine the words' POS tag. Based on the set of training tokens $\mathcal{W}$ we determine all prefixes $p \in P$ and suffixes $s \in S$ of maximal length five. We assess the

lexical probabilities for a given word w with its prefix $p(w)$ by:

$$\hat{P}_p(t \mid w) = \frac{|\{i|\hat{t}_i=t \wedge p(\hat{w}_i)=p(w)\}|}{|\{i|p(\hat{w}_i)=p(w)\}|}$$

Lexical probabilities $\hat{P}_s(t \mid w)$ are defined equivalently. The open question is, how to combine prefix and suffix tag distributions. In our approach we propose four different combination methods and discuss and compare them in Section 5. First, we assume prefix and suffix tag distributions to be independent and hence use the joint probability distribution

$$\hat{P}^g_{ps}(t \mid w) = \frac{\hat{P}_p(t|w)\hat{P}_s(t|w)}{\sum_t \hat{P}_p(t|w)\hat{P}_s(t|w)}$$

later referred as *geometric mean*. Combining prefix and suffix distributions in that way has been successfully be applied to POS tagging performed on newspaper texts in (Schmid, 1995). However, the characteristics of unknown tokens in social media texts differ from those appearing in newspaper texts. A more robust method for uncommon prefix and/or suffix, which arise from informal writing style characteristics, e.g. word shortenings or typing errors is needed. Therefore, in a second step we combine prefix and suffix tag distributions by building the *arithmetic mean* value for each tag probability, as proposed in our previous work, (Neunerdt et al., 2013a):

$$\hat{P}^a_{ps}(t \mid w) = \frac{\hat{P}_p(t|w)+\hat{P}_s(t|w)}{\sum_t \left(\hat{P}_p(t|w)+\hat{P}_s(t|w)\right)} \qquad (2)$$

In a third step, we define an approach aiming at choosing the most reliable tag distribution between $\hat{P}_p(t \mid w), \hat{P}_s(t \mid w)$. Therefore the entropy of prefix and suffix tag distributions is used as a criteria. We introduce random variables $T_{p(w)} \sim \left(\hat{P}_p\left(t \mid w\right)\right)_{t\in\mathcal{T}}$ and $T_{s(w)}$ analogously. The idea is to minimize the conditional entropy and hence chose the tag distributions, which contains less uncertainty about the tag t to predict:

$$\hat{X} = \arg\min_{X\in\left\{T_{p(w)},T_{s(w)}\right\}} H(X) \qquad (3)$$

with

$$H(T_{p(w)}) = -\sum_{t\in\mathcal{T}} \hat{P}_p(t \mid w) \log \hat{P}_p(t \mid w)$$

and $H(T_{s(w)})$ analogously. However, the significance of the empirical prefix/suffix POS tag distribution, strongly depends on the frequency of prefixes/suffixes. A prefix, which has been seen once, leeds to zero uncertainty about the tag and will fulfill the minimum criteria. Hence, we apply some simple tests on the frequencies before applying the minimum entropy approach (3). The first test checks, if the frequencies of both prefix and suffix exceed a predefined threshold α, i.e.,

$$\hat{P}_{p(w)} > \alpha \wedge \hat{P}_{s(w)} > \alpha \qquad (4)$$

In that case the distribution given by $\hat{X}$ in (3) is used. As optional tests we check if exactly one of the thresholds is exceeded and use the corresponding probability distribution. If all these tests fail the distribution from (2) is taken. We will evaluate this strategy later on, with and without the optional tests, referred as *Rule-based-2-case* and *Rule-based-4-case*.

4.2 Semi-supervised Verb Auxiliary Lexicon

Investigating tagging results of state-of-the art newspaper taggers applied to social media texts, exhibit a frequent number of unknown verbs. This can be explained by the different dialogic style of social media texts, where different verb conjugations occur. Even a tagger trained on social media data, only contains a small part of such verbs, due to the small corpus size. Furthermore, lexical probabilities can not reliably be estimated from prefix and suffix tag distributions for such verbs. However, preparing a fully-supervised social media training text with adequate corpus size is extremely time-consuming and demands expert knowledge from the annotator. We propose an alternativ approach, which reduces annotation effort significantly.

The basic idea is to create a verb auxiliary lexicon with corresponding tag sets for each token. For approximately 14,000 verbs, a conjugation table including indicative and subjunctive for different tenses as well as the imperative, participle and infinitive is extracted from *www.verbformen.de*. For an exemplary conjugation table, the corresponding POS tag is assigned manually to each verb form. Corresponding POS tags are automatically transferred to all other conjugation tables. Based on that conjugation tables all possible tokens with their corresponding tags denoted by $\mathcal{T}_{w_m}$ are combined in a verb auxiliary lexicon $\mathcal{V}^+$ containing 115,000 entries. If there is more than one possible tag, an adequate tag distribution needs to be assigned. Therefore, two approaches

are utilized. First, all words $\hat{w}_k$ of the manually annotated training corpus with the same POS tag set $\mathcal{T}_{w_m}$ are determined and the cumulated tag distribution of those words is taken. Hence, the lexical probability is refined as

$$\hat{P}_{\mathcal{V}^+}(t \mid w_m) = \frac{|\{k|\hat{t}_k{=}t \wedge \mathcal{T}_{\hat{w}_k}{=}\mathcal{T}_{w_m})\}|}{|\{k|\mathcal{T}_{\hat{w}_k}{=}\mathcal{T}_{w_m}\}|},$$

where $\mathcal{T}_{\hat{w}_k} = \{\hat{t}_l \mid \hat{w}_l = \hat{w}_k\}$. We assume all $t \in \mathcal{T}_{w_m}$ to be equally distributed, if no word with the same POS tag set $\mathcal{T}_{w_m}$ exists. If a token is not known from training or the verb auxiliary lexicon, prefix-/suffix estimations described in the previous section is performed.

4.3 Joint-Domain Training

In this section, the term *domain* is associated with a text corpus characterized by a particular style characteristic. A social media text corpus is mentioned as in-domain corpus, whereas all text with different characterization are out-domain texts. We define the combination of in- and out-domain training data as joint-domain training. Different experimental studies have shown that out-domain training data can improve tagging accuracies, e.g., (Rehbein, 2013; Neunerdt et al., 2013a). This particularly holds, if the available in-domain corpus of small size only. A typical approach is to stepwise increase the amount of out-domain training and retrain the tagger on such data. Then the amount of out-domain training data achieving best results is determined.

In contrast to existing approaches, we suggest an alternative method for combining in- and out-domain training data. The basic idea is a weighted joint-domain training. A manually annotated newspaper training corpus

$$\mathcal{TR}_{OD} = \{(\dot{w}_n, \dot{t}_n) \mid 1 \leq n \leq O\}$$

is added to our *WebTrain* corpus (1). In contrast to other approaches information from the whole available out-domain training corpus is used, no matter about corpus size. To cope with the different corpora sizes , we apply oversampling to the in-domain social media text corpus. Therefore, we multiply the *WebTrain* corpus $\beta \in \mathbb{N}$ times, while combining it with the newspaper corpus. We use a set of combined training pairs

$$\text{with } \mathcal{TR} = \{(\tilde{w}_n, \tilde{t}_n) \mid 1 \leq n \leq \tilde{N} = O + \beta I\}$$

$$(\tilde{w}_n, \tilde{t}_n) = \begin{cases} (\dot{w}_n, \dot{t}_n) & 1 \leq n \leq O \\ (\hat{w}_i, \hat{t}_i) & n > O\,, i = (n - O - 1 \mod I) + 1. \end{cases}$$

Table 1: Tagger evaluation for different estimation methods based on prefix and suffix information.

	Mean Precision		**Mean Recall**		**Mean Accuracy**	
	Pref/Suf	Total	Pref/Suf	Total	Pref/Suf	Total
***WebTrain* Test**						
Geometric	**61.43**	**84.96**	43.16	85.66	71.37	94.66
Arithmetic	53.06	84.43	51.03	85.82	**73.97**	**94.79**
Rule-base 2-case	51.58	84.70	50.86	85.76	73.50	94.77
Rule-base 4-case	41.88	84.65	**51.58**	**86.00**	71.90	94.68
***WebTypes* Test**						
Geometric	**37.96**	**80.67**	26.64	80.13	57.08	90.42
Arithmetic	35.02	78.67	**35.24**	80.47	58.02	90.63
Rule-base 2-case	35.84	78.73	34.31	**80.68**	**58.09**	**90.66**
Rule-base 4-case	29.96	78.29	34.07	80.42	56.28	90.48

The method of oversampling, see ,e.g., (Pelayo and Dick, 2007), has originally been proposed to handle the class imbalance problem in a sample corpus. Combining imbalanced in- and out-domain training data corpora has not yet been performed to the problem of POS tagging.

5 Experimental Results

We first evaluate the treatment of unknown words with different prefix/suffix estimation methods and with the semi-supervised verb auxiliary lexicon. After comparing the proposed WebTagger with two state-of-the art taggers, the performance increase by weighted joint-domain training is pointed out in more detail in 5.2.

For the purpose of training two corpora, an in-domain social media corpus and out-domain newspaper text corpus are used. As social media texts, we use the *WebTrain* corpus with Web comments containing 36,000 tokens, introduced by (Neunerdt et al., 2013b). A detailed description and further corpus statistics can be found in (Neunerdt et al., 2013b; Neunerdt et al., 2013a). Annotation rules, particularly for social media text characteristics, and inter-annotator agreement results are given in (Trevisan et al., 2012). As a newspaper corpus we use the *TIGER* treebank (Brants et al., 2004) text corpus, containing 890,000 tokens. In order to test the tagger with different parameter settings on different social media text types, we use the *WebTypes* corpus (Neunerdt et al., 2013a) as additional test data. All corpora are annotated with manually validated POS tags according to the STTS annotation guideline.

5.1 Unknown Word Treatment Analysis

For all evaluations in this section, we perform ten 10-fold cross validations on the *WebTrain* corpus.

WebTrain subsets are created by randomly selecting sentences. The following results are mean values over the resulting 100 training and test sample pairs. Note that for all cross validations the taggers are trained in a combination with 90% of the *TIGER* corpus. The remaining *TIGER* subset is used for testing.

First, we discuss different prefix/suffix combinations methods. All cross validation results for the different methods are depicted in the upper part of Table 1. On the average each *WebTrain* test set contains about 4.22% tokens, where prefix/suffix estimation is applied. We calculate mean class precision and recall rates and the total accuracies for the whole test text (Total) and for the tokens, where the prefix/suffix estimation is applied (Pref/Suf). Experimentally we determine $\alpha = 50$ to be the best threshold for the *Rule-based-2-case (R-b2c)* and *Rule-based-4-case (R-b4c)* method and depict results for that value. In order to investigate the influence on different social media text types, we additionally apply all taggers to the *WebTypes* corpus, where prefix/suffix estimation is applied to 8.44% tokens on average. Results are depicted in the lower part of Table 1. The arithmetic mean method results in the best overall *WebTrain* accuracies. However, considering the mean class precision, the geometric mean method significantly outperforms the other methods with 61.43% accuracy achieved on prefix/suffix tokens. The R-b4c approach reaches slightly better mean class recall results compared to the arithmetic mean. Hence, depending on the later application, requiring POS tag information, one might be rather interested in a high per class accuracy in contrast to the total accuracy and rather prefer one of the later mentioned methods. Results achieved on the *WebTypes* data basically confirm these cross validation results. However, the R-b2c method slightly increases mean accuracies and total recall rates.

In the following, we evaluate the performance of the semi-supervised verb auxiliary lexicon and decide to use prefix/suffix combination by the *arithmetic mean* method for all following evaluations. Cross validation accuracies achieved for *WebTrain*, *WebTypes* and *TIGER* are depicted in Table 2 without (*) and with (-) verb lexicon. In addition to total accuracies, unknown word accuracies are depicted. The introduction of the verb lexicon increases the unknown word accuracy about 1 percentage point, whereas the verb lexicon achieves about 80% accuracy, which is significantly higher compared to prefix/suffix methods. Noteable is that the performance of the verb lexicon drops about 20 percentage points, when applied to *WebTypes*. This can be explained by a high number of verbs, where no known word with the same POS tagset T_{w_m} exists and hence estimates are less reliable, due to the equal tag distribution. Furthermore, it has to be considered that accuracies are averaged over 100 different trainings but the *WebTypes* test set is fixed and hence not exactly comparable.

Finally, we compare the adapted WebTagger to two state-of-the art taggers, TreeTagger (Schmid, 1995) and Stanford (Toutanova et al., 2003), see Table 2. Both taggers are trained and tested on the same 100 samples using their standard parameters. Influence of linear combined joint-domain training leads to 0.36 and 0.51 percentage points improvement for *WebTrain* and *WebTypes* (forth column). Joint-domain training methods are studied in more detail in Section 5.2. The adapted WebTagger significantly outperforms both state-of-the art taggers, when applied to social media texts. Differences between the taggers are statistically significant according to a corrected resampled paired t-test (Nadeau and Bengio, 2001) applied to all cross validation with a significance level of $p = 0.001$. All results achieved with the adapted WebTagger on the newspaper test drop slightly. This is due to the β factorization towards the *WebTrain* corpus. However, the tagger is developed for social media texts.

5.2 Influence of Joint-domain Training

In this section we investigate the influence of out-domain training data in more detail. We particularly compare our proposed linear combination of joint-domain training to existing approaches, where the ratio between in- and out-domain training is adjusted by the out-domain corpus size. First we stepwise increase the amount of *TIGER* training data. Starting with a size equal to *WebTrain corpus* size, we randomly choose sentences in each step. This is performed 100 times and data is added to the data selected in the previous

Table 2: Tagger evaluation for different text types trained on joint-domain data.

	#Tokens	WebTagger (Neunerdt et al., 2013a) (*)		WebTagger (*) +Verblexicon $\mathcal{V}^+$ (-)			WebTagger (-) $+\beta = 10$ fact.	TreeTagger	Stanford
		Unknown	Total	Unknown	Verb	Total	Total	Total	Total
WebTrain test	3,628	76.06	94.38 ± 0.46	77.05	80.72	94.43 ± 0.47	$\mathbf{94.79 \pm 0.45}$	93.84 ± 0.55	93.50 ± 0.56
WebTypes	4,006	62.53	90.11 ± 0.11	62.89	60.96	90.12 ± 0.12	$\mathbf{90.63 \pm 0.12}$	88.02 ± 0.13	86.27 ± 0.10
TIGER test	88,910	88.87	97.22 ± 0.01	90.07	90.58	97.28 ± 0.01	$\mathbf{97.13 \pm 0.01}$	97.98 ± 0.01	98.69 ± 0.01

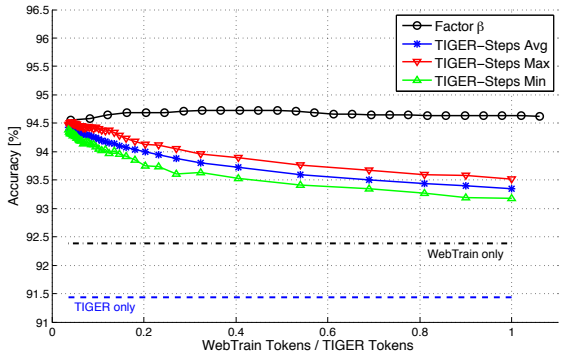

Figure 1: Influence of different joint-domain trainings evaluated on *WebTrain*.

step. Each of these out-domain training samples is combined with each training of a 10-fold *WebTrain* cross validation (3,600 tokens each part). Mean accuracies of cross validation tagging over all 1000 training samples are depicted for different in-/out domain ratios in the blue curve ($\star$) in Figure 1. Additionally the minimum and maximum accuracy of the 100 *TIGER* training samples is depicted in the green ($\triangle$) and red curve (∇). In order to give some reference values, we train our tagger exclusively on the *TIGER*/*WebTrain* corpus. Accuracies are depicted by the blue and black dotted line in both figures. Second we apply our linear combination approach and combine the *TIGER* and *WebTrain* corpus in the same cross validation for different β values. Cross validation results and test results achieved are depicted in the black curve ($\circ$). First, we compare the accuracies achieved with our approach (black curve, $\circ$) to those achieved with the best *TIGER* training part (red curve, ∇). The black curve ($\circ$) stays above the red curve (∇) over all in-/out-domain ratios. The red curve (∇) represents the optimum result for the given number of out-domain tokens. The plot indicates that exploiting this degree of freedom the performance of our approach is hardly reached. Determining the optimum training corpus results in a huge evaluation effort, which is very time consuming. If the *TIGER* training part is not determined properly and, e.g., chosen randomly, tagging accuracies can be significantly lower. In the worst case minimum accuracies depicted in the green curve ($\triangle$) are achieved. Applying our method with $\beta = 10$ results in a maximum cross validation accuracy of 94.79%. Determining the best β is considerably faster compared to identifying the best *TIGER* training part. Even if no effort is spent on determining the best β, accuracies are only slightly lower than optimum. Considering these evaluations it is obvious that our approach is robust in the sense that the performance slightly changes, if the ratio of tokens is changed. The result depicted in Figure 1 show the robustness of our method, no matter what β values we choose. Finally, we compare the results achieved for exclusively trained taggers on *TIGER*/*WebTrain* corpus. All combination methods significantly exceed accuracies achieved for single training over all in-/out domain ratios. This states that a joint-domain training approach is always reasonable.

6 Conclusion

We have compared state-of-the art taggers with our adapted WebTagger. It outperforms the others considerably with an average accuracy of 94.8% applied to a German social media text corpus. Additionally, it yields a minimum improvement compared to state-of-the art taggers of 2.6% percentage points for a social media text type corpus different from the training corpus type. In our approach we have improved the following two items of the original WebTagger. First we have amended the estimation of lexical probabilities for unknown tokens by introducing tag distributions derived from prefix and suffix information and a semi-supervised verb auxiliary lexicon. Second we have enriched the social media text corpus by a linear combination following an oversampling technique with a newspaper training corpus.

References

Sabine Brants, Stefanie Dipper, Peter Eisenberg, Silvia Hansen-Schirra, Esther König, Wolfgang Lezius, Christian Rohrer, George Smith, and Hans Uszkoreit. 2004. TIGER: Linguistic Interpretation of a German Corpus. *Research on Language & Computation*, pages 597–620.

Phani Gadde, L. V. 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*, pages 5:1–5:8.

Eugenie Giesbrecht and Stefan Evert. 2009. Is Part-of-Speech Tagging a Solved Task? An Evaluation of POS Taggers for the German Web as Corpus. In *Proceedings of the Fifth Web as Corpus Workshop*, pages 27–35.

Kevin Gimpel, Nathan Schneider, Brendan O'Connor, Dipanjan Das, Daniel Mills, Jacob Eisenstein, Michael Heilman, Dani Yogatama, Jeffrey Flanigan, and Noah A. Smith. 2011. Part-of-Speech tagging for Twitter: annotation, features, and experiments. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*, pages 42–47.

Claude Nadeau and Yoshua Bengio. 2001. Inference for the generalization error. *Maschine Learning*.

Melanie Neunerdt, Michael Reyer, and Rudolf Mathar. 2013a. A POS Tagger for Social Media Texts trained on Web Comments. *Polibits*, 48:59–66.

Melanie Neunerdt, Michael Reyer, and Rudolf Mathar. 2013b. Part-of-Speech Tagging for Social Media Texts. In *Proceedings of The International Conference of the German Society for Computational Linguistics and Language Technology*.

Olutobi Owoputi, Brendan O'Connor, Chris Dyer, Kevin Gimpel, and Nathan Schneider. 2012. Part-of-Speech Tagging for Twitter: Word Clusters and Other Advances. Technical report, School of Computer Science, Carnegie Mellon University.

Olutobi Owoputi, Brendan OConnor, 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.

L. Pelayo and S. Dick. 2007. Applying novel resampling strategies to software defect prediction. In *Fuzzy Information Processing Society, 2007. NAFIPS '07. Annual Meeting of the North American*, pages 69–72, June.

Ines Rehbein. 2013. Fine-grained pos tagging of german tweets. In *Language Processing and Knowledge in the Web*, volume 8105 of *Lecture Notes in Computer Science*, pages 162–175. Springer Berlin Heidelberg.

Helmut Schmid. 1995. Improvements in Part-of-Speech Tagging With an Application to German. In *Proceedings of the ACL SIGDAT-Workshop*, pages 47–50.

Kristina Toutanova, Dan Klein, Christopher D. Manning, and Yoram Singer. 2003. Feature-rich Part-of-Speech Tagging With a Cyclic Dependency Network. In *Proceedings of Human Language Technology Conference*, pages 173–180.

Bianka Trevisan, Melanie Neunerdt, and Eva-Maria Jakobs. 2012. A Multi-level Annotation Model for Fine-grained Opinion Detection in German Blog Comments. In *Proceedings of KONVENS 2012*, pages 179–188.