Improving POS Tagging of German Learner Language in a Reading Comprehension Scenario

Lena Keiper, Andrea Horbach, Stefan Thater
Saarland University
Department of Computational Linguistics
Saarbrücken, Germany
{lkeiper, andrea, stth}@coli.uni-saarland.de

Abstract

We present a novel method to automatically improve the accurrcy of part-of-speech taggers on learner language. The key idea underlying our approach is to exploit the structure of a typical language learner task and automatically induce POS information for out-of-vocabulary (OOV) words. To evaluate the effectiveness of our approach, we add manual POS and normalization information to an existing language learner corpus. Our evaluation shows an increase in accuracy from 72.4% to 81.5% on OOV words.

Keywords: Part-of-Speech Tagging, Learner Language, Corpus Annotation

1. Introduction

With the increasing availability of computer-assisted language learning (CALL) applications, there comes a growing need to develop and improve NLP tools that are able to process *learner language* automatically in order to give meaningful feedback to language learners. For most tasks involving free-text input by the learner (unlike for instance simple multiple choice questions or gap-filling tasks), applications need to deal with unrestricted learner language input using NLP techniques.

In this paper, we consider the preprocessing step of part-of-speech (POS) tagging of learner language. POS tagging is a fundamental part of most NLP tool chains and provides necessary input for higher-level processing steps such as algorithms for scoring the contents of learner answers.

The challenge is that out-of-the-box POS tagging models are usually trained on standard language like newspaper articles, and consequently also perform best on newspaper text. Learner language, however, typically differs substantially from the newspaper data, so that out-of-the-box-models are not directly applicable in a CALL setting. The standard model of the tagger considered in this paper, for instance, achieves an accuracy of over 97% when applied to newspaper text, but only 93% when applied to learner language.

One important reason for this performance drop is that learner language contains many out-of-vocabulary (OOV) tokens, i.e. tokens that the tagger has not seen during training. They are frequent in learner language for two reasons: One is that the domain of the learner language input differs from typical newswire texts. We call these OOV words *lexical gaps*. The second and more important reason is that learner language contains a lot of noise such as typos and other spelling errors, as well as grammar problems. They lead to nonexistent word forms which we call *misspellings*. Since learner language also tends to differ on the syntactic level, the tagger often cannot exploit contextual information to guess the correct POS tag for OOV as effectively as in the case of standard language.

Examples (1) and (2) illustrate these phenomena: The first example shows two typical types of misspelling errors at the lexical level: The noun "Erflog" (correct form: Erfolg, Eng.: success) is a spelling error with a letter swap; "geld" (correct form: Geld, Eng.: money) is written in lowercase although in German all nouns are capitalized. Due to these errors a standard tagger tags the two nouns as a verb in the former and an adjective in the latter case. In the second example, the noun "Dusche" (Eng.: shower) is correctly spelled, but it is a *lexical gap*, because the word does not occur in the training data. The tagger chooses erroneously to tag the word as an adjective instead of a noun.

(1)	**Learner:**	Viel	**Erflog**	und	**geld**	haben
	Normalized:	Viel	**Erfolg**	und	**Geld**	haben
	Tagger:	ADV	**VVFIN**	KON	**ADJA**	VAINF
	Gold:	ADV	**NN**	KON	**NN**	VAINF

(2)	**Learner:**	Der	Herd	und	die	**Dusche**
	Normalized:	Der	Herd	und	die	**Dusche**
	Tagger:	ART	NN	KON	ART	**ADJA**
	Gold:	ART	NN	KON	ART	**NN**

This paper makes two contributions. First, we describe a method to improve the POS tagging performance on learner language by automatically inducing POS information for OOV tokens. Second, we provide POS and normalization annotation on top of the CREG corpus (Meurers et al., 2011). We use the annotation to evaluate our method, but it is also a potentially useful resource in its own right.

As no substantially large learner language corpora are readily available and learner language lacks the systematicity of other non-standard texts (such as microposts from Twitter), most domain adaptation methods from related work like Han et al. (2012), Rehbein (2013) or Prange et al. (2015) are not applicable. Instead our approach is to address the problem by exploiting the structure of reading comprehension questions, a typical language learning task: Students' answers are linked to a reading text in standard language; when students answer reading questions, they rely on the given textual material and tend to repeat words or even copy whole phrases from the question or the reading text (referred to as *reference* texts below), a phenomenon known as *lifting*. Therefore learner answers tend to have a high lexical overlap with the reference texts. This has implications which we leverage in our tagging approach: Whenever a word in a learner answer does not occur in the tagger's vocabulary, we check whether this word or a similarly spelled word occurs in the reference. If that is the case, we assume that the learner really meant this word: We are therefore able to normalize misspellings to words occurring in the reference. We add lexical gaps that occur in the reference to the the tagger lexicon together with the POS tag that has been assigned to them by the tagger in the reference. As the reference texts are standard language and therefore tagged with a high degree of accuracy, we trust their POS annotation and propagate the POS label of OOV words in the reference back to the corresponding token in the learner answer.

Consider the following example of a learner answer that contains the OOV word *verlossen*. In the reference, the word as is does not occur; therefore we assume it to be a misspelling and not a lexical gap. While standard spell checkers would correct the word as either a form of *verlassen* or one of *verlieren (verloren)* we find the correct variant by comparing the word to all words occurring in the reference that contains *verloren*, but no instance of *verlassen*.

(3) **Learner:** Apfelwein ist einer traditionalle Wein für ein hundert Jahre. Eine Konsequenz ist Kultur **verlossen**. (...)
Text: (...) Würde dieser Begriff verboten, hätte das Land Hessen eines seiner bedeutendsten Identifikationssymbole **verloren** (...)

To evaluate the effectiveness of our approach, we manually annotate the learner answers within CREG, the Corpus of Reading Comprehension Exercises for German (Meurers et al., 2011) with POS tags and normalization information, i.e., correct word forms of incorrect learner words. The added value of our annotations compared to previous annotation efforts such as the FALKO corpus (Reznicek et al., 2012) is that we provide both manual POS tags and normalization information. It is also grounded in the structure of the CREG corpus, whose reference texts allow both a context-aware manual and automatic normalization. Our automatic tagging approach gives us an improvement of 10% on OOV words.

The remainder of this paper is organized as follows: We provide more background and an overview of related work in Section 2. We describe our corpus annotations in Section 3, and the design of our POS-tagging architecture in Section 4. We present an evaluation of the POS-tagging in Section 5, and conclude with Section 6.

2. Background and Related Work

Learner language can differ quite substantially from standard language. Deviations from the standard can occur on all levels, including, but not limited to spelling, lexicon, syntax and morphology. They are not universal for all learners, but depend on factors such as native language, current level in the foreign language and learning strategies. Selinker (1972) coined the term *interlanguage* for these language variants of individual learners.

Those differences from standard language affect automatic POS tagging most notably on the areas of spelling variance, punctuation, morphology and word order. However, individual differences make it hard to build one POS tagger model for all learners. Instead of building a single tagger model, we therefore make use of what we know about the context of the learning task and exploit information from the reference material to make assumptions on the target hypothesis. In doing so, we adopt a common way of coping with learner language in NLP applications: the integration of a normalization (spelling error correction) step into a linguistic pipeline that tries to bring the input closer to standard language.

The task of (manual as well as automatic) normalization of learner language has been addressed in several works, for German data most notably in the FALKO corpus (Reznicek et al., 2012). This corpus consists of summaries and essays written by language learners and native speakers. Similar to our annotations, the FALKO corpus provides (manual) normalization information on several linguistic levels. Their *minimal target hypothesis* has the aim of transforming the text into a parsable structure to enable automatic processing, while the *extended target hypothesis* also remedies errors on semantic, lexical, pragmatical and stylistic levels. The corpus also comes with POS information, but unlike us they do not provide a manual POS annotation, but use automatically assigned POS tags on the minimal target hypothesis.

Reznicek and Zinsmeister (2013) provide a study about automatic POS tagging performance for learner texts from a subcorpus of FALKO. However, they evaluate only those tokens where an ensemble of taggers disagree and manually annotate only a very small data sample of learner essays.

POS annotation of learner language with a completely different goal has been approached by Díaz-Negrillo et al. (2010), who annotated NOCE, an English learner corpus by Spanish learners. Their approach differs from ours in that they are not normalizing learner language into standard language, but explicitly deal with properties of learner language by annotating separately the three individual sources of evidence for a POS tag: lexicon, morphology and distribution. This approach allows them to identify sources of errors and to query the corpus searching for particular learner language phenomena and, is thus a valuable resource for both researchers and teachers in the study of learner language. With our different goal of improving POS tagging to enable automatic linguistic processing, we are instead trying to fit learner language into the framework of a standard tagset in order to enable higher NLP processing steps.

A different way to look at POS tagging of learner language is to see it as a problem of domain adaptation. Recent work on domain adaptation has focused on Computer-Mediated Communication (CMC) data. For instance, Gadde et al. (2011) leverage word clusters based on surface similarity to link OOV words from an SMS corpus to known words and, similar to us, they also use language models to find the most plausible normalized sentence variant as a preprocessing step for tagging. Han et al. (2012) create a normalization dictionary for OOV words from English Twitter data based on distributional similarity and rank them based on string similarity.

We see one main difference between our learner language corpus and other resources of non-standard language: For the methods described above to work, OOV words have to be frequent enough in some untagged corpus that they can be covered in e.g. distributional models. For many CMC domains such as Twitter, large untagged corpora are available and thus many OOV tokens indeed occur with a frequency that allows relation to known words to be learned from unannotated data. Such approaches do not work in our case because of the small size of our corpus and the fact that individual learner errors are not phenomena that occur with the same frequency as deviations in CMC data. We overcome this problem by instead leveraging information from the reading material that narrows down the pool of potential replacement candidates.

3. Corpus and Annotations

This section introduces one of the key contributions of our paper: the annotation project that adds both part-of-speech and normalization information to the learner answers in the CREG corpus.

We organize the annotation into two subsequent steps. In a first step, we normalize the input, i.e., we replace incorrect words in the learner answers by the words that the learner presumably wanted to use (the *target hypothesis*). In a second step, we then label the words in the learner answer with POS information based on the target hypothesis. The normalization step is a necessary prerequisite to POS annotation since the POS annotation should reflect what the learner intended to express (on the lexical level). We thus decided to make the target hypothesis explicit as a separate annotation layer in order to make the POS annotations as transparent as possible and also to provide a gold standard for automatic normalization approaches.

3.1. Data

We use CREG, the Corpus of Reading Comprehension Exercises (Meurers et al., 2011), as the basis of our annotations. It is the main German corpus for the task of short-answer scoring (Ziai et al., 2012; Koleva et al., 2014) and consists of 1032 learner answers (*LAs*) given to 177 questions about a total of 32 reading text, as well as teacher-specified target answers. The data had been collected in German language courses at two universities in the United States, all learners were American native speakers. For our annotations, we focus on the learner language material, i.e. the learner answers. We use only those answers given primarily in German. Answers had been transcribed twice in the corpus; we always use the first transcript and tokenized it using the Stanford CoreNLP tokenizer (Manning et al., 2014). The tokens we thereby get for our annotation study sum up to a total of 12175. In our approach to automatically improve POS tagging on learner answers, we also make use of the additional material, which consists of standard language data that is lexically related to the learner language data.

3.2. Normalization

In cases where a learner answer deviates from standard language we asked our annotators to form a target hypothesis, i.e., to formulate what the language learner presumably intended to say. We distinguish two normalization levels: on the first level (N1), we normalize misspellings; on the second level (N2), we additionally normalize grammatical errors such as incorrect case assignments or missing articles or prepositions. Our system, described in the next section, uses information from level N1 only; level N2 is used only in the evaluation to estimate an upper bound of tagger performance.

Consider Example 4, where the learner used “das”, which is the definite article in standard German, but most likely wanted to use the subordinate conjunction “dass” (*that*). This mistake was potentially due to the phonological similarity of the two words. While a word with the surface form *das* in German can only be tagged as an article, relative or demonstrative pronoun, only the form *dass* can occur as a conjunction and was presumably intended in this sentence. Therefore, the annotator first normalized *das* to *dass* before tagging the word as subordinate conjunction KOUS. Additionally, we see in this example a normalization where the learner uses an untypical spelling for the German umlaut in “für” and mixed up accusative with dative for the personal pronouns.

(4)										
	LA:	Sie	dachte	**das**	es	war	nicht	**fuer**	**ihr**	.
	N1:	Sie	dachte	**dass**	es	war	nicht	**für**	**ihr**	.
	N2:	Sie	dachte	**dass**	es	war	nicht	**für**	**sie**	.
	POS:	PPER	VVFIN	**KOUS**	PPER	VAFIN	PTKNEG	APPR	PPER	$.

She thought that it was not for her.

Our perspective on normalization as a prerequisite for POS annotation motivates our main annotation guideline: Whenever possible we only normalize on token level, i.e., we do not insert or delete words and especially do not change word order, apart from the following three exceptions:

- separate one (accidentally fused) word into two individual words (“einsman” becomes to “eins man”)
- combining two words into one word, mostly for German compound nouns which have to be written as one token, but are often split by the English native learners (“Polizei Gewalt” becomes to “Polizeigewalt”)
- adding and deleting articles and prepositions

Unlike Reznicek et al. (2012) we do not concern ourselves with lexical or pragmatic aspects in the normalization.

3.2.1. Annotation Process

Annotations were carried out by two trained computational linguistics students (native speakers of German). For each learner answer they had access to the reference material, so that they could form their target hypothesis based on the context of an answer. Cases of disagreement were checked by a third annotator, one of the authors, who adjudicated instances that were clearly annotation errors. As Lüdeling (2008) pointed out, it is often difficult, if not impossible, to find exactly one target hypothesis. Therefore, if two different target hypothesis were both plausible they were both kept as alternatives, to maintain the diversity of potential linguistic interpretations of an answer.

3.2.2. Analysis

On the binary task of whether an item has to be normalized or not, annotators reached an inter-annotator agreement of $\kappa = 0.78$ for normalizations in N1 and 0.68 for normalizations in N2, indicating a substantial agreement according

	IV	OOV
N1 or N2	9% (1023)	35% (452)
N1	3% (369)	32% (414)
N2	6% (691)	6% (83)

Table 1: Number of all IV and OOV that were normalized on N1 and N2 by at least one annotator

to Landis and Koch (1977). For those tokens where the annotators agreed to normalize them, they produced the same annotation in 86.2% of all normalizations in level N1, and 89.2% of all normalizations in level N2.

In the adjudication step, 47% of all disagreements were resolved into only one correct form, while for the remaining 53% both normalizations were kept as plausible.

After adjudication, 12.1% of all tokens (1475) had been normalized by at least one annotator and 10.0% (1220) by both. To test the influence of normalization on automatic POS tagging, we tagged both the normalized and the original version of the data with a standard tagger. 27.3% of all normalized tokens changed their POS tag between these two runs.

When dividing tokens into those that are in the lexicon of a standard tagger and those that are not, we see that 35% of all OOV tokens were normalized by at least one annotator (32% on N1 and 6% on N2), and only 9% of the in-vocabulary (IV) tokens. 82 tokens were normalized on both levels.

The semantic correctness of an answer, however, has no influence on the number of normalizations; correct learner answers contain on average as many normalizations as incorrect answers. This is a plausible result, given that teachers are instructed to ignore spelling errors when scoring short-answer questions.

39% of all orthographic normalizations (N1) concerned capitalization issues and 11% German umlaut spelling. Although we allowed some operations beyond token level, they occur rarely in our annotations: Only in 7.3% of all normalizations in N1 did a token have to be split or two tokens were merged, and on level N2 only 14 tokens (1.8%) were deleted and 52 (6.7%) inserted.

3.3. Part-of-Speech Annotation

In the second annotation step, learner answers were annotated with POS tags using the Stuttgart-Tübingen tagset (Schiller et al., 1999). We extended the standard tagset with one extra tag, *LL*, for learner language that annotators could always use when they felt that the language used was so corrupt that no other tag would fit the token.

3.3.1. Annotation Process

All material were annotated by the same two student annotators with access to the reference material. As described in the normalization section, we tried to stay as close as possible to the surface form of each token and correct what we assumed to be spelling mistakes. POS tags were then selected in such a way that they fit this normalized version (step 1) of a token.

	Newspaper	LA	References
NN	22%	22%	16%
ART	11%	10%	8%
APPR	9%	5%	6%
ADJA	6%	3%	3%
NE	6%	4%	3%
ADV	4%	2%	5%
VVFIN	4%	4%	7%
KON	3%	3%	2%
VAFIN	3%	4%	2%
PPER	2%	3%	6%
REST	30%	40%	42%

Table 2: The most frequent POS tags

3.3.2. Analysis

The two annotators reached an almost perfect agreement of $\kappa = 0.95$, even if they found different normalizations. All disagreements (577) were reannotated by a third annotator. A majority vote between these three annotators was used to determine the final POS tag. 17 remaining cases of disagreement were adjudicated by the authors. After adjudication 58 tokens with two possible normalizations had two different POS tags.

The tag LL was used in 38 cases. The following table shows some of the occurrences.

(5) a) Es ist etwas die besser **pkte** wohnenen.

b) **Machen** sind 66,1 Prozent Frauen.

c) Die Salzburger lassen etwas der Brunnen im Winter **frhn** zu werden.

d) Man muss deutscher Staatsbrger sein **zu** eine GmbH gegrndet werden.

Table 2 compares the frequency of the 10 most frequent POS tags in newspaper texts, our annotated learner data and the reference texts. We can see that learner data and reference texts have a similar distribution of POS tags that differs from that of newspaper texts.

4. System Description

This section describes the architecture of our automatic tagging system. The general aim is to minimize the number of words that are OOV to the tagger, as tagging accuracy for these words is generally much lower than for IV words. We leverage two methods to achieve this goal: One is to correct *misspelled words* before tagging into their most likely correct form, while the second is to extend the tagger lexicon with OOV words that occur in the reference material of a learner tex; we call them *lexical gaps*. We first describe the decision process, for which method will be applied, and then discuss both variants of dealing with OOV words in more detail.

We use the TnT tagger (Brants, 2000) in all of our experiments, and the TIGER corpus (Brants et al., 2004) is the newspaper corpus, which we use to train a standard tagger model.

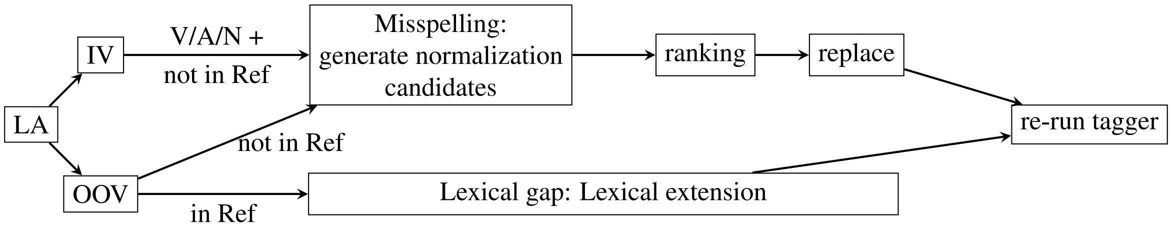

Figure 1: System overview with the handling of misspellings (upper branch) and lexical gaps (lower branch)

4.1. Decision Between Lexical Gaps and Misspellings

The lexicon of a POS tagger depends on the training corpus, i.e. all words not seen during training are OOV. Words can be OOV for two different reasons: First, it is possible that a perfectly correct word is a lexical gap and just does not occur in the training data. Second, words that are misspelled are also OOV to a tagger that has been trained on standard newspaper text.

The references for a learner answer are one major resource for deciding whether a word belongs to the first or second of these categories. In our corpus, 84% of all tokens in the normalized learner answers also occur in their references. This supports the claim that learners lift material from the reference into their answers. Therefore we apply the following rule: If a word that is OOV to the tagger occurs (in some inflectional form) in the reference it is likely that the learner intended to use this word (instead of misspelling a different word). We will therefore treat such a word as a lexical gap, and words that do not occur in the related material will be treated as potential misspellings.

In the corpus, about 59% of all OOV words indeed occur in the references and are treated as lexical gaps; the others are treated as potential spelling errors.

4.2. Lexicon Extension for Lexical Gaps

We determine the lexical gaps by the rule described above. Next, we determine the POS tag(s) with which words are added to the tagger lexicon. We exploit the fact that the references are mostly well-formed texts and that TnT is able to guess the correct POS tag for OOV words using suffix and context information. A sample annotation of 1000 tokens of references showed a tagging accuracy of 90% for OOV words from the references.

Inspecting tagging results on the learner answers, we find that the accuracy on words that are lexical gaps is much lower (81%). Our strategy therefore is to retrieve the tags that have been assigned to lexical gaps in the references and include them in the lexicon. If a word occurs multiple times in the the text with different POS tags, we save all of them, i.e. the word is treated as ambiguous.

4.3. Automatic Normalization of Misspellings

After adding lexical gaps to the lexicon, we treat the remaining OOV words as potential misspellings.

4.3.1. Candidate Generation

We exploit the fact that most tokens in the LA stem from the reference: We compare each remaining OOV word with all words from its reference and collect all words with a Damerau-Levenshtein distance below some threshold as normalization candidates. We use a slightly modified version of the distance measure that assigns lower penalties to frequent learner issues such as capitalization and problems with German umlauts. Moreover, we compare not only to the specific word form that occurs in the reference, but extract all word forms for each lemma that is known in the TIGER corpus and compare to all of them. That means if we encounter the OOV form *verlossen* and have the verb infinitive *verlieren* in the reference, we compare *verlossen* also to other inflectional forms of *verlieren* that occur in TIGER, such as *verloren*.

This method only deals with OOV words, but about 25% of orthographic corrections in our manual normalizations concern tokens whose surface form is known in the training data, for example the correction from “das” to “dass”. Therefore it is desirable to correct such misspellings that result in an IV word as well. As it might, however, introduce noise to treat every word as potentially misspelled, we restrict ourselves to IV words which do not occur in the reference and normalize them against the TIGER lexicon. We only normalize words that have been tagged as content words, because function words are often very short and similar on the surface level, so that they result in a large number of orthographic neighbors.

We determine the thresholds for normalization with the reference and with TIGER individually via 10-fold cross-validation on the complete dataset.

4.3.2. Candidate Ranking

We use language models to choose between the different normalizations option for a token, including the original form of the token: We combinatorically enumerate all possible normalized versions of a sentence and run them though a language model built with the SRILM toolkit (Stolcke, 2002) in order to retrieve normalizations that fit the sentence context. The language model has been trained using the Mannheimer Corpus[1] and the German part of the Wikipedia Corpus (Margaretha and Lüngen, 2014) as well as all the reading texts from the references. We keep those normalizations that constitute the variant of the sentence with the lowest perplexity.

Consider the following example sentence where two words have been automatically normalized, one with only one alternative and one with two, here listed in parentheses.

(6) Eine europäische Studie daüber (drüber, **darüber**) , worauf sie nicht vorzichten (**verzichten**) könnten

[1] http://www1.ids-mannheim.de/kl/projekte/korpora/archiv/mk.html

	Number of Tokens	Accuracy
Original LA	12175	92.8%
IV	10902	95.2%
OOV	1273	72.4%
Normalized LA	12196 (- 12)	95.5% (+ 2.7)
IV	11174 (+ 270)	96.6% (+ 1.4)
OOV	989 (- 282)	82.4% (+ 10.0)

Table 3: Accuracies for an out-of-the box tagging model on the original and the normalized data on OOV and in-vocabulary (*IV*) tokens.

We feed all six combinatorical variants of the sentence into the language model and find the words in bold print as the ones with the highest probability and, choose them as the normalization.

5. Experiments

5.1. Experiment 1: Baselines and Upper Bounds

To establish a baseline, we evaluate the out-of-the-box model trained on the TIGER corpus on our annotated gold standard. As an upper bound for the normalization component, we also run the model on the normalization gold-standard version of the data. Results are reported in Table 3.

Compared to tagging accuracy on standard texts of 96 to 97%, the tagger performs significantly worse on our data set. Unsurprisingly, the performance is much better for the normalized LA, the accuracy gain from 92.8% to 95.5% bringing us back into the region of tagger performance on standard text.

The accuracy on OOV tokens increases by 10% to 82%, both because more precise contextual information leads to better tagging results and because normalization leads to a better performance of suffix heuristics used for OOV words. Also for IV tokens, we can observe an accuracy gain. One reason is that normalization from one IV token to another increases accuracy: E.g. the conjunction *dass* misspelled as *das* is always mistagged as an article or pronoun, whereas the normalized version can be correctly tagged.

The difference in the total number of tokens between the two evaluations is due to some normalizations that split, merge or insert tokens. The number of OOV tokens is obviously reduced because many normalizations of misspellings result in IV words. Among the remaining 989 OOV tokens, 1.1% were tokens in which the learner language was so corrupt that the annotators were not able to find a normalization.

We also evaluated for comparison the performance on the original partitioning of tokens into OOV and IV under the out-of-the-box model and see very similar results.

5.2. Experiment 2: Evaluating our Tagging Approach

Table 4 shows the performance of our full system (+Norm+Lex) compared to the TIGER baseline. We reach an accuracy improvement of 1% for all and 9.1% for OOV tokens. The improvement is statistically significant according to a McNemar test (p<0.001)

	Accuracy
TIGER	92.8%
IV	95.2%
OOV	72.4%
+Norm+Lex	93.8% (+ 1.0)*
IV	95.3% (+ 0.1)
OOV	81.5% (+ 9.1)*
+Norm	93.7% (+ 0.9)*
IV	95.3% (+ 0.1)
OOV	80.7% (+ 8.3)*
+Lex	92.8% (+ 0.0)
IV	95.2% (+ 0.0)
OOV	72.6% (+ 0.2)

Table 4: Accuracy of our system (+Norm+Lex), compared to the TIGER baseline, and to variants that use just one component. * denotes improvement compared to TIGER that is significant according to a McNemar test (p<0.001)

	Accuracy
TIGER	92.8%
IV	95.2%
OOV	72.4%
+Gold	93.2% (+ 0.4)*
IV	95.4% (+ 0.2) *
OOV	74.3% (+ 2.1) *
+Norm+Lex	93.8% (+ 1.0) **
IV	95.3% (+ 0.1)
OOV	81.5% (+ 9.1)**

Table 5: Accuracy of a straightforward retraining approach compared to our system. * denotes improvement compared to TIGER that is significant according to a McNemar test (p<0.001); ** denotes improvements compared to TIGER and +Gold that are significant according to a McNemar test (p<0.001)

5.3. Experiment 3: Retraining the Tagger

One obvious alternative approach for adapting a tagger to a new domain is to train it on in-domain training data. Following Horbach et al. (2014), we add two-thirds of our annotated data to the TIGER corpus (*+Gold*), retrain the tagger models and evaluate on the remaining 4000 tokens. Table 5, however, shows that this approach performs significantly worse than our system.

5.4. Evaluation of Individual System Components

To assess the performance of our system's components, we also evaluate them individually: The last two blocks of Table 4 show the results if we run our system with only one of the two components. We can see that the contribution of the normalization is much more pronounced than that of lexical extension, and that the combination of the two brings some additional advantage over the individual improvements.

	D-L-Distanz+Ref	Aspell
Precision	0.85	0.51
Recall	0.57	0.44
F-Score	0.69	0.47
Correct Token	0.86	0.54
Correct Lemma	0.89	0.62

Table 6: Aspell vs. Damerau-Levenshtein Distance; upper half: number of normalizations on the right tokens; bottom half: number of all tokens with the right normalization

Performance of Lexicon Extension Additionally, we evaluate the automatic generated lexicon extensions: Two human annotators determine the POS tags of the added words as a gold standard for our automatically retrieved tags. We found that about 90% of all words are correctly tagged. Frequently, errors are confusions between normal nouns and named entities.

Performance of Normalization We compare our normalization method against our gold-standard normalizations and compare the results to a baseline produced by running the spell checker GNU Aspell[2] and taking the first proposed normalization. In Table 6 the results are summarized: First we evaluate the binary decision on whether a token should be normalized in terms of precision, recall and F-score. Next, if a token is normalized both in the gold standard and by our normalizer or Aspell respectively, we compare how often the correct normalization is found. In both aspects our system produces better results than a state-of-the-art spell checker. The F-score increases by 20%. If a normalization is in the right place, in 86% of all cases it is exactly the same as in the manual annotation for our system.

Evaluation of Language Models The normalization alternatives were ranked by a language model. The language model distinguishes between an average of 2.2 alternatives. In 86% of all cases one of them is the correct one, and in 69% the language model ranks the correct one highest.

5.5. Analysis

Table 7 shows precision, recall and F-score for the out-of-the-box model and the improvements by our approach. For the sake of simplicity, we merged similar tags, e.g. all verb tags, into a coarser-grained label. We can make two observations: First, the performance of the TIGER model on learner data is in general reduced; there are no prominent outliers, apart from quite infrequent classes. Second, the improvement we get from our approach manifests across all POS tags.

	TIGER			+Norm+Lex		
POS	Precision	Recall	F-Score	Precision	Recall	F-Score
Adjective	93.7	84.5	88.8	+01.0	+ 07.5	+04.5
Adverb	93.4	84.0	88.4	+01.0	+00.7	+00.9
Preposition	96.4	97.4	96.9	+02.3	-00.7	+00.8
Determiner	98.9	97.9	98.4	+00.0	-00.1	-00.1
Cardinal	92.2	93.3	92.7	+02.8	-00.9	+01.0
FM	76.8	65.5	70.7	+00.0	+03.6	+02.0
Conjunction	95.0	96.6	95.8	+00.6	-00.2	+00.2
Noun	96.3	97.8	97.0	+01.5	+ 00.2	+00.9
Pronoun	92.5	96.5	94.4	+01.0	-00.6	+00.3
Particle	96.9	95.1	96.0	+00.4	+00.4	+00.4
TRUNC	75.0	100.0	85.7	+00.0	+00.0	+00.0
Verb	97.6	96.6	97.1	-00.6	+01.5	+00.4
XY	99.8	99.8	99.8	-00.2	+00.1	+00.0

Table 7: Precision, recall, and F-score percentage values for the out-of-the-box TIGER model and changes in performance for our approach

6. Conclusions

In this paper, we addressed the task of POS tagging for learner language. We presented both normalization and POS annotation for the CREG corpus, providing a gold standard as an evaluation basis for further POS tagging and normalization approaches.

The structure of the corpus, i.e. the fact that every piece of learner language is an answer to a specific question about a text, allows us to make educated guesses if a token is OOV a standard tagger, both through lexical extension and normalization. In doing so, we get a significant improvement in tagging performance from 92.8 to 93.8% . On OOV words alone, we improve from 72.4 to 81.5%.

7. Acknowledgements

We thank the three anonymous reviewers for their helpful comments. We also thank our annotators Sophie Henning and Maximilian Wolf for their help with this study.
This work was funded by the Cluster of Excellence “Multimodal Computing and Interaction” of the German Excellence Initiative.

8. Bibliographical References

Brants, S., Dipper, S., Eisenberg, P., Hansen, S., König, E., Lezius, W., Rohrer, C., Smith, G., and Uszkoreit, H. (2004). Tiger: linguistic interpretation of a german corpus. *Research on Language and Computation*, 2(4):597–620.

Brants, T. (2000). Tnt – a statistical part-of-speech tagger. In *Proceedings of the Sixth Conference on Applied Natural Language Processing*, pages 224–231, Seattle, Washington, USA, April. Association for Computational Linguistics.

Díaz-Negrillo, A., Meurers, D., Valera, S., and Wunsch, H. (2010). Towards interlanguage pos annotation for effective learner corpora in sla and flt. *Language Forum*, 36(1–2):139–154. Special Issue on Corpus Linguistics for Teaching and Learning. In Honour of John Sinclair.

Gadde, P., Subramaniam, L. V., and Faruquie, T. A. (2011). Adapting a WSJ trained part-of-speech tagger to noisy text: preliminary results. In *Proceedings of the 2011 Joint*

[2] aspell.net

Workshop on Multilingual OCR and Analytics for Noisy Unstructured Text Data, page 5. ACM.

Han, B., Cook, P., and Baldwin, T. (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.

Horbach, A., Steffen, D., Thater, S., and Manfred, P. (2014). Improving the performance of standard part-of-speech taggers for computer-mediated communication. In *Proceedings of the 12th edition of the KONVENS conference Vol. 1. - Hildesheim.*

Koleva, N., Horbach, A., Palmer, A., Ostermann, S., and Pinkal, M. (2014). Paraphrase detection for short answer scoring. In *Proceedings of the third workshop on NLP for computer-assisted language learning at SLTC 2014*, Uppsala, Sweden.

Landis, J. R. and Koch, G. G. (1977). The measurement of observer agreement for categorical data. *Biometrics*, 33(1):pp. 159–174.

Lüdeling, A. (2008). Mehrdeutigkeiten und kategorisierung: Probleme bei der annotation von lernerkorpora. *Fortgeschrittene Lernervarietäten*, pages 119–140.

Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S. J., and McClosky, D. (2014). The Stanford CoreNLP natural language processing toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, pages 55–60.

Margaretha, E. and Lüngen, H. (2014). Building linguistic corpora from wikipedia articles and discussions. *JLCL*, 29(2):59–82.

Meurers, D., Ziai, R., Ott, N., and Kopp, J. (2011). Evaluating answers to reading comprehension questions in context: Results for german and the role of information structure. In *Proceedings of the TextInfer 2011 Workshop on Textual Entailment*, pages 1–9, Edinburgh, Scottland, UK. Association for Computational Linguistics.

Prange, J., Thater, S., and Horbach, A. (2015). Unsupervised induction of part-of-speech information for oov words in german internet forum posts. In *proceedings of NLP4CMC workshop*.

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

Reznicek, M. and Zinsmeister, H. (2013). Stts-konfusionsklassen beim tagging von fremdsprachlernertexten. *JLCL*, 28(1):63–83.

Reznicek, M., Ldeling, A., Krummes, C., Schwantuschke, F., Walter, M., Schmidt, K., Hirschmann, H., and Andreas, T. (2012). Das falko-handbuch korpusaufbau und annotationen version 2.01.

Schiller, A., Teufel, S., Stckert, C., and Thielen, C. (1999). Guidelines fr das tagging deutscher textcorpora mit stts. Technical report, Universitt Stuttgart, Universitt Tubingen.

Selinker, L. (1972). Interlanguage. *International Review of Applied Linguistics in Language Teaching*, 10(1–4):209–232.

Stolcke, A. (2002). Srilm - an extensible language modeling toolkit. In *Proceedings International Conference on Spoken Language Processing*.

Ziai, R., Ott, N., and Meurers, D. (2012). Short Answer Assessment: Establishing Links Between Research Strands. In *Proceedings of the 7th Workshop on Innovative Use of NLP for Building Educational Applications (BEA7)*, pages 190–200, Montreal, Canada. Association for Computational Linguistics.

9. Language Resource References

Meurers, Detmar and Ziai, Ramon and Ott, Niels and Kopp, Janina. (2011). *Corpus of Reading Comprehension Exercises in German*. SFB 833: Bedeutungskonstitution - Dynamik und Adaptivität sprachlicher Strukturen, Project A4, Universität Tübingen, CREG-1032.