The KiezDeutsch Korpus (KiDKo) Release 1.0

Ines Rehbein, Sören Schalowski and Heike Wiese
Potsdam University
German Department, SFB 632 "Information Structure"
irehbein|heike.wiese@uni-postdam.de

Abstract

This paper presents the first release of the KiezDeutsch Korpus (KiDKo), a new language resource with multiparty spoken dialogues of Kiezdeutsch, a newly emerging language variety spoken by adolescents from multiethnic urban areas in Germany. The first release of the corpus includes the transcriptions of the data as well as a normalisation layer and part-of-speech annotations. In the paper, we describe the main features of the new resource and then focus on automatic POS tagging of informal spoken language. Our tagger achieves an accuracy of nearly 97% on KiDKo. While we did not succeed in further improving the tagger using ensemble tagging, we present our approach to using the tagger ensembles for identifying error patterns in the automatically tagged data.

Keywords: spoken language corpora; urban youth language; Kiezdeutsch

1. Introduction

Linguistically annotated corpora are an essential basis for (quantitative) studies of language variation. However, most language resources are based on canonical written language, often from the newspaper domain, while only few corpora exist which are large enough for investigating variation in spoken language. The reasons for this are obvious. Written text is easy to come by in an already digitised format, whereas the creation of spoken language corpora requires time-consuming preprocessing. Besides the highly cost-intensive transcription process, applying automatic preprocessing tools like POS taggers and syntactic parsers to spoken language also results in a substantially lower accuracy than the one we can expect for canonical, written text, as these tools are usually trained on data from a written register.

This decrease in accuracy is partly due to data sparseness, caused by the high number of different pronunciation variants for each canonical lexical form. In addition, we observe elements not typically used in written language and thus not known to the preprocessing tools. For instance, in spoken language we find a great number of filled pauses like *uh, uhm*, backchannel signals (*hm, m-hm*), question tags (*ne, wa, gell*) and interjections. Many morphological and syntactic structures typical for spoken language are also not covered by the training data, which again leads to a decrease in tagging accuracy. Examples are cliticisations, exclamations, verbless utterances, or non-canonical word order, for instance verb-second word order in subordinate sentences with *weil* (because). In nonstandard dialects, there will be additional lexical and grammatical characteristics that might cause problems, such as specific lexemes, different inflectional patterns or syntactic options. Furthermore, the different distribution of lexical elements in the (written) training data and in spoken language results in erroneous tagger predictions. Finally, when working with informal spoken data, we also have to deal with abandoned utterances, unfinished words, and repairs.

The contribution of our paper is threefold. First of all, we present a new resource for general investigations of spoken, informal youth language and, in particular, for investigations of language use in monolingual and multilingual urban settings. Second, the new corpus provides training data for the development or adaptation of POS taggers for informal spoken language. Finally, we present our efforts to improve a POS tagger for spoken, informal German and to automatically detect tagging errors in the corpus.

2. Kiezdeutsch – the data

Kiezdeutsch (*'hood German*) is a new variety of Germany emerging in multiethnic urban neighbourhoods (Wiese, 2009; Wiese, 2013). This urban dialect is characteristic of informal peer-group conversations among adolescents, and is spoken across multilingual and monolingual speakers and different heritage language backgrounds. The linguistically highly diverse context in which it emerges, with its wealth of language contact opportunities, makes Kiezdeutsch more open to variation and innovation and results in a special linguistic dynamics. Kiezdeutsch thus offers a special access to ongoing tendencies of language development and change in contemporary German.

The lexical and grammatical features that make it interesting for linguistic investigations at the same time also constitute a challenge for automatic annotation. As a new, emerging dialect, Kiezdeutsch shows characteristic features at phonological/phonetic, lexical, and grammatical levels, such as some non-canonical pronunciation patterns (e.g., coronalisation of [ç]), the development of new particles, the integration of new loan words from other languages, some non-canonical inflections, variations in the use of functional categories such as articles and pronouns, and new word order options (for overviews cf., e.g. Wiese (2009; 2013), Auer (2013), and references therein).

(1) and (2) give some linguistic examples from the corpus material,[1] illustrating the occurrence of bare NPs for local expressions ((1); in contrast to Standard German

[1]Capitalisation indicates main stress; speakers' codes include information on corpus part (first two letters: in this, case, all data is from the multiethnic main corpus, "Mu"), gender (last but one letter: all speakers are male, "M"), and family/heritage language (last letter, in the examples above: "A" for Arabic, and "D" for German).

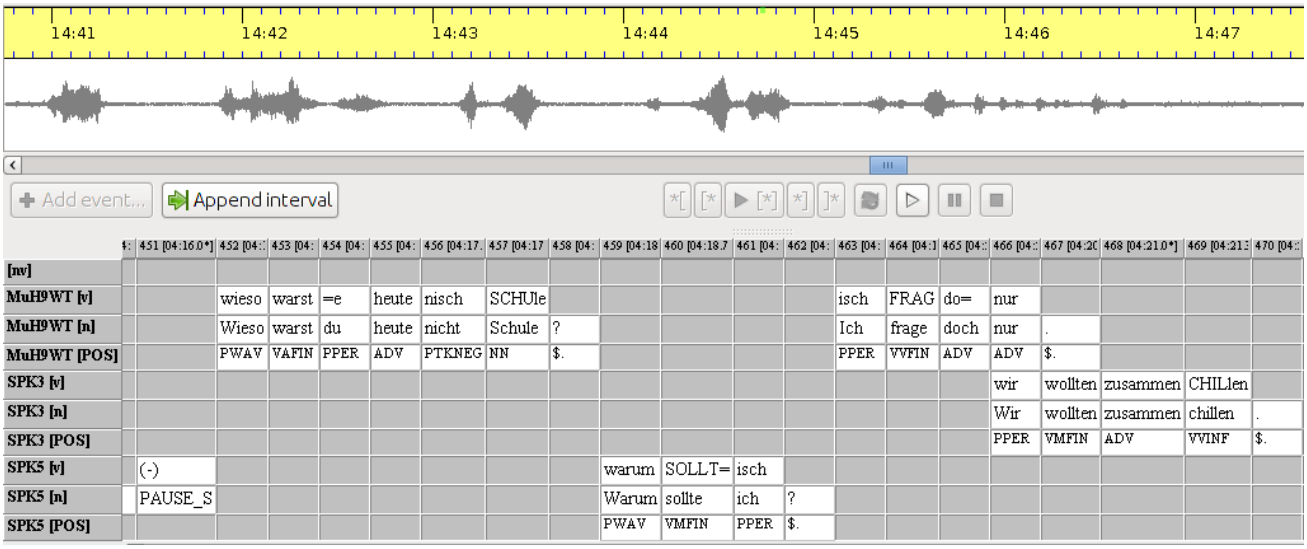

Figure 1: Screenshot KiDKo sample of a short dialogue between 3 speakers (MuH9WT, SPK3, SPK5) in EXMARaLDA (non-verbal layer (nv), transcription (v), normalisation (norm) and POS) (engl. transliteration: MuH9WT: Why were you today not school ? *"Why weren't you at school today?"* SPK3: Why should I ? *"Why should I?"* MuH9WT: I ask PTCL just . *"I'm just asking."* SPK5: We wanted together chill. *"We wanted to chill together."*)

PP[DP[NP]]), coronalisation ((1); *isch* instead of Standard German *ich*), and the option to use two constituents (1) or none (2) before the finite verb in declarative main clauses (in addition to the option of using exactly one constituent, which would lead to canonical verb-second word order).

(1) GEStern isch war KUdamm
yesterday I was Kudamm
"Yesterday I was (at the) Kudamm."
. [KiDKo, MuH25MA]

(2) brauchst du VIER alter
need you four old.one
"You need four of those, man!"
(= parts for building virtual cars in a computer game) [KiDKo, MuH11MD]

The data was collected in the first phase of project B6 "Grammatical reduction and information structural preferences in a contact variety of German: Kiezdeutsch" as part of the SFB (Collaborative Research Centre) 632 "Information Structure" in Potsdam. It contains spontaneous peer-group dialogues of adolescents from multiethnic Berlin-Kreuzberg (around 266,000 tokens) and a supplementary corpus with adolescent speakers from monoethnic Berlin-Hellersdorf (around 111,000 tokens, excluding punctuation). On the normalisation layer where punctuation is included, the token counts add up to around 359,000 tokens (main corpus) and 149,000 tokens (supplementary corpus). For a more detailed description of the data see (Wiese et al., 2012). The current, second and final, phase of the project is dedicated to corpus compilation including annotation.

2.1. Corpus architecture

The current version of the corpus contains the audio signals aligned with transcriptions. The data was transcribed using an adapted version of the transcription inventory GAT 2 (Selting et al., 1998), also called GAT *minimal transcript*, which includes information on primary accent and pauses. Release 1.0 of KiDKo also includes a level of orthographic normalisation where non-canonical pronunciations, punctuation, and capitalisation are transferred to Standard German spelling, as well as a layer of annotation for part-of-speech tags (Section 3.).[2]

The normalisation layer is necessary for different reasons. First, the normalised version of the data allows users to search for all pronunciation variants of a particular word and thus increases the usability of the corpus. Second, it provides the input for automatic POS tagging, which considerably reduces the number of unknown words in the data and thus increases tagging accuracy considerably. The normalised version of the data, however, should be considered as an annotation and thus as an interpretation of the data. Often, missing context information or poor audio quality (caused by noisy environments) complicate the transcription and license different possible interpretations of the same audio sequence. Here, the normalisation layer makes explicit what has been understood by the transcriber and thus can be considered as a *poor man's target hypothesis* where decisions made during the transcription become more transparent (also see Hirschmann et al. (2007), Reznicek et al. (2010) for a discussion of the importance of target hypotheses for the analysis of learner language).

Figure 1 shows an example transcript from KiDKo in the transcription tool EXMARaLDA (Schmidt, 2012), displaying the transcription and the normalisation layer, the POS tags and a layer for non-verbal information. Uppercase letters on the transcription layer mark the main accent of the

[2]Please note that the normalisation does not transfer the data into canonical structures. We do not change nonstandard patterns, e.g., in such domains as inflection or word order. The normalised layer also includes disfluencies, repetitions, and abandoned utterances.

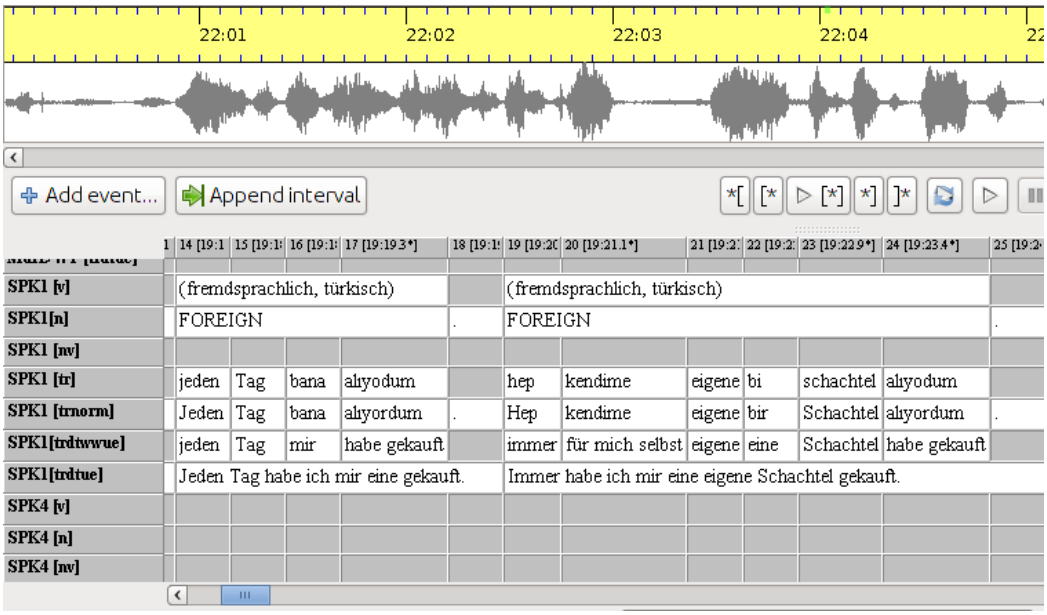

Figure 2: Screenshot KiDKo sample of German-Turkish code-mixing (English transliteration SPK1: Every day me have bought. *"Every day I bought one for myself."* Always for myself own one box have bough. *"I always bought my own box."*)

utterance. The equals sign is used to encode the tight continuation of a word form with a following form, where one of the two forms (or both of them) is reduced (e.g. such cliticisations as in 'warst =e' (warst du) "were you"). The (-) marks a silent pause of short length.

Since the data for the main corpus was recorded in conversations with a lot of multilingual speakers, it also includes some code-mixing and code-switching of German with other heritage languages, mostly Turkish. For those passages, the Turkish part has been transcribed and translated (Figure 2). On the (German) transcription and normalisation layer, the utterance has been marked as foreign language material. We provide a Turkish transcription layer (tr) that captures nonstandard pronunciation, but does not mark the main accent of the utterance. The Turkish normalisation layer (trnorm) translates this into Standard Turkish. In addition, we provide a literal German translation (trdtwwue) and a free translation (trdtue).

2.2. Corpus access and future work

We plan to release the POS tagged version of the corpus in spring 2014. Due to legal constraints, the audio files will have restricted access and can only be accessed locally while the transcribed and annotated version of the corpus will be available over the internet via ANNIS (Zeldes et al., 2009).[3]

In the near future, we will augment the corpus with a flat syntactic analysis and topological field information (Drach, 1937; Höhle, 1998). The new layers will enable users to conduct corpus searches for complex syntactic phenomena. In the remainder of the paper we focus on the challenges of automatic POS tagging of spoken language and report our efforts to improve the tagger and to identify error patterns in the automatically tagged data.

3. POS tagging

The procedure for adding a POS annotation layer to KiDKo is as follows. First, the data is transcribed. Then, we automatically add the normalisation layer by copying the transcriptions to a separate layer and automatically correcting spelling and frequent pronunciation variants based on a dictionary lookup. Then the normalisation is checked by the transcriber and remaining errors are corrected manually. Afterwards, the normalisation is automatically POS tagged, using a CRF-based tagger developed for the annotation of Kiezdeutsch (Rehbein and Schalowski, To appear)[4] and manually corrected in a post-processing phase.

The tagger is based on the CRFSuite package (Okazaki, 2007) and uses features like word form, word length, or the number of upper case letters or digits in a word. In addition, we use prefix/suffix features (the first/last n characters of the input word form) as well as feature templates which generate new features of word ngrams where the input word form is combined with preceding and following word forms. To address the unknown word problem in our data, we add features from LDA word clusters (Chrupała, 2011) learned on untagged Twitter data and an automatically created dictionary which was harvested from the Huge

[3] ANNIS (ANNotation of Information Structure) is a corpus search and visualisation interface which allows the user to formulate complex search queries which can combine multiple layers of annotation. (`http://www.sfb632.uni-potsdam.de/annis/`)

[4] The annotation scheme we use is an extended version of the Stuttgart-Tübingen Tagset (STTS) (Schiller et al., 1999) with 11 new tags tailored to the annotation of spoken discourse. Our annotators achieved an inter-annotator agreement of 0.975 (Fleiss' κ) on KiDKo data using the extended tagset.

German Corpus (HGC) (Fitschen, 2004) which had been POS tagged using the Treetagger (Schmid, 1995).

Our tagger achieves an accuracy of 95.8% on the normalised transcripts when trained on a small training set with 10,682 tokens, and of 96.9% when trained on a larger training set (66,043 tokens; 5-fold cross validation).

The accuracy of the tagger is in the same range as state-of-the-art taggers on newspaper text. However, the results might be a bit too optimistic as we also tag silent pauses and foreign as well as uninterpretable material, which are all unambiguous and occur with a high frequency in the corpus. To give a more realistic assessment of the tag quality, we exclude punctuation, silent pauses and uninterpretable/foreign language material from the evaluation and compare the KiDKo results to results achieved by our tagger when trained and tested on the TIGER treebank (Brants et al., 2002), using the data split from the CoNLL-2006 shared task.[5] Results show that the impact of silent pauses and foreign/uninterpretable material on tagging accuracy is quite low but that the large number of punctuation signs, owed to the shorter utterance lengths in the corpus, has a crucial influence on tagging accuracy. Removing punctuation results in a decrease in accuracy of 1.4% for KiDKo whereas the accuracy on TIGER only decreases from 98.3% to 98.0%.

Figure 3 shows the learning curve for our best tagger, the CRF tagger. In the beginning, the curve is quite steep up to a training size of around 50,000 tokens. After that, adding more training data does not have such a strong effect on accuracy any more.

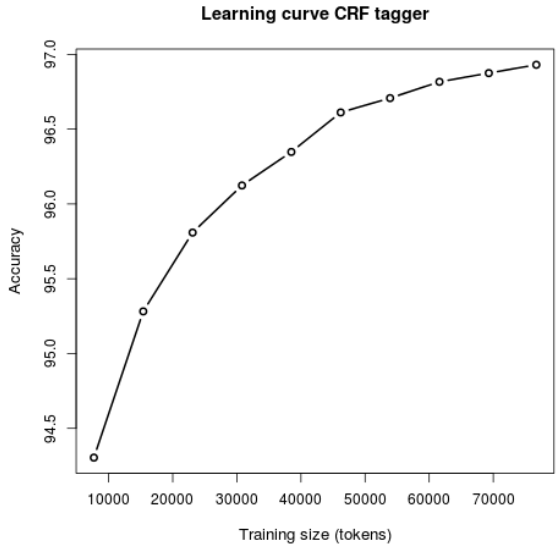

Figure 3: Learning curve for the CRF tagger (5-fold cross validation)

[5]In the experiments we also use LDA word clusters from Twitter. Replacing those by word clusters learned from the HGC gives a small improvement of around 0.1%

Baseline taggers	with punc	w/o punc
Brill	94.4	91.8
Treetagger	95.1	92.8
Stanford	95.3	93.5
Hunpos	95.6	93.6
CRF	**96.9**	**95.5**
majority vote	96.4	94.8
stacking (brill, crf, hun, stan, tree)	96.8	95.4
stacking (brill, hun, stan, tree)	96.8	95.3
stacking (hun, stan, tree)	96.8	95.4
stacking (hun, stan)	96.8	95.4

Table 1: Baseline and ensemble results for different taggers and tagger combinations, using majority vote and stacking a CRF tagger with the output of the baseline taggers (5-fold cross validation on the training set; second column shows results excluding punctuation)

3.1. Ensemble tagging

It has often been shown that combining different taggers and either using a simple majority vote or stacking a tagger with POS tags predicted by other taggers does improve tagging results (Brill and Wu, 1998; Màrquez et al., 1999; Søgaard, 2010).

Thus, we tried to improve tagging accuracy by combining the output of five different taggers. The taggers used in our experiments are

- the Brill tagger (Brill, 1992)
- the Stanford tagger (Toutanova and Manning, 2000)
- the Hunpos tagger[6]
- the Treetagger (Schmid, 1995)
- our CRF-based tagger[7]

We tried two different approaches. In the first one, we used a simple majority vote. In the second approach, we trained a new CRF-based classifier, using the output of the five different taggers as additional features. The results are shown in Table 1.

Surprisingly, we were not able to improve over our best baseline tagger (CRF: 96.9%). Results for classifier stacking are a bit higher than for the simple majority vote, but still below the results of the CRF tagger. We suspect that the gap in accuracy between our best tagger and the other systems is too large so that the highest-scoring system could not benefit from the output of the other taggers.

[6]The Hunpos tagger is an open source reimplementation of the TnT tagger (`https://code.google.com/p/hunpos`)

[7]`http://www.chokkan.org/software/crfsuite/`

	ALL	w/o	NE	PRF	PTKZU	PTKVZ	VAINF	VVFIN	VVIMP
CRF	96.9	95.5	89.8	71.3	84.8	93.5	77.3	94.5	89.0
2STEP	97.2	96.0	92.3	82.6	88.2	100.0	89.6	95.2	90.7

Table 2: Improvments for all tags (with and without punctuation) and for individual POS tags (NE: proper name, PRF: reflexive pronoun, PTKZU: infinitive with *zu*, PTKVZ: seperated verb particle, VAINF: infinite auxiliary, VVFIN: finite full verb, VVIMP: imperative full verb

3.2. Improved tagging with linguistically motivated features

Our error analysis shows that the tagger often mistakes proper names for nouns and vice versa. Other frequent errors are the confusion of finite and infinite verbs, of personal and reflexive pronouns, and of demonstrative pronouns and determiners.[8]

Most of this is not really surprising, as the distinction between nouns and proper names is also problematic on a theoretical level, and some of the decisions made in the annotation guidelines seem to be arbitrary (see Schiller et al. (1999), pp. 15.) Discriminating between finite and infinite verbs, however, is easy for human annotators even in cases where surface forms are identical, as, e.g., for some verb forms inflected for 2PL and infinitives. In order to accomplish the task, the tagger needs more global context. Example 3 illustrates this. In 3 a) the finite, plural form *machen* (do) should be assigned the `VVFIN` tag, while in b) and c) *machen* is infinite and should be tagged as `VVINF`.

(3) a. weil sie Hausaufgaben **machen** .
because they homework do$_{2.pl}$.
"because they are doing their homework."

b. weil sie Hauaufgaben **machen** muss .
because she homework do$_{infinite}$ must .
"because she has to do her homework."

c. weil sie Hauaufgaben **machen** nicht mag .
because she homework do$_{infinite}$ not likes .
"because she doesn't like to do homework."

The left context in these three examples is exactly the same. The only clue is the modal verb in the right context in b) (*muss*) and c) (*mag*). While in b) the direct adjacency of the two word forms enables the tagger to use this information, in c) the modal verb is out of range, resulting in the false prediction for *machen* as a finite verb.[9]

Due to the semi-free word order in German, we often observe cases like the one above where global information is not locally accessible. We thus use a two-step approach where in the first step we assign POS tags to the text, using our best baseline system. In the second step we extract new features from the output of the first tagger and train a second classifier, adding linguistically motivated clues from the left and right context.

3.2.1. Finite vs. infinite verbs

To better distinguish between finite and infinite verb forms, we search in the right context of each token for a verb, starting from the end of the sentence. If we find one, we add the POS for this verb predicted by the CRF tagger as a new feature.[10]. This feature is added for each token.

The left context feature is only added for tokens that have been identified as a verb form in the first step. For all other tokens, this feature is set to *null*. Starting from the token we want to tag, we search the left context for either another verb form or for a subordinating conjunction, a relative or interrogative pronoun. If we find one, we add the tag as a new feature.

3.2.2. Personal vs. reflexive pronouns

To help the tagger making a more informed decision on identifying reflexive pronouns, we add a new feature for each of the following word forms: *dich, dir, euch, mich, mir, sich, uns*. These forms are ambiguous between a reflexive and an irreflexive reading. We thus search the clausal context for another pronoun agreeing in person with the first form.

(4) a. Ich habe mich geschnitten .
I have myself$_{reflexive}$ cut .
"I have cut myself."

b. Sie hat mich geküsst .
she has myself$_{irreflexive}$ kissed .
"She has kissed me."

c. Hab mich geschnitten .
have myself$_{reflexive}$ cut .
"Have cut myself."

In 4 a) we would find the pronoun *ich* (I) which is in agreement with *mich* (myself). We thus add a new feature `RFLX`. In 4 b), the pronoun *sie* (she) does not agree with *mich* and thus the feature value is set to *null*. Our new feature does not fire in elliptical contexts (4 c) where the relevant information is not present in the surface structure. To capture these cases, we would need a morphological analysis of the verb. However, the accuracy of morphological tools on informal spoken language is not as good as on Standard German text. We thus did not follow up on this approach but left it to future work.

[8]These errors are not typical for informal, spoken youth language, but also occur when tagging newspaper text.

[9]It is, of course, possible to train tagging models utilising a larger context window. This, however, usually results in sparse data problems and thus in a lower accuracy.

[10]The STTS distinguishes 12 verb tags: V(V|A|M)INF (full/auxiliary/modal infinite verbs), V(V|A|M)FIN (full/auxiliary/modal finite verbs), V(V|A|M)PP (full/auxiliary/modal past participles), (V|A)IMP (full/auxiliary imperatives), VVIZU (infinitive with *zu*)

3.2.3. Nouns vs. proper names

To see if we can further improve the accuracy for nouns and proper names, we also add features extracted from Brown clusters learned on unannotated data from Twitter.

3.2.4. Results

Table 2 gives results for the two-step approach. We observed a modest improvement of 0.3% (0.5% when excluding punctuation) over our best baseline system. While these numbers do not seem very impressive, the detailed results for individual POS tags (Table 2) show that our new features did increase accuracy for reflexive pronouns by more than 11%. For infinite auxiliaries, the increase is also substantial with more than 12%. In addition, we observe a small, but positive effect on most verb tags and also on the identification of seperated verb particles. The Brown cluster features improved POS accuracy for proper names by 2.5%, showing that the hierarchical clustering adds complementary information not already captured by the LDA cluster features.

4. Error detection

Our POS accuracy, now in the range of 96-97%, is quite good, considering that we are dealing with a non-canonical variety of spoken language. However, as our goal is to build a new resource for linguistic research, the remaining error rate of 3-4% is still too high. Unfortunately, we do not have the funds necessary for doing a complete manual correction of the whole corpus, least of all for double annotation. We thus have to find efficient ways to identify errors in the tagger output and to correct these.

In this section, we describe our approach to automatic error detection where we use the predictions of the different ensemble taggers (Section 3.1.) to identify tagging errors.

4.1. Related work

Most work on (semi-)automatic POS error detection has focussed on identifying errors in POS assigned by *human annotators* where variation in word-POS assignments in the corpus can be caused either by ambiguous word forms which, depending on the context, can belong to different word classes, or by erroneous annotator decisions (Eskin, 2000; van Halteren, 2000; Květoň and Oliva, 2002; Dickinson and Meurers, 2003; Loftsson, 2009).

The *variation n-gram algorithm* (Dickinson and Meurers, 2003) allows users to identify potentially incorrect tagger predictions by looking at the variation in the assignment of POS tags to a particular word ngram. The algorithm produces a ranked list of varying tagger decisions that have to be processed by a human annotator. Potentional tagger errors are positioned at the top of the list. Later work (Dickinson, 2006) extends this approach and explores the possiblities of automatic correction of the detected errors.

Eskin (2000) describes a method for error identification using *anomaly detection*. Anomalies in this approach are defined as elements coming from a different distribution than the one in the data at hand.

Květoň and Oliva (2002) present an approach to error detection based on a semi-automatically compiled list of *impossible ngrams*. Instances of these ngrams in the data are assumend to be tagging errors.

	tokens	candidates	true err.	out of	(% err)
train	66,024	4,120	986	1,840	53.6
dev	16,530	1,228	267	437	61.1
test	20,472	1,797	558	788	70.8

Table 3: Number of error candidates identified by disagreements in the ensemble tagger predictions

Loftsson (2009) evaluates different methods for error detection, using the method of Dickinson and Meurers (2003) as well as an ensemble of five POS taggers, showing that both approaches allow for the successful identification of POS errors and increase tagging accuracy.

All these approaches are tailored towards identifying human annotation errors and cannot be applied to our setting, where we have to detect systematic errors made by automatic POS taggers. Thus, we can not rely on *anomalies* or *impossible ngrams* in the data, as the errors made by the taggers are consistent and, furthermore, our corpus of non-canonical spoken language includes many structures which are considered *impossible* in Standard German.

Rocio et al. (2007) address the problem of finding systematic errors in POS tagger predictions. Their method is based on a modified multiword unit extraction algorithm that extracts cohesive sequences of tags from the corpus. These sequences are then sorted manually into linguistically sound ngrams and potential errors. This approach hence focusses on correcting large, automatically annotated corpora. It successfully identifies (a small number of) incorrectly tagged high-frequency sequences in the text which are often based on tokenisation errors. The more diverse errors due to lexical ambiguity, which we have to deal with in our data, however, are not captured by this approach.

4.2. Using tagger ensembles for error detection

We follow Loftsson (2009) and use the predictions of the different ensemble taggers described above to identify POS errors in the corpus. We use the same training/development/test set split as described in Section 3. In the training data, our tagger ensembles agree on 61,904 out of 66,024 instances. For 4,120 tokens, the ensemble taggers' decisions diverge (Table 3). Out of these 4,120 instances, 986 were in fact errors, which gives us an error detection precision of 23.9%. For the development and test data, the precision is 21.7 and 33.0, respectively. This is somewhat higher than the precision of 16.6% reported by Loftsson (2009) for the Icelandic tagger ensemble, meaning that we have to look at a smaller number of instances to correct the same amount of errors in our data.

The ensemble tagger approach succeeds in detecting more than 50% of all errors in the data, with reasonable effort. After manually correcting those instances, the POS tag accuracy in the corpus increases up to 98.7% (development set) and to 99.0% on the test set.

5. Increasing POS accuracy to over 99%

To attain our goal of creating a high-quality annotated corpus, we follow a second approach to identifying POS errors

	tokens	candidates	true err.	out of	acc.
train	66,024	7,472	505	854	99.5
dev	16,530	2,022	108	213	99.4
test	20,472	2,104	66	207	99.3

Table 4: Correcting ambiguous word forms

in the tagger output. While our last approach relied on the judgements of automatic taggers, this time we make use of the manually annotated training data used to develop the taggers.

From the training data, we extract word forms where our best tagger frequently made mistakes. We use a threshold of 5, meaning that we extract all word forms that have been assigned an incorrect POS tag at least 5 times in the training data. This threshold can, of course, be adjusted according to the quality requirements and resources available for manual correction.

Setting the threshold to 5, we extract a list of 72 different word forms from the training data. As we already corrected those instances where the different taggers disagreed in their judgements, we now only have to look at instances where all five taggers predicted the same tag. This gives us 7,472 instances for the training set and a bit more than 2,000 instances for the development and test set (Table 4). This means that we have to manually check around 10% of all instances in the different sets, which can be done quite efficiently by providing the annotators with a tool that highlights these instances, sorted by word form.

Most of the instances are, in fact, correct. Only around 3-7% of these error candidates are real POS errors. However, after applying this simple heuristic, the overall POS accuracy in the corpus increases up to 99.5% (training set), 99.3% (development set) and 99.3% (test set).

These number are achievable if the annotators are well trained and always assign the correct POS tag. This assumption is, of course, overly optimistic. However, our inter-annotator agreement of 0.975 (Fleiss' κ) for three human annotators on a subset of the corpus showed that POS annotation on such informal spoken language can be done with good reliability, and thus potential annotator errors are not expected to have a crucial impact on the final POS accuracy in the corpus.

6. Conclusions and Future Work

We presented KiDKo, a new, POS annotated corpus for investigations of informal youth language and of language variation in monolingual and multilingual urban settings. Release 1.0 of the corpus includes the transcriptions, a normalisation layer and POS annotations, as well as the transcription and translation of Turkish language material from code-mixing and -switching. The corpus will be made freely available for research purposes.

In future work, we will augment the corpus with a shallow syntactic analysis and topological field information.

7. Acknowledgements

This work was supported by a grant from the German Research Association (DFG) awarded to SFB 632 "Information Structure" of Universität Potsdam, Humboldt Universität Berlin and Freie Universität Berlin, Project B6: "The Kiezdeutsch Korpus (KiDKo)". We acknowledge the work of our transcribers and annotators, Anne Junghans, Banu Hueck, Charlotte Pauli, Emiel Visser, Franziska Rohland, Jana Kiolbassa, Julia Kostka, Marlen Leisner, Nadine Lestmann, Nadja Reinhold, Oli Bunk, and Sophie Hamm. Additional researchers involved in the first phase of the project were Ulrike Freywald, Tiner Özçelik, and Katharina Mayr, who contributed to gathering the linguistic material, compiling the corpus data, and organising first transcriptions. We would also like to thank the anonymous reviewers for helpful comments.

8. References

Auer, P. (2013). Ethnische Marker im Deutschen zwischen Varietät und Stil. In Deppermann, A., editor, *Das Deutsch der Migranten [IDS Yearbook 2012]*, pages 9–40. Berlin, New York: de Gruyter.

Brants, S., Dipper, S., Hansen, S., Lezius, W., and Smith, G. (2002). The TIGER treebank. In *Proceedings of the First Workshop on Treebanks and Linguistic Theories*, pages 24–42.

Brill, E. and Wu, J. (1998). Classifier combination for improved lexical disambiguation. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics - Volume 1*, ACL '98.

Brill, E. (1992). A simple rule-based part of speech tagger. In *3rd conference on Applied natural language processing (ANLC'92)*, Trento, Italy.

Chrupała, G. (2011). Efficient induction of probabilistic word classes with LDA. In *Proceedings of 5th International Joint Conference on Natural Language Processing*, pages 363–372, Chiang Mai, Thailand, November. Asian Federation of Natural Language Processing.

Dickinson, M. and Meurers, D. W. (2003). Detecting errors in part-of-speech annotation. In *10th Conference of the European Chapter of the Association for Computational Linguistics (EACL-03)*.

Dickinson, M. (2006). From detecting errors to automatically correcting them. In *Annual Meeting of The European Chapter of The Association of Computational Linguistics (EACL-06)*, Trento, Italy.

Drach, E. (1937). Grundgedanken der Deutschen Satzlehre.

Eskin, E. (2000). Automatic corpus correction with anomaly detection. In *1st Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, Seattle, Washington.

Fitschen, A. (2004). *Ein computerlinguistisches Lexikon als komplexes System*. Ph.D. thesis, Institut für Maschinelle Sprachverarbeitung der Universität Stuttgart.

Hirschmann, H., Doolittle, S., and Lüdeling, A. (2007). Syntactic annotation of non-canonical linguistic structures. In *Proceedings of Corpus Linguistics 2007*, Birmingham, UK.

Höhle, T. (1998). Der Begriff "Mittelfeld", Anmerkungen über die Theorie der topologischen Felder. In *Ak-*

ten des Siebten Internationalen Germanistenkongresses 1985, pages 329–340, Göttingen, Germany.

Květoň, P. and Oliva, K. (2002). (Semi-)Automatic detection of errors in PoS-tagged corpora. In *19th International Conference on Computational Linguistics (COLING-02)*.

Loftsson, H. (2009). Correcting a POS-tagged corpus using three complementary methods. In *Proceedings of the 12th Conference of the European Chapter of the ACL (EACL 2009)*, Athens, Greece, March.

Màrquez, L., Rodrfguez, H., Carmona, J., and Montolio, J. (1999). Improving pos tagging using machine-learning techniques. In *In Proceedings of the 1999 Joint SIGDAT Conference on empirical Methods in Natural Language Processing and very large corpora*, pages 53–62.

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

Rehbein, I. and Schalowski, S. (To appear). STTS goes Kiez – Experiments on annotating and tagging urban youth language. *Journal for Language Technology and Computational Linguistics*.

Reznicek, M., Walter, M., Schmidt, K., Lüdeling, A., Hirschmann, H., Krummes, C., and Andreas, T., (2010). *Das Falko-Handbuch: Korpusaufbau und Annotationen*. Institut für deutsche Sprache und Linguistik, Humboldt-Universität zu Berlin, Berlin.

Rocio, V., Silva, J., and Lopes, G. (2007). Detection of strange and wrong automatic part-of-speech tagging. In *Proceedings of the Aritficial Intelligence 13th Portuguese Conference on Progress in Artificial Intelligence*, EPIA'07.

Schiller, A., Teufel, S., and Thielen, C. (1999). Guidelines für das Tagging deutscher Textkorpora mit STTS. Technical report, Universität Stuttgart, Universität Tübingen.

Schmid, H. (1995). Improvements in part-of-speech tagging with an application to German. In *ACL SIGDAT-Workshop*.

Schmidt, T. (2012). EXMARaLDA and the FOLK tools. In *The 8th International Conference on Language Resources and Evaluation (LREC-12)*, Istanbul, Turkey.

Selting, M., Auer, P., Barden, B., Bergmann, J., Couper-Kuhlen, E., Günthner, S., Quasthoff, U., Meier, C., Schlobinski, P., and Uhmannet, S. (1998). Gesprächsanalytisches Transkriptionssystem (GAT). *Linguistische Berichte*, 173:91–122.

Søgaard, A. (2010). Simple semi-supervised training of part-of-speech taggers. In *Proceedings of the ACL 2010 Conference Short Papers*, ACLShort '10, pages 205–208, Stroudsburg, PA, USA. Association for Computational Linguistics.

Toutanova, K. and Manning, C. D. (2000). Enriching the knowledge sources used in a maximum entropy part-of-speech tagger. In *Proceedings of the conference on Empirical methods in natural language processing and very large corpora*, EMNLP '00, Hong Kong.

van Halteren, H. (2000). The detection of inconsistency in manually tagged text. In *Proceedings of the COLING-2000 Workshop on Linguistically Interpreted Corpora*, Centre Universitaire, Luxembourg, August.

Wiese, H., Freywald, U., Schalowski, S., and Mayr, K. (2012). Das KiezDeutsch-Korpus. Spontansprachliche Daten Jugendlicher aus urbanen Wohngebieten. *Deutsche Sprache*, 2(40):797–123.

Wiese, H. (2009). Grammatical innovation in multiethnic urban Europe: New linguistic practices among adolescents. *Lingua*, 119:782–806.

Wiese, H. (2013). What can new urban dialects tell us about internal language dynamics? The power of language diversity. In Abraham, W. and Leiss, E., editors, *Dialektologie in neuem Gewand. Zu Mikro-/Varietätenlinguistik, Sprachenvergleich und Universalgrammatik*, number 19, pages 207–245.

Zeldes, A., Ritz, J., Lüdeling, A., and Chiarcos, C. (2009). Annis: A search tool for multi-layer annotated corpora. In *Corpus Linguistics 2009*.