Adding Manual Constraints and Lexical Look-up to a Brill-Tagger for German

Gerold Schneider and Martin Volk
University of Zurich
Department of Computer Science
Computational Linguistics Group
Winterthurerstr. 190, CH-8057 Zurich
gschneid|volk@ifi.unizh.ch

Abstract

We have trained the rule-based Brill-Tagger for German. In this paper we show how the tagging performance improves with increasing corpus size. Training over a corpus of only 28'500 words results in an error rate of around 5% for unseen text. In addition we demonstrate that the error rate can be reduced by looking up unknown words in an external lexicon, and by manually adding rules to the rule set that has been learned by the tagger. We thus obtain an error rate of 2.79% for the reference corpus to which the manual rules were tuned. For a second general reference corpus lexical-lookup and manual rules lead to an error rate of 4.13%.

1 Introduction

There already exist a number of taggers for German (Lezius et al., 1996). We have noticed, however, that none of them is rule-based. But as Samuelsson and Voutilainen (1997) have demonstrated rule-based taggers can be superior to statistical taggers. We have therefore adapted and trained the supervised version of the rule-based Brill-Tagger to German. To this end we have been building up a German training corpus, which currently consists of about 38'000 tagged words, where all tags have been manually checked.[1] In this paper we show how the tagging performance improves with increasing corpus size. In addition we demonstrate that the error rate can be further reduced by looking up unknown words in an external lexicon, and by manually adding rules to the rule set that has been learned by the tagger.[2]

1.1 Rule-Based Tagging

We have chosen the Brill-Tagger for the following reasons:

Practical Performance The rule-based Brill-Tagger (Brill, 1992, Brill, 1994) has shown good results for English. Samuelsson and Voutilainen (1997) show that a rule-based tagger for English can achieve better results than a stochastic one. Chanod and Tapanainen (1995) prove the same for French.

Theoretical Advantages While the constraints for French by Chanod and Tapanainen (1995) and for English by Samuelsson and Voutilainen (1997) are hand-written, the Brill-Tagger is self-learning. It employs a transformation-based error-driven learning method. Ramshaw and Marcus (1996) describe this as a compromise method, which means that it involves both a statistical and a symbolic component. Instead of pure n-grams the Brill-Tagger uses rule templates to restrict the search space.

Linguistic Accessibility and Extensibility Another advantage of rule-based tagging over statistical approaches is the linguistic control, as Ramshaw and Marcus (1996) point out. Linguistic knowledge first defines the linguistic principles to be statistically investigated, i.e. the Brill-Tagger set of rule templates. Second, they allow to tune an automatically abstracted

[1] We would like to acknowledge the help of Gero Bassenge, Alexander Glintschert, Sven Hartrumpf, Sebastian Hübner, Sandra Kübler, Andreas Mertens and Elke Teich in checking part of our training corpus.

[2] The web version of our tagger and information about its availability can be found at http://www.ifi.unizh.ch/CL/tagger.

description of the training corpus, i.e. the rule files. Third, they help in analysing the results and in pin-pointing the remaining errors.

1.2 The Brill-Tagger for German

The Brill-Tagger was originally developped for English.[3] For German, we had to start from scratch, first finding a suitable tag-set, adapting the tagger code slightly, and then manually tagging a German corpus. The changes in the tagger code needed for German are well documented in the tagger manuals. It is necessary to adapt the initial guess for capitalized words, which for English is a tag for proper noun in the original code. This had to be changed to the tag for common noun in our tag-set, because in German all nouns are capitalized.

1.2.1 Training Phase

The Brill-Tagger is trained in two steps. In the first step, each word is assigned its most likely tag, based on the training corpus. In the second step, the errors made in step one are recorded, and the tagger finds rules for the biggest possible error elimination based on the context or internal build-up of words. The tagger formulates these rules on the basis of the rule templates. Every rule is then tested against the training corpus, the number of corrected errors are weighed against the number of errors newly introduced by this rule. The rule with the greatest net improvement is included in the rule set. This learning procedure continues iteratively, until a certain threshold is reached. Due to its iterative character, newly acquired rules are already respected for the elimination of the next error.

Using this procedure, the Brill-Tagger generates a lexicon and two rule files, one for context rules and one for lexical rules.

1.2.2 Application Phase

For the application of the tagger to a text, the tagger uses the files generated during the training. The lexicon contains each word with all its tags as they occurred in the training corpus. The tag at the first position is the most likely tag, which will be assigned to a word as a first guess. These guesses are then corrected according to the learned context rules.

Context rules take the context of a word into consideration. The Brill-Tagger has an observation window of size 4: the furthest reaching rule template allows for the consideration of three words to the left or to the right. This is bigger than in most statistical taggers. We will give examples of context rules in section 4.1.

The other rule file contains lexical rules. Lexical rules are solely used for tagging unknown words. The following is an example of a simplified lexical rule

```
lich hassuf 4 ADV
```

This lexical rule will have the effect that unknown words with a four-letter suffix `-lich` are transformed into adverbs from whatever their first guessed tag was.

Brill rules are transformation rules, which means that a tag is transformed into another tag if a rule applies. But at any stage a word will have exactly one tag. In this sense, transformation rules (Ramshaw and Marcus, 1996) are different from constraint rules (Samuelsson and Voutilainen, 1997).

1.3 Tag-set and Corpora

We use a tag-set widely acknowledged for German, the so-called Stuttgart-Tübingen Tag-Set (STTS) (Schiller et al., 1995), which contains 51 part-of-speech tags plus some punctuation tags. Our corpus consists of texts from the University of Zurich annual report and currently contains about 38'000 words.

2 Performance of the Brill-Tagger for German

In this chapter we show how the tagging accuracy increases with increasing corpus size, until the progress flattens out, partly due to the tagging difficulties for German, which we will describe below. In separate experiments, which are documented in (Volk and Schneider, 1998), we show that the Brill-Tagger and the statistical tagger by Schmid (1995) achieve similar results for German.

2.1 Training the Brill-Tagger with our Corpus

We used a utility provided by Brill to split the 38'000 word corpus into two halves, say A and

[3] The Brill-Tagger is available from its author at `http://www.cs.jhu.edu/~brill`.

B. This utility repeatedly takes the next two sentences from the corpus and randomly puts one sentence to file A and the other one to file B. We divide these halves again by the same method to get four parts. We use the first three parts as the training corpus, which we call TC. The remaining quarter is reference material. We divide this reference material again in the same way. We call the reference corpora we thus get RC1 and RC2.

2.1.1 Training Progress

In order to illustrate the training progress, we first train the tagger with only 12.5% of the corpus and tag RC1 with the data obtained from the training. Then, we move on to 25%, 50% and 75% of the corpus and tag RC1 again. Table 1 illustrates the training progress. The 75% corpus is TC, i.e. the training corpus we will use for the rest of this paper. After training the Brill-Tagger with the training corpus TC, the training module has learnt 186 lexical rules and 176 context rules. When applying these rules to RC1 the error rate is at 5.04%. This means that 94.96% of all the tokens in RC1 receive the same tag as manually prespecified.

For illustration purposes, we also add RC2 to TC and tag RC1 again, reducing the error rate to 4.81%. As expected, the error rate in table 1 flattens out, suggesting that the increase in tagging accuracy from using bigger training corpora will become increasingly smaller.

Size of the training corpus	Error rate for RC1	Avg. Ambig. per Token for RC1
12.5%	11.96%	1.209
25.0%	9.01%	1.274
50.0%	6.33%	1.307
TC = 75.0%	5.04%	1.309
TC+RC2 = 87.5%	4.81%	1.373

Table 1: Error Rates on RC 1

Of course, we may also tag RC2 with TC. Coincidentally, the error rate is a little higher than with reference corpus RC1, at 5.59%.

When we add the reference material to the training corpus, the error rate drops significantly, as table 2 illustrates. Of course this error rate of 1.49% has no "real world" significance, because no matter how big a training corpus there will always be new words and new syntactic constructions in a new text to be tagged.

Size of training corpus	Error rate for RC1
TC+RC2+RC1 = 100%	1.49%

Table 2: Drastic Error Rate Reduction on Including Reference Material into Training Material

2.1.2 Unknown Words

But we found this increase so striking that we wanted to know if it is rather due to the known vocabulary or due to the larger number of transformation rules. When tagging RC1 with the rule files learned from TC, but the lexicon learned from the entire corpus (i.e. TC+RC1+RC2) the error rate was only 1.86%, almost as good as when tagging with the rule files learned from the entire corpus. On the contrary, when tagging with the rule files learned from the entire corpus but using only the lexicon learned from TC, the error rate was 4.86% (even a little worse than when using the TC rules). This indicates that the Brill part-of-speech guesser for unknown words is still unsatisfactory (Brill, 1994). Section 3 describes one way to increase the tagging accuracy for unknown words.

2.1.3 Average Ambiguity per Token

Success and error rates alone are not enough as a measure for the efficiency of a tagger. If few words in the text to be tagged are ambiguous in the tagger lexicon, it is easy for the tagger to achieve good results. We therefore provide a figure on every training step for the average ambiguity per token. This figure is calculated for all tokens in a text (RC1 in our case) that are contained in the tagger lexicon. Unknown words are not used in this calculation. Enlarging the training corpus has two effects on the tagger lexicon. First, there will be more tokens in the lexicon and second, many tokens are assigned multiple tags. This accounts for the increase in the average ambiguity of tokens listed in the third column of table 1.

2.2 Tagging Difficulties for German

Adjective vs. Past Participle: In German the distinction between predicative adjec-

tives and past particles is difficult for linguistic experts and therefore also for the tagger. E.g. in the following corpus sentence

(1) ... während die technikbezogenen Disziplinen an der ETH Zürich vertreten sind.

it is difficult to judge whether *vertreten* is a past participle or an independent adjective. The sentence can be transformed into a similar active sentence, but it is debatable whether this involves a semantic change. In quantitative terms, however, only 3% of the errors from tagging RC1 are mistakes of this type.

Verb-Forms: The STTS tag set calls for a distinction of finite verb form, infinitive form and past participle form. But in German the finite verb form for the first and third person plural, present tense, is identical with the infinitive form. In addition there are many verbs where the past participle is identical with the infinitive or with a finite verb form. That means that one can decide on the verb form only by looking at the complete verb group in a clause.

But in German matrix clauses the verb group is a discontinuous constituent with the finite verb in second and the rest of the verb group in clause final position. This means that the distance between a finite auxiliary verb and the rest of the verb group can easily become too big for the window of a tri-gram tagger, as (Schmid, 1995) notes. Unfortunately, the Brill-Tagger window in many cases is not big enough either. In the following examples, our tagger mis-tagged `beantragt` as finite verb, while `verlangt` is mis-tagged as past participle.

(2) Hier hat der Ausschuss ... für die ersten beiden Punkte der Erziehungsdirektion beantragt, ...

(3) Die Theologische Fakultät verlangt Kenntnisse in Latein, Griechisch und Hebräisch.

When analysing the remaining 5.04% errors from tagging RC1 with our TC we find that indeed 25% of the errors involve a wrong verb form.

Capitalisation: Unlike in English, all nouns are capitalised in German. This means that the tagger mis-tags many unknown proper names as common names, and that sentence-initial unknown words are also often mis-tagged as common nouns. When analysing the errors in RC1 we find that 17% of the errors involve capitalisation.

3 The Impact of Lexical Lookup

As shown in 2.1.2, unknown words account for a large portion of the errors. We therefore experimented with sending the words not present in the tagger lexicon to the wide-coverage morphological analyser Gertwol (Oy, 1994). An automated mapping procedure over the Gertwol output extracts all possible STTS tags for a given word and temporarily appends these new words with their tags to the tagger lexicon. There is an obvious increase in the tagging accuracy, as table 3 shows.

Training Corpus	Lexical look-up	Error rate for RC1	Error rate for RC2
TC	no	5.04%	5.59%
TC	yes	4.33%	4.74%

Table 3: The impact of lexical lookup

But the impact of this lexicon-lookup is smaller than expected. The problem is that Gertwol delivers an unordered list of tags for a given word-form, which includes even rare readings. The Brill-Tagger, on the other hand, needs the most likely part-of-speech at the first position in the lexicon. We have not yet found out a method to weigh the Gertwol output appropriately.

4 The Impact of Manual Constraints

As stated in 1.1 the Brill-Tagger has the advantage that it finds linguistic rules from the training corpus which can be inspected, assessed and extended.

The net of automatically learned and partly interdependent rules might be a fragile system however, lenient to decrease in efficiency after manual editing. Since the rules depend on each other, their position within the rule file is relevant. How can a linguist know at which place

he or she should insert rules? And to what an extent are the rules interdependent?

Ramshaw and Marcus (1996) have investigated this question. They state [p. 151-2]:

> The trees for a run on 50K words of the Brown Corpus bear out that rule dependencies, at least in the part-of-speech tagging application, are limited. ... [T]he great majority of the learning in this case came from templates that applied in one step directly to the baseline tags, with leveraging being involved in only about 12% of the changes. The relatively small amount of interaction found between the rules also suggests that the order in which the rules are applied may not be a major factor in the success of the method for this particular application, and initial experiments tend to bear this out.

Given this reassurance, we added rules manually to the end of the contextual rule file.

4.1 Examples of Context Rules

As the Brill-Tagger has only little built-in linguistic knowledge it is on the one hand almost language-independent, on the other hand it has to rely on statistical data for learning linguistic rules.

It is striking to see that the learning algorithm automatically learns many well-known and linguistically sound context rules like:

```
APPR PTKVZ NEXTTAG $.
```

This rule means that what is initially tagged as a preposition (`APPR`) should be transformed into a separated verb prefix tag (`PTKVZ`) if found at the end of a sentence (`NEXTTAG $.`) - if the word in question can be found as `PTKVZ` in the tagger lexicon. Prepositions never occur at the end of a sentence indeed.[4] In the sentence

(4) Ich gebe nie auf.

`auf/APPR` is thus correctly transformed into `auf/PTKVZ`. More surprisingly, the learning algorithm even detects rules one is hardly aware of:

```
VAINF VAFIN NEXTTAG ADV
```

This rule transforms the infinite auxiliary verb tag into a finite auxiliary verb if followed by an adverb. Indeed, German word order seems to forbid sentences in which infinite auxiliaries are post-modified by an adverb:

(5) * Wir werden haben sehr schönes Wetter.

(6) * ... um zu sein ganz sicher.

The learning algorithm also detects a number of linguistically more questionable context rules which, however, correctly work in the majority of language uses, but which may also lead to mistakes:

```
VVFIN VVPP NEXT1OR2TAG $.
```

This rule means that sentence-final finite verb tags should be transformed into verb participles. While this rule is correct for e.g.

(7) Der Arzt hat seine Patienten behandelt.

it will produce wrong results for - in our corpus apparently less frequent - sentences like

(8) Der Arzt ist aufmerksam, wenn er seine Patienten behandelt.

Moreover, the learning algorithm misses some basic linguistic facts. In a reference corpus, we find e.g.

(9) unseres/PPOSS Reformvorhabens/NN

`PPOSS` stands for substituting possessive pronoun. The tag-set distinguishes between substituting and attributive pronouns. Without any linguistic knowledge, the tagger cannot know that a substituting pronoun will hardly be followed by a common noun (`NN`). A transformation to attributive personal pronoun seems plausible. This is where our **manual rules** come in. We add the following rule the contextual rule file:

```
PPOSS PPOSAT NEXTTAG NN
```

[4] Our tagset distinguishes prepositions and postpositions.

After training our tagger with TC we tag RC1 with lexical look-up (cf. section 3). The error rate is 4.33%. We manually checked the 189 errors and wrote and individually tested manual rules where we believe rules are linguistically plausible and expressable in the formalism - like the one for attributive personal pronouns above. We added the manual rules to the automatically learned rules. With 97 manual rules the error rate drops to 2.79%. Coming up with and testing new rules is a time-consuming process. Determining these 97 manual rules took us around 4 hours.

4.2 Results with Manual Constraints

The fact that rule interdependence is low (cf.section 4) suggests that we can safely add rules. On the other hand this may also indicate that rules are so independent because each of them only has a limited effect. In order to answer this question, we tag RC2, first only with lexical look-up (as described in 3) and with the automatically learned context rules from TC - we get an error rate of 4.74% -, then together with the above 97 manual context rules, for which we get an error rate of 4.13%.

Because of the small interaction, we may freely add manual rules written at other occasions. At an earlier stage of our research, when our training corpus comprised of 28'000 words, we wrote 141 manual rules for test purposes (cf. 4.3). We now add these manual rules to the file containing the automatically learned rules and the manual rules tuned for RC1. If we use this rule file for RC2 (again with lexical look-up), we get another increase in accuracy to 4.09%.

One may fear that manual rules are corpus-specific and bear no general linguistic siginificance, which would entail that they lead to an error increase in other corpora. In order to test this, we used the above rule file (i.e. automatically learned rules plus manual rules tuned for RC1 plus manual rules from earlier stage) to tag RC1 (with lexical look-up). We get only a slight error increase from 2.79% to 2.86%. This indicates that only a small fraction of the tuned rules are indeed corpus-specific, while the majority are linguistically accurate, at least in the sense of linguistic performance.

We conclude that because the context rules of the Brill-Tagger are independent and each has only a limited effect, the knowledge can be freely accumulated and will lead to better results in most cases. Adding manual rules is thus a feasible and useful practice for Brill tagging.

4.3 The Limits of Tagging Performance

As mentioned above, at an earlier stage of our research, when the entire corpus comprised of about 28'000 words, we wrote context rules based on the results of tagging the training corpus itself.

When a tagger tags its training corpus the error rate is naturally much lower than in tagging a new text. In the case of our 28'000 word corpus the error rate was at 1.81%. Based on these errors we wrote 141 manual context rules and added them to the 121 automatically learned context rules. The resulting error rate was just below 1%, at 0.95%. For the remaining 266 errors, no context rule could be found that resulted in any improvements. We therefore suggest that, given the window of the context rules in the Brill formalism and the restricted expressivity of the rules, and given the relatively free word order of the German language and the STTS tag set, an error rate of just slightly below 1% is about the best possible rate that can be achieved by a Brill-Tagger for German.

5 Conclusions

We have shown that the rule-based Brill-Tagger can be trained successfully over a relatively small annotated corpus. Tagging performance then suffers from unknown words but this can be alleviated by looking-up these words in an external lexicon. This lexicon should not only provide all possible tags but also identify the most likely tag. Current wide-coverage lexical resources like Gertwol do not contain this information. Perhaps a statistical analysis of online dictionaries, as proposed by Coughlin (1996), could help to compute this missing information.

Tagging performance can also be improved by adding manual rules to the automatically learned rule set. In our experiments a set of about 100 manual rules sufficed to increase the tagging accuracy from 95% to around 96%. We also demonstrated that the Brill-Tagger is relatively robust as to the order in which the manual rules are added. Unfortunately many of the remaining errors (e.g. verb form problems) lie outside the scope of the tagger's observation

window. Therefore we need to add a more powerful component to the tagger or build a shallow parsing post-processor for error correction.

References

Eric Brill. 1992. A simple rule-based part-of-speech tagger. In *Proceedings of ANLP*, pages 152–155, Trento/Italy. ACL.

Eric Brill. 1994. A report of recent progress in transformation-based error-driven learning. In *Proceedings of AAAI*.

Jean-Pierre Chanod and Pasi Tapanainen. 1995. Tagging French – comparing a statistical and a constraint-based method. In *Proceedings of EACL-95*, Dublin.

Deborah A. Coughlin. 1996. Deriving part of speech probabilities from a machine-readable dictionary. In *Proceedings of the Second International Conference on New Methods in Natural Language Processing*, pages 37–44, Ankara, Turkey.

W. Lezius, R. Rapp, and M. Wettler. 1996. A morphology-system and part-of-speech tagger for German. In D. Gibbon, editor, *Natural Language Processing and Speech Technology. Results of the 3rd KONVENS Conference (Bielefeld)*, pages 369–378, Berlin. Mouton de Gruyter.

Lingsoft Oy. 1994. Gertwol. Questionnaire for Morpholympics 1994. *LDV-Forum*, 11(1):17–29.

L.A. Ramshaw and M.P. Marcus. 1996. Exploring the nature of transformation-based learning. In J. Klavans and P. Resnik, editors, *The balancing act. Combining symbolic and statistical approaches to language*. MIT Press, Cambridge,MA.

C. Samuelsson and A. Voutilainen. 1997. Comparing a linguistic and a stochastic tagger. In *Proc. of ACL/EACL Joint Conference*, pages 246–253, Madrid.

A. Schiller, S. Teufel, and C. Thielen. 1995. Guidelines für das Tagging deutscher Textcorpora mit STTS (Draft). Technical report, Universität Stuttgart. Institut für maschinelle Sprachverarbeitung.

Helmut Schmid. 1995. Improvements in part-of-speech tagging with an application to German. Technical report, Universität Stuttgart. Institut für maschinelle Sprachverarbeitung. (Revised version of a paper presented at EACL SIGDAT, Dublin 1995).

Martin Volk and Gerold Schneider. 1998. Comparing a statistical and a rule-based tagger for German. Manuscript.