A Hybrid System for the Designation of Grammatical Case in German [1)].

Manfred Wettler & Christian Hellmann
Department of Psychology
University of Paderborn
Warburger Straße 100
D-33098 Paderborn
e-mail: wettler@psycho.uni-paderborn.de
fax: 0049-5251-603528
tel: 0049-5251-602900

1. Introduction

A necessary component of most applications in natural language processing is a tagger. A tagger is a program that determines the grammatical features of the input words. For the analysis of languages with rich morphologies, for example German, taggers are especially important, since information about number, gender, case, tense, and other grammatical features of words considerably reduces the burden of later syntactic analysis.

Most taggers use a machine readable dictionary which may be either a root lexicon or a full form lexicon. Root lexica contain the stems of the words. They have one entry for each lexeme combined with indications of its flexional paradigm (the rules for the derivation of the inflected form). Full form lexica have a separate entry for each inflected word form combined with all its tags, i.e. an exhaustive list of all combinations of grammatical features corresponding to the word form. Both kinds of lexica have the same capability. The decision about which kind of lexicon to use depends on practical considerations. Root lexica use less memory space whereas full form lexica reduce processing time.

The main problem of tagging is the disambiguation of ambiguous word forms, that more than one tag can be assigned to.

Alle	süßen	Kekse
pronoun indef. head nominative plural	mainverb infinitive	noun masculine dative singular
pronoun indef. head accusative plural	mainverb 3rd plural present	noun masculine nominative plural
pronoun indef. attributive definite accusative plural	adjective dative	noun masculine genitive plural
pronoun indef. attrivutive definite accusative plural	adjective nominative singular	noun masculine accusative plural
	adjective accusative singular masculine	
	adjective definite plural	

Table 1: Some of the tags for the three words *Alle süßen Kekse*.

Table 1, which illustrates these ambiguities, shows some of the tags for the three words *Alle süßen Kekse*. This table includes only grammatical features that are relevant for the derivation of the word form. For example, the uppermost tag of the indefinite pronoun *alle* says that *alle*, when used as the head of a noun phrase may be nominative plural in all three genders. The third tag of *süßen* indicates that it may be an adjective in the dative case accompanying singular or plural masculine, feminine, or neutral nouns in definite or indefinite noun phrases.

The main difficulty of taggers is finding the tag of an ambiguous word form that is contextually adequate. In order to solve this problem, two approaches have been used. In the more traditional symbolic approach the disambiguation of ambiguous word forms is accomplished by a syntactic parser. The parser begins with all of the different tags that might be assigned to the words of the input string in isolation. It then selects those sequences of tags giving syntactically correct constructions. In the example given in Table 1, the part-of-speech ambiguity of *süßen* might be solved with the rule that each sentence must have a finite verb.

In the probabilistic approach, the choice between different possible tags is made on the basis of the frequencies of tag sequences counted in previously tagged corpora. In this case, the interpretation of *süßen* as an adjective or as a verb would depend on the probability that an indefinite pronoun (*alle)* is followed by an adjective or by a verb, and the probability that a noun (*Kekse*) is preceded by a member of these two parts of speech.

The quality of taggers is usually measured by the percentage of words in unrestricted texts for which the tagger produces correct descriptions. The quality not only depends on the algorithms which the tagger uses and the size of the pretagged training corpus, but also on the language which is being tagged and on the richness of the grammatical descriptions produced by the tagger. Probabilistic taggers for unrestricted German texts which produce elaborate descriptions including information on number, case, gender etc. usually have hit rates which are between 80 and 90 percent. Comparable numbers for symbolic taggers are not available, but it is safe to assume that they do not perform any better. The above values are much too low to seriously consider applications of these programs outside the laboratory.

A qualitative analysis of the errors produced by the probabilistic German tagger MORPHY (Lezius, 1996; Lezius et al., 1996, 1998) has shown that many, if not the majority, of these errors result from erroneous assignments of grammatical cases. Occasionally, the tagger gets whole noun phrases wrong, frequently interpreting accusative objects as subjects and vice versa. Other times it assigns different cases to the constituents of the same noun phrase. This especially happens with noun phrases that are more than two words long. As the correct case identification of the grammatical case is an important and necessary step during the analysis of German sentences, such case errors pose a serious problem. One might expect that even a shallow and incomplete syntactical analysis of German sentences would be a considerable aid in overcoming this problem.

The aim of the following sections is to present just such a hybrid system. The system consists of three components. The first component is a morphological analyser which assigns all possible grammatical descriptions to German words independent of their context. The second component carries out a shallow syntactic analysis of all sentences without commas. It then removes all grammatical descriptions which do not conform to the syntactic constraints from the list produced by the first component. The third component consists of a probabilistic tagger.

The first and the third component have been taken over from MORPHY, an existing German tagger, which will be described in the following section. This description will make it possible to evaluate the improvement gained by using the syntactic component in comparison with the performance of a purely probabilistic tagger. The comparison between the two systems will be made in section four (Results).

2. **The probabilistic tagger MORPHY**

MORPHY is a probabilistic tagger which produces grammatical descriptions for simple and compound German words, including words not in the dictionary. The system is public domain and can be downloaded from the world wide web (http://www-psycho.uni-paderborn.de/lezius/). It can be used in two versions. One version uses a small feature set which specifies the parts of speech only. Another version uses a large feature set which includes information about case, number, gender, etc. Table 2 shows the main features for five parts of speech. The study presented here used the large feature set of the MORPHY program. The feature "role" has two values: "head", if the word is used as the head of a noun phrase, and "attribute", if not. The remaining features shown in Table 2 need no further explanations.

	# of values	Determiner	Adjective	Noun	Pronoun	Verb finite	Verb participle
case	4	+	+	+	+		+
number	2	+	+	+	+	+	+
gender	3	+	+	+	+		+
person	3				(+)	+	
definiteness	2	+	+		(+)		+
role	2				(+)		
tense	2					+	+
mood	3					+	
aux	2					+	+

Table 2: Grammatical features of different parts of speech. Plus signs indicate which features are assigned to the different parts of speech. Plus signs in brackets are used when the feature is assigned to certain subcategories only.

The program used in this study uses a root lexicon with 50,000 entries from which 324,000 different word forms can be derived. The lexicon does not include compound nouns. It includes all words of the German dictionary *Wahrig Deutsches Wörterbuch* (Wahrig,1997).
The morphological analysis of a word starts with the construction of a list of all possible roots by cutting off possible prefixes, infixes and endings, the reversion of vowel mutation if there are umlauts, and possible shifts between *ß* and *ss*. Compound nouns are segmented using a longest-matching rule which works from right to left and allows for possible gap vowels. A word form generator then checks if the word form being analysed could have been generated from the corresponding root (Lezius, 1996). This component produces a list of possible tags similar to the one shown in table 1. On average each word gets 5.4 tags.
The second part of MORPHY is a probabilistic tagger which uses the trigram algorithm described by Church (1988). This algorithms selects the tag with the highest probability for each word, i.e. the tag for which the product of its lexical probability and its contextual probability is highest. The lexical probability of a tag is the probability that this tag is the correct one for the corresponding word independent of its context. The contextual probability of a tag is the probability of observing this tag, taking into account the preceeding two tags. This contextual probability of a tag Z is calculated by dividing the trigram frequency XYZ by the bigram frequency of XY, i.e. the two preceding tags. The estimates for these frequencies have been calculated on the basis of a hand tagged corpus of 20,000 words from the daily German newspaper *Frankfurter Rundschau*. The performance of the tagger when using the large feature set has been evaluated with a 5000 word test corpus. It produced a total of 84.7 % correct solutions.

3. The syntactic component

In designing the parser we were guided by two pragmatic considerations. On the one hand, the parser should be able to work with arbitrary texts, as can MORHPHY. On the other hand, one does not need a complete parser in order to show that a syntactic component may augment the quality of a hitherto purely probabilistic tagger. We therefore decided that the parser should work with a defined subset of our newspaper corpora. This subset is the set of all sentences without commas.

The syntactic analysis proceeds in two steps. First, the boundaries of noun and prepositional phrases and the structures of the verbs are identified. This is accomplished using a pattern matching process. For the identification of the verb structure the program uses a list of 60 different patterns of main, auxiliary and modal verbs which could appear in German main clauses. These patterns include all tenses, sentence forms, and up to three modal verbs. The following example shows the pattern for passive sentences in the future with one modal verb:

> +,[root={werden},modtemp={PRÄ,PRT,KJ1,KJ2}],*,[pos={verb},form={PA2},mainverb, passive],[root={werden},form={INF}],[pos={verb},type={modal}, form={INF}]

The pattern consists of six specifications which are separated by commas. (1) The + sign says that there must be at least one word before the auxiliary *werden*. This specification excludes questions and imperatives. (2) The auxiliary *werden* must be in a finite form. (3) the * sign specifies that there may be an arbitrary string of words between the auxiliary and the participle. (4) The past participle is the main verb of passive sentences. (5) the infinitive form of *werden*. (6) The infinitive form of a modal verb. This pattern matches the sentence:

> *Der Apfel wird morgen gegessen werden müssen*.
> (The apple will have to be eaten tomorrow.)

The shortest pattern contains one verb, which is the finite main verb, and the longest patterns contain six verbs: three modals, two auxiliaries, and a participle of the main verb. If a sentence matches more than one verb pattern, the pattern with the highest number of verbs is chosen.

The identification of the verb pattern serves two purposes. First, the information about the main verb and about the sentence form will be used later to establish the list of expected cases. Second, it permits the disambiguation of words which can be verbs as well as other parts of speech. If the three word example which is given in Table 1 is regarded as a sentence, the verb matcher finds a predicate structure with only one main verb, *süßen*, and the interpretations of *süßen* as an adjective and as an infinitive can be discarded.
In order to bracket the noun phrases the program uses an ordered list of all possible patterns for noun phrases up to a length of five words. This list includes all possible combinations of indefinite, possessive, demonstrative and personal pronouns, numerals, adjectives, participles, proper nouns, etc. It does not include embeddings of prepositional phrases in nouns phrases, which have to be analysed using special heuristics.

After the noun phrase has been bracketed, each phrase is unified for case, number, gender, and definiteness. During this process some of the parsings are discarded. The three words *Alle süßen Kekse* can be parsed into one noun phrase, which includes an indefinite pronoun (*alle*), an adjective (*süßen*), and a noun (*Kekse*). They can also be parsed into two noun phrases, the first formed by the indefinite pronoun as its head and the second by the adjective and the noun. However, this second noun phrase would not unify: Noun phrases which consist of an adjective and a noun are indefinite. *Kekse* may be nominative, genitive, or accusative plural, but not dative. In definite noun phrases in plural, the adjective *süßen* goes with all cases. In indefinite noun phrases however, *süßen* must be dative. Therefore, an indefinite noun phrase *süßen Kekse* does not exist. This example shows how useful an elaborate morphological analysis can be for the interpretation of German case and sentence structures.

The last two steps of the syntactic analysis consist of the unification of the sentence for person and number and the elimination of noun phrase interpretations in which the case does not conform with the case frame of the main verb. A case frame is defined as an unordered list of the cases of the objects which the main verb takes. Some examples of case frames are:

haben (to have): nominative, accusative
sein (to be): nominative, nominative
geben (to give): nominative, dative, accusative
lehren (to teach): nominative, accusative1, accusative2
leben (to live): nominative

The following rule uses these case frames.
IF there is a noun phrase allowing only one case, and
this case is part of the case frame of the sentence, and
there is also a second noun phrase which is ambiguous with regard to case, and
one of its cases corresponds to the case of the unambiguous noun phrase,
THEN that interpretation is removed from the list of interpretations of the second noun phrase.

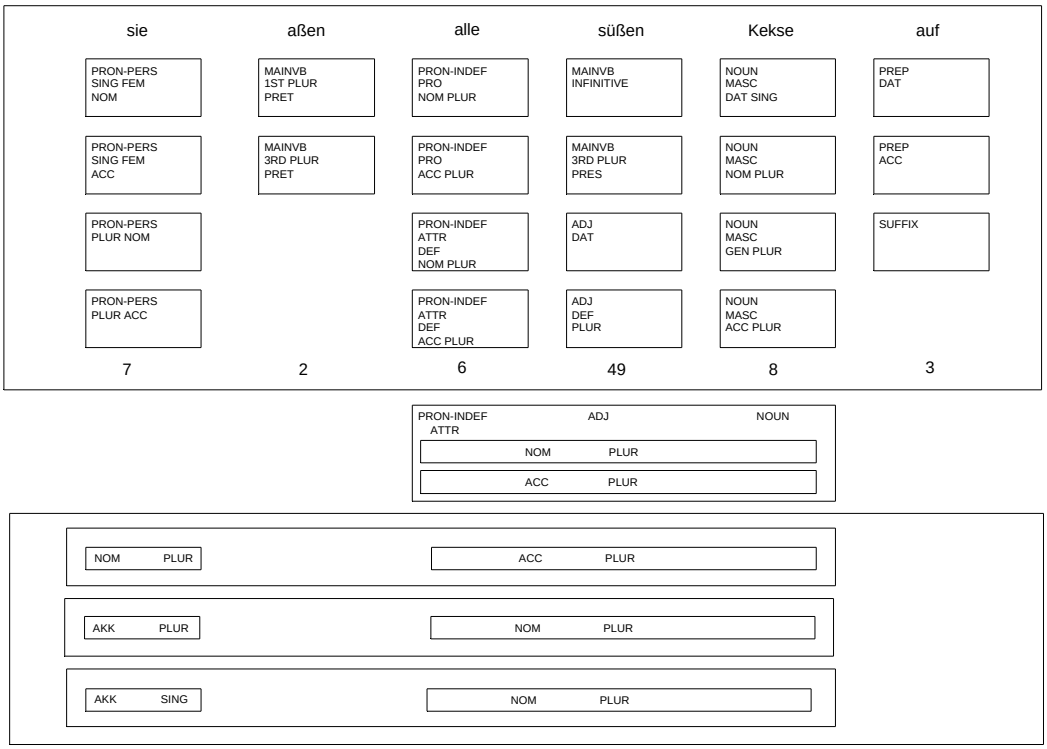

Table 3: Three steps during the analysis of the sentence: *Sie aßen alle süßen Kekse auf.*

This rule allows the interpretation of the following sentence:
Den Mann liebt die Frau.
Here the verb *liebt* (loves) requires a nominative and an accusative noun phrase. *Den Mann* must be accusative whereas *die Frau* can be either nominative or accusative. Therefore, it is the woman who loves the man and not the other way around.

Another rule for the detection of the grammatical case of a noun phrase goes is:
IF one of the possible cases of an ambiguous noun phrase is part of the case frame, and
no other noun phrase can possibly have this case,
THEN all other cases of this ambiguous noun phrase are removed from the list of interpretations.

Table 3 shows an example which demonstrates the successive reduction of tags during the syntactic analysis of the sentence

Sie	*aßen*	*alle*	*süßen*	*Kekse*	*auf.*
They	ate	all	sweet	cookies	up.

The upper part of the figure shows some of the tags produced by the first component of MORPHY, i.e. before the context of the words was considered. The total number of tags for each word is given below the shown tags. Since the number of interpretations which can be given to a sentence is the product of the number of interpretations of each word this sentence has 98.784 readings during this stage of analysis. The pronoun *sie* is ambiguous with regard to case and to number and the verb *aßen* with regard to person. The different interpretations of the three words *alle*, *süßen*, and *Kekse* have already been explained in Section 1. *Auf* can either be a preposition which must be followed by an object in the dative or the accusative case, or it can be a prefix of the finite main verb which has been shifted to the second position. The syntactic analysis starts after the assignment of all possible tags by the first component of MORPHY. The first step is the identification of the predicate structure. Here the following five possible verb patterns were found:

(1) finite main verb (*aßen*)
(2) finite main verb (*süßen*)
(3) infinitive of a main verb (*süßen*)
(4) finite main verb (*aßen*) & finite main verb (*süßen*)
(5) finite main verb (*aßen*) & infinitive of a main verb (*süßen*)

Only the first two of these five possibilities are legal predicates. In selecting the main verb the following heuristic is used: when there are two candidates for the main verb and one of them is ambiguous with regard to part of speech, then select the other candidate. If this heuristic had not been applicable the program would find the main verb after parsing the prepositional phrases where *auf* is recognised as the prefix of a shifted verb. *Aufessen* is a common verb and part of the dictionary whereas **aufsüßen* is not. Following this interpretation of the predicate structure, the tags which describe *süßen* as a verb are discarded.

During the following steps the two noun phrases *sie* and *alle süßen Kekse* are parsed and unified with the procedures previously described. The middle part of Table 3 shows the two interpretations of the second noun phrase after unification: the phrase must be a nominative or accusative plural. The set of tags for the noun phrase *sie* did not change. At this stage of analysis the number of interpretations of the sentence is reduced to 16: four interpretations of *sie* times two of *aßen* times two for the second noun phrase. Two additional constraints reduce this number to three interpretations. The case frame of *aufessen* specifies a nominative and an accusative case. Therefore, the interpretations in which both noun phrases are nominative and in which both are accusative are eliminated The second constraint applied is the number and person agreement between the subject and the finite verb. Finally, there are three interpretations left corresponding to the following three English sentences:

They ate up all the sweet cookies.
All the sweet cookies ate them up.
All the sweet cookies ate her up.

4. **Results**

The aim of the study was to find out if the quality of probabilistic taggers can be enhanced by means of additional syntactic rules. In order to answer this question we carried out two tests. In the first test our aim was to discover whether the predicate patterns allow a reliable definition of the main verb. This test was carried out with all 581 sentences of the LIMAS corpus not containing commas. Human raters designated the main verbs of these sentences plus their infinitive forms. In comparing these »correct" main verbs to the infinitives of the main verbs found by the program, we discovered six different possible results:

1. The two verbs are identical, meaning that the prediction is correct.
2. The program finds the same word designated by the human raters. But because the word form is ambiguous, the program can not determine which of the possible infinitive forms is correct. In the test sentence

   Dann schaltet sich der Vorschub aus.

   the main verb *schaltet* can be derived from the infinitive *ausschalten* (switch off) or it can be the second person plural imperfect of the infinitive *ausschelten* (reproach).
3. The program correctly identifies the infinitive form of one main verb. However, the sentence includes at least one other main verb not found by the program. This error may result from the heuristic: If there are two candidates for the main verb, and one of them is ambiguous with regard to the part of speech, then select the other candidate.
4. The program finds two main verbs whereas the sentence has only one. This error may happen when complex noun phrases include an embedded verb as in the test sentence:
   Die Übernahme von § 136ff der Weimarer Verfassung von 1919 in das Grundgesetz erfordert einerseits die Auseinandersetzung mit den unter der Geltung der Reichsverfassung hierzu entwickelten Lehrmeinungen.
   In this case, the participle *entwickelten* has been defined erroneously as the second finite main verb.
5. The parser identifies the main verb incorrectly.
6. The main verb cannot be found because it is not in the lexicon.

For the purpose of evaluating the difference between the main verb assignments of the program and the designations of the human raters, only case 1 was considered as a correct solution. Cases 2 and 6 were discarded. Cases 3, 4, and 5 were regarded as incorrect assignments. This procedure gave the following results:

main verb correctly identified:	562 sentences
wrong assignment of the main verb:	6 sentences
discarded (cases 2 and 6): :	13 sentences.

The program identified and lemmatised 97 % of the main verbs correctly. If the errors due to shortcomings of the lexicon are discarded, this number goes up to 99 %.
The second test compared the results of MORPHY with the syntactic component to the results without the syntactic component. This test was done with 350 randomly chosen sentences without commas from the German newspaper *Frankfurter Rundschau*. These sentences, which had never been previously used in this project, have a total number of 4034 words. In the case of 3777 words the two programs produced identical tags. We subsequently considered the remaining 257 words for which the tagger with the syntactic component gave a different result than the tagger without the syntactic component.
In 169 of these 257 cases (66 %) the tagger with syntax selected the correct tag and the tagger without the syntax the wrong one. In 29 cases (11 %) the program without syntax was correct and the program with syntax wrong. In 59 cases (23 %) both programs had at least one grammatical feature wrong. Thus the syntactic component augmented the quality of the probabilistic tagger considerably. The criterion defining correct judgement was very strict. The tags consist of several features, and if only one of them was wrong then the whole tag was considered wrong.

The results do not specify the percentage of the tags that MORPHY without syntax got wrong that were corrected by the syntactic component, since we could not tell the number of cases in which both components made the same error. However, this value can be estimated on the basis of an earlier study which showed that on the average, 84,7 % of the tags which MORPHY produces with the large feature set are correct. The corpus of this study also consisted of texts of the *Frankfurter Rundschau*. If we subtract the 29 cases in which only the program without syntax produced the correct tag from the 169 cases in which the program with syntax produced the correct and the program without syntax produced the wrong tag, then we get a net gain of 140 words equalling 3.5 % of the test corpus. The effect of the syntactic component would thus be an increase in the hit rate from 84,7 to 88,2 percent. In other words, the error rate is reduced by about 20 percent.

5. General considerations

There seems to be something unusual about what we have done in our study. When a tagger is used in connection with a syntax parser, it usually functions as a front end for the parser. In this study however, the parser is a component of the tagger. One might consider using our tagger, which includes a parser, as a front end for a parser which is even more effective. However, a language understanding system containing two syntax parsers would not make much sense. There are two considerations which lead us to use a syntax parser as a component of a tagger. The first is more pragmatic: tagging is a task for which the performance or quality of a program can be measured easily, as in the percentage of correct tags. This gives us a criterion to quantify the contribution of the parser (20 % fewer errors). In other applications of natural language understanding systems it is more difficult to judge the success of the programs. The second consideration is more theoretical: it is an accepted fact nowadays that language understanding works bottom up and top down at the same time. Morphological ambiguities, for instance, are resolved on the basis of syntactic constraints (top down) and syntactic ambiguities on the basis of the results of the morphological analysis (bottom up). It is therefore not fully appropriate to think of either one of the two systems as a component of the other. Morphology, syntax, and also semantics should rather be seen as different modules which transfer information in all directions.

The question now remaining is what would be the optimal way of relating the syntactical and the probabilistic components. If we assume that two different systems, a probabilistic and a rule governed system, work together in a modular fashion during language comprehension, then the flow of information between these modules should be governed by the principle that the certain information gets transfered first. This would signify a change in the organisation of the interaction of the modules as has been carried out here. In this study there were two points of contact between the modules: first, the lexicon component of MORPHY passed all possible grammatical descriptions to the parser. After this, the parser passed the reduced list of syntactically legal descriptions to the probabilistic component of MORPHY. A better, but more difficult way of interaction would be for each component to pass the certain results on to the other component. The certain results consist of those interpretations of the parser for which no others exist, as well as the guesses of the tagger with the highest probabilities. The results of our study demonstrate that further work in this direction is promising.

References

Church, K.W. (1988). A stochastic parts program and noun phrase parser for unrestricted Text. Second Conference on Applied Natural language Processing. Austin, Texas, 136-143.

Lezius, W. (1996). Morphologiesystem Morphy. In: R. Hausser (ed.): Linguistische Verifikation. Dokumentation zur Ersten Morpholympics. Niemeyer: Tübingen. 25-35.

Lezius, W.; Rapp, R.; Wettler, M. (1996). A morphology system and part-of-speech tagger for German. In: D. Gibbon (ed.): Natural Language Processing and Speech Technology. Berlin: Mouton de Gruyter. 369-378.

Lezius, W.; Rapp, R.; Wettler, M. (1998), A freely available morphological analyzer, disambiguator, and context sensitive lemmatizer for German. Proceedings of the COLING-ACL, 1998.

Wahrig, G. (1997), Deutsches Wörterbuch. Gütersloh: Bertelsmann.

[1)] this work has been supported by the Heinz Nixdorf Institute grant no. 595/98