The Analysis of German Separable Prefix Verbs in the Microsoft Natural Language Processing System

Michael Gamon
Tom Reutter

9/25/97

Technical Report
MSR-TR-97-15

Microsoft Research
Microsoft Corporation
One Microsoft Way
Redmond, WA 98052

The Analysis of German Separable Prefix Verbs in the Microsoft Natural Language Processing System*

Michael Gamon and Tom Reutter
Microsoft Research
One Microsoft Way
Redmond, WA 98052
USA
mgamon@microsoft.com, treutter@microsoft.com

0. Abstract

In this paper we present an analysis of separable prefixes in German within the Microsoft Natural Language Processing System. We illustrate a simple solution to the problem of separated prefixes in our augmented phrase structure grammar which uses binary syntactic rules. The solution is based on the treatment of separated prefixes as verbs. We discuss the semi-automatic generation of the necessary lexical entries and the syntactic analysis based on these entries. We show that under our approach only one new rule needs to be added to our grammar to accommodate the facts, and that only small modifications to existing rules are necessary[1].

1. Separable Prefixes – the Problem

Separable prefixes in German pose an interesting linguistic and computational challenge[2]. In this section we briefly outline the relevant generalizations and facts.

Separable prefixes can be separated from the verb stem in two ways: morphologically and syntactically. Morphologically, the prefix and the stem can be separated by the infinitival marker "zu" and the past participle prefix "ge" as in the examples in (1) below:

(1) abfahren ab*ge*fahren ab*zu*fahren

Syntactically, the prefix can be in verb-final position, while the finite verb stem is in verb-second or in verb-initial position:

(2) Hans fährt heute ab.
Fahrt doch heute ab!
Fahren sie heute ab, können wir morgen unsere Ruhe haben.

For reasons of space, we will exclusively deal with syntactic separation of the prefix from its stem and its analysis in the German grammar in this paper. Our morphology component, however, does handle the facts in (1).

It is a well-known fact that the separated prefix occupies the same position in the sentence as non-finite verb forms in analytic tenses do, while the finite stem is located in the typical "verb-second" position of the finite verb in main declarative sentences and a subset of complement clauses. In other words, the prefix occurs where the past participle would occur in a perfective sentence.

Linguistically, the syntactic separation of prefix and stem has given support to generative analyses in terms of verb-movement[3] of the finite stem from a VP-final base position. In more descriptively oriented approaches, the separable prefixes have been called "Nachverben" (Weinrich 1993). From a computational point of view, the challenge consists of two parts: assigning a correct structure to a sentence with separated prefix, and doing a look-up of the complete verb (prefix plus stem) to provide the correct lemma and subcategorization information associated with that entry[4]. In order for these challenges to be met, an appropriate set of lexical entries has to be created in the dictionary, containing information about possible prefix-stem combinations.

In the next section, we briefly introduce the MSNLP system. Section 3 deals with the lexical and syntactic details of our analysis.

2. The Microsoft Natural Language Processing System

The MSNLP system currently encompasses English, French, German, Spanish, Chinese, Japanese and Korean. The parsing engine is separate from the grammar, morphology, and dictionary files of the languages involved. The English system is the most fully developed, work on the German grammar started in October 1996.

The first syntactic component of the system is called "Sketch". Sketch grammars use a computational dictionary containing part-of-speech, morphological, and subcategorization information to yield an initial syntactic analysis (the *sketch*). The rules used in *sketch* have no access to any semantic information that would allow the assignment of semantic structure such as case frames or thematic roles.

Further analysis in the English system (and not yet implemented for the other languages) proceeds through a stage of reattachment of phrases using both semantic and syntactic information to produce the *portrait*, then to a first representation of some aspects of meaning, the *logical form*, and to word sense-disambiguation and higher representations of meaning. In this paper we restrict our attention to the German sketch grammar.

---

* Thanks to Lucy Vanderwende and Hisami Suzuki for comments on an earlier draft of this paper.

[1] We do not make any claims about the feasibility of a similar solution in other formal frameworks such as LFG, HPSG etc.

[2] In fact, the standard initial question of visitors who hear about the German part of the MSNLP system is: "So how do you handle the prefixes?"

[3] See, for example, Bierwisch (1973), Grewendorf (1988, 1990), Wunderlich (1983).

[4] See also Volk(1988), Russell (1985), Uszkoreit (1982, 1984) for analyses of separable prefixes in the GPSG framework.

A bottom-up parallel parsing algorithm is applied to the sketch grammar rules, resulting in one or more analyses for each input string, and defaulting in cases (such as PP attachment) where semantic information is needed at a later stage of the processing (*portrait*) to give the correct result. Binary rules in an augmented phrase structure grammar are used because they have been found well-suited for the successful analysis of natural languages (Jensen et al. 1993, pp. 33-35; Jensen 1987, pp. 65-86). Figure 1 gives a template for the rule formalism for a binary rule, in this case a rule that combines a verb phrase with a prepositional phrase to its right.

Each sentence parse contains syntactic and functional role information. This information is carried through the system in the form of arbitrarily complex attribute-value pairs, which can be manipulated on the right-hand side of the rules (Figure 1). The *sketch* always produces at least one constituent analysis, even for syntactically invalid input, and displays its analyses as parse trees. FITTED parses are obtained when an input string cannot be parsed up to a sentence node (possibly because it is a noun phrase, a sentence fragment, or otherwise deficient). FITTED parses contain as much constituent structure as the grammar could assign to the input string.

```
VPwPPr:
VP ( Condition 1 &
       Condition 2 & ..... )
PP ( Condition 1 &
       Condition 2 &  ..... )
--> VP {action 1;
          action 2; ....}
```

Figure 1. Outline of the binary rule combining a VP with a PP to its right (VPwPPr)

Two representations of the syntactic analyses are available for display (Figure 2). One strictly follows the derivational history of the parse, and is therefore binary-branching. In the binary tree the names of the rules that have produced a node are displayed to the right of that node. The second type of tree exclusively contains dependency information of heads/non-heads. It is n-ary branching, or "flattened," and is computed from a small set of syntactic attributes of the analysis record. The * indicates the head of the phrase. This second type of tree accords better with our intuitive understanding of many structures.

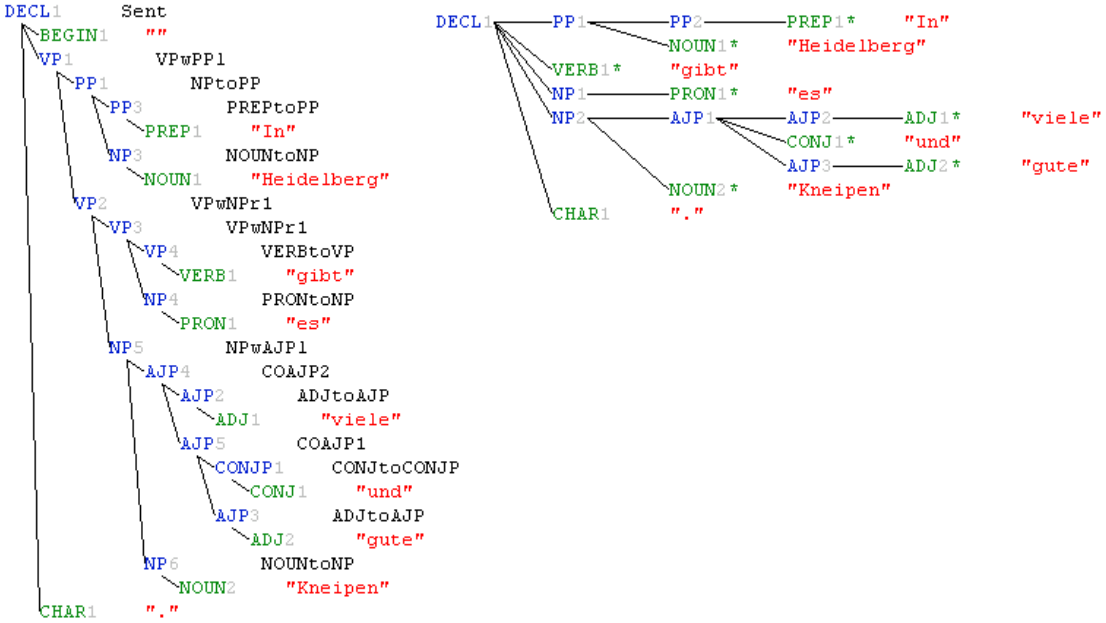


Figure 2: A derivational tree and a "flattened" tree for the sentence "In Heidelberg gibt es viele und gute Kneipen"

The sketch grammar is written in G, a Microsoft-internal programming language that has been specially designed for NLP. The German grammar contains 152 rules, which vary in size from two to 600 lines of code, according to the number of conditions and actions that they contain. The German dictionary contains 147,000 entries (uninflected, except for pronouns); inflectional morphology is nearly complete. Of a corpus of 424 sentences from various sources (news, technical writing, novels etc.) 68% of the sentences currently receive a complete, non-FITTED parse. The average sentence length of that test-corpus is 15.3 words.

For the Indo-European languages in the MSNLP system, sketch grammars are not being developed from scratch, but rather by taking the English grammar as a starting point and over time modifying it and adapting it to the language-specific needs[5].

The goal of all Natural Language Research and Development at Microsoft Research is to produce a broad coverage multilingual NLP system that is not tailored to any specific application, but has the potential to be used by any of them. We would like to stress the point that

[5] For a discussion of this "grammar sharing" approach, see Pinkham(1996) for French, and Gamon et al. (1997) for French, Spanish and German.

this system, by its nature as a broad-coverage system, does not restrict itself to any particular theoretical framework and is not intended to mirror speaker intuitions in the way generative grammar does. The goal is to parse even ungrammatical input as far as possible, setting error flags for disagreement etc.

3. Separable Prefixes in the MSNLP System

3.1 Outline

Recall from section 1 that the two computational challenges posed by separable prefixes are the assignment of a correct structure and the lookup of the complete verb. German grammar in the MSNLP system, encoded in binary rules and based on the English grammar, faced the challenge of accommodating separable prefixes in as efficient a way as possible.

The solution we chose is based on the linguistic generalizations mentioned in section 1. Because the separated prefix occupies the same position that a non-finite verb occupies in an analytic tense, we decided to assign VERB as the part of speech for all prefixes. The result of this is that separated prefixes can expand into VPs and most importantly that all rules which pick up complements and modifiers to the left (and to some extent also on the right) of a non-finite verb will perform the same task for separated prefixes. We only needed to add one new rule that combines the finite verb stem with the VP of the separated prefix and its modifiers/complements, and then performs the lookup of the complete verb. This rule is very much parallel to the rule that combines a finite auxiliary with a participial VP.

In the following two sections we will present the details of this analysis, starting with the semi-automatic derivation of the necessary lexical entries.

3.2 The Lexical Entries

Our starting point is a dictionary containing a set $\mathbf{V_{spfx}}$ of verbs with separable prefixes. Each of these verbs is marked with a binary *Sepfix* feature, but with no explicit indication of what the separable prefix is[6]. Our approach requires that all separable prefixes be treated as verbs and also that we be able to match the verbal stem and its separated prefix during parsing. Therefore, for each element in the set of lexicalized separable prefix verbs $\mathbf{V_{spfx}}$, we need to determine its prefix **P** and its stem **S**; then, to the entry **P** we must add a verbal sense (marked with the special feature *Sepfix*) to allow the prefix to be parsed as a verb. Finally, we must merge the prefix **P** into a list *Vptc* of possible separable prefixes for the entry **S**. For example, if our dictionary contains just the three separable prefix entries *abgehen, abmachen*, and *aufmachen*, then we:

- determine the prefix and stem for each separable prefix verb: *ab|gehen*, *ab|machen*, *auf|machen*
- add a verb sense to the entries *ab* and *auf*, and
- define the possible prefix list for the stem *gehen* to be the singleton list (ab), and define the possible prefix list for the stem *machen* to be the list (ab, auf).

We choose a two-step approach to these dictionary modifications: first, we create an exhaustive list of separable prefixes occurring in the dictionary. Then, using this list, we effect the required changes to the dictionary. Validating the exhaustive list requires a small amount of human intervention, while the dictionary modifications are fully automatic. Although there is some redundancy between the two steps, step one has the fortuitous side effect of yielding a comprehensive list of separable prefixes, which is useful for our morphological component.

3.2.1 Step 1: Generating an Exhaustive List of Separable Prefixes

We start with a list $\mathbf{SP_{com}}$ of 121 common separable prefixes (*ab*, *auf*, *hin*, *hinauf*, etc.), such as could be found in a standard German grammar, but not including the less frequent separable prefixes (e.g. *glatt*, *höher*, *ski*, *rad*, *liegen*, *stecken*, *nebeneinander*). Our goal is two-fold:

- Generate an exhaustive list $\mathbf{SP_{exh}}$ of all the separable prefixes actually contained in the separable prefix verb entries $\mathbf{V_{spfx}}$ in our lexicon (remember that a separable prefix verb entry does not explicitly name its prefix)
- Ensure that all verb stems contained in $\mathbf{V_{spfx}}$ are added to the dictionary, if not already present.

To accomplish these objectives, we classify each separable prefix verb $\mathbf{V_i}$ in the set $\mathbf{V_{spfx}}$ according to the (putative) prefix P and stem S comprising $\mathbf{V_i}$:

1. $\mathbf{S_1}$: P is a common separable prefix (i.e. P $\in$ $\mathbf{SP_{com.}}$) and S is a lexicalized verb.
2. $\mathbf{S_2}$: P is a lexicalized word (but not a common separable prefix) and S is a lexicalized verb.
3. $\mathbf{S_3}$: P is neither lexicalized nor a common prefix, but S is a lexicalized verb.
4. $\mathbf{S_4}$: P is a common separable prefix (i.e. P $\in$ $\mathbf{SP_{com.}}$) but S is not lexicalized..
5. $\mathbf{S_5}$: $\mathbf{V_i}$ has two segmentations, one fulfilling $\mathbf{S_3}$, the other $\mathbf{S_4}$
6. $\mathbf{S_6}$: No prefix P can be found (neither in $\mathbf{SP_{com}}$ nor in the lexicon), nor a lexicalized stem S

$\mathbf{S_3}$ and $\mathbf{S_4}$ are, of course, mutually exclusive, since $\mathbf{S_5}$ contains their intersection. An example of a verb belonging to $\mathbf{S_5}$ is *einverleiben*: *leiben* is a lexicalized verb, but *einver* is not lexicalized, while *ein* is a common separable prefix, but *verleiben* is not lexicalized[7]

It is a simple programming task to traverse the dictionary, classify each lexicalized separable prefix verb into one of these six sets, determine its prefix P and stem S in the process, adding P to the list of candidates for expanding $\mathbf{SP_{com}}$ and, if S is not already lexicalized,

---

[6] Our broad-coverage dictionary contains 8292 such separable prefix verbs.

[7] Set intersections other than $\mathbf{S_5}$ are theoretically possible, but less interesting. In particular, we postulate that if any element of $\mathbf{S_1}$ happens also to belong to another set, its membership in the second set is purely coincidental, i.e. the "right" segmentation of the verb is the one given in $\mathbf{S_1}$. We have found no counterexamples.

adding it to a list of verb stems to be automatically merged into the dictionary. For sets S_2 through S_6 the automatic determination of prefixes and stems can occasionally be ambiguous and, very rarely, yield spurious segmentations, so the lexicographer reviews the new prefixes and stems before they are actually folded into the dictionary.[8]

Almost 91% of the separable prefix verbs fall into the set S_1 and require no further processing. Just one single verb is found in S_6, which turns out to be the result of a lexical error. The remaining sets provide a well-motivated list of prefixes to be added to SP_{com} and verb stems to be added to the lexicon.

S_2, the set of verbs containing a lexicalized stem preceded by a putative prefix which is lexicalized but not yet in the list of prefixes SP_{com}, represents 6.8% of all separable prefix verbs, and provides a wealth of new prefixes. A small percentage (less than 2%) of the verbs in S_2 are ambiguous with respect to segmentation, e.g. *bereit|erklären* vs. *bereiter|klären* (*Bereiter* is in the lexicon), *hinten|ansetzen* vs. *hintenan|setzen*, *vier|teilen* vs. *viert|eilen*, etc. In such cases, our algorithm produces both segmentations, and the lexicographer selects the appropriate one. Additionally, the lexicographer reviews the unambiguous cases to ensure that spurious prefixes are not added to SP_{com}. In fact, only one such case is identified: *uraufführen* should not be segmented as *ur|aufführen.* All in all, the 564 verbs in S_2 yield 185 new prefixes to be added to SP_{com}, including *acht*, *dagegen*, *hintan*, *klug*, *vorweg*, etc.

S_3, in which the putative prefix is not even lexicalized, is much smaller than S_2 (S_3 contains 69 entries, or 0.8% of all separable prefix verbs), but it also yields a few new separable prefixes: *brach(-liegen)*, *dünne(machen)*, *verschütt(gehen)*, etc. Conversely, S_4 (91 entries, or 1.1% of all separable prefix verbs) yields a number of new verbal entries to the dictionary, most of which only occur in conjunction with their separable prefix: *(aus)bogen*, (aus)*ixen*, etc. These entries are automatically generated in a form in which they can be automatically added to the dictionary. Again, the lexicographer resolves a few ambiguities: *hinauf|kraxeln* vs. *hin|aufkraxeln*, etc., before the additions are finalized.

Finally, all verbs in S_5 have to be resolved by the lexicographer, since they are by definition ambiguous: *ab|basten* vs. *abb|asten*, *ab|specken* vs. *absp|ecken,* etc.. This very small set of verbs (23 items, or 0.28% of all separable prefix verbs) is fairly evenly split between ones yielding a new verb entry (*basten* and *knutschen* are added based on *abbasten* and *abknutschen*) and ones yielding new separable prefixes (*zusammen* is added because of *zusammenstellen*).

Thus, with minimal human intervention (a day or two of programming and one day of lexicographic review of the data), about a hundred new entries are added to the dictionary to accommodate previously unlexicalized verb stems, and the list of common separable prefixes SP_{com} containing just 121 prefixes grows to an exhaustive list SP_{exh} containing over 300 prefixes, representing all the separable prefixes contained in separable prefix verb entries in the lexicon. This enhanced prefix list facilitates morphological processing, and is required for the next step: modifying the dictionary to accommodate our approach to parsing separable prefix verbs.

3.2.2 Step 2: Modifying the Dictionary

Given a dictionary containing a set V_{spfx} of separable prefix verbs and an exhaustive list SP_{exh} of separable prefixes contained in V_{spfx} and given that all verb stems contained in V_{spfx} are also lexicalized, our tasks are:

- For each prefix P in SP_{exh} add a verb sense to the entry P, flagged with the bit *Sepfix*
- For each separable prefix verb in V_{spfx} with prefix P and stem S, modify the entry S to include P in its list of possible separable prefixes *Vptc*

We use the same approach as in the previous step to process each separable prefix verb in the dictionary. This time, though, all of the separable prefix verbs fall into the class S_1, i.e. the verb can be (fully automatically) split into a prefix P from the set SP_{exh} and a lexicalized stem S. Then:

1. A verbal sense is added for the entry **P**, if it does not already exist.
2. In the entry **S**, the prefix **P** is merged into its list of separable prefixes *Vptc*

For example, if the dictionary contains just the four separable prefix verbs, *angeben*, *anheimstellen, anstellen*, and *aufstellen*, then the exhaustive list of separable prefixes SP_{exh} consists of {anheim, auf, an}. The following modifications are made to the dictionary

an: Add verb sense with special feature *Sepfix*
anheim: Add verb sense with special feature *Sepfix*
auf Add verb sense with special feature *Sepfix*
geben: Add attribute-value pair [Vptc = (an)]
stellen: Add attribute-value pair [Vptc = (an, anheim, auf)]

This step requires no intervention on the part of the lexicographer, but now the way is paved for syntactic analysis of separable prefixes.

3.3 The Syntactic Analysis

Once separated prefixes have a lexical entry with a verbal part of speech and a particular feature *Sepfix* that sets them apart from regular verbs, and once verbal stems have been augmented with the *Vptc* attribute containing a list of separable prefixes, that can occur with the stem, we have all the necessary prerequisites for a simple syntactic analysis in binary rules.

In syntax, one new binary rule is needed to combine the finite stem and the verb phrase headed by the prefix. This rule, called "VPwSepfix" combines two VPs to form another VP. Note that all binary rules in the MSNLP system operate on phrases, not on heads. The basic schema of the rule is illustrated below[9]:

---

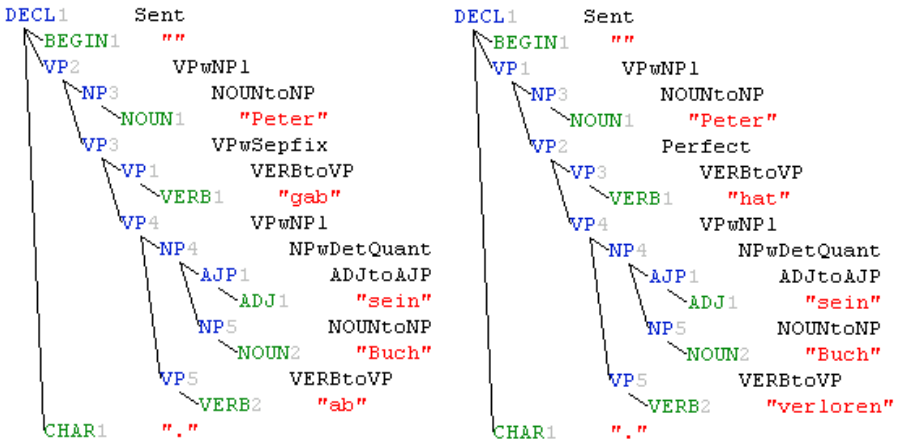
[8] This lexicographic review required less than a day's worth of work.

```
VPwSepfix:
        VP ( ^Prmods &  (Psmods -> (nvalues(Psmods)==1 & Comma)))
        VP ( Sepfix & ^Infld & Lemma in? Vptc(VP#1)
              & ^SepfixVerb.....)
        -->VP { %%VP#2; Vform=Vform(VP#1); FrstV=Head(VP#1); Mood=Mood(VP#1);
                 +Vsecond; -Vfinal;
                 -Lemma; Lemma = add_prefix(Lemma(VP#2), Lemma(VP#1));
                Temp=Verb(lex_get(normalize(Lemma, "case_accents"),0)); Subcat = Subcat(Temp);
                 -Sepfix; +SepfixVerb; .....}
```

In words, the first VP that has no premodifiers (has not been combined with anything to its left) and has no postmodifier except, possibly, a comma (as in [[er gab ,] [[ohne zu zögern ,] [seine Schlüssel ab]]). The second VP has to bear the *Sepfix* feature and must not be inflected. Importantly, it has to be in the Vptc list of the first VP. Furthermore, it must not bear the feature *SepfixVerb* (see below) to avoid application of the rule to a VP that already contains both a stem and its prefix. The result of the successful application of this rule is a new VP record, which bears the inflectional information of the finite verb stem, and is marked as a Vsecond compatible record (meaning that the V2 position is occupied). The lemma of the new record is removed and replaced by a concatenation of the prefix and the lemma of the first VP. A lexical lookup is performed on the new lemma, and the subcategorization information of the new lemma (prefix plus stem) is added to the VP record. Finally, the *Sepfix* feature is removed from the new record, and the *SepfixVerb* feature is added. This helps other syntactic rules to distinguish VPs that already contain both a separable prefix and its stem from those that do not.

Apart from the addition of this new rule, only minor changes are necessary in other rules, to ensure that separated prefixes and their VPs are treated properly. To give just some simplified examples, modal or auxiliary VPs should not combine with either *Sepfix*- or *SepfixVerb*- VPs (to prevent formation of [wurde ab]) and clause-formation in general had to be blocked for VPs with the *Sepfix* feature (i.e. VPs which only contain a separated prefix without a corresponding stem).

With these minor modifications in the German grammar, separable prefixes can successfully be analyzed. With the exception of one binary rule no additional rule apparatus is necessary at all. Take, for example, the rule that picks up noun phrases to the left of a VP. This rule will apply exactly in the same fashion to form a verb phrase [sein Buch ab] that it will for a verb phrase [sein Buch verloren]. The same holds for all other rules that combine verb phrases with preverbal or postverbal modifiers and complements. 13 such rules exist in the current MSNLP German grammar, all of which would have to be duplicated for separated prefixes if the separated prefix were not analyzed as a verb. To illustrate the parallelism of structures assigned to separated prefix examples and similar examples with an analytic tense, two sample parses are shown below. The reader should keep in mind that the parallelism in structure extends to much more complicated structures as well, involving complement clauses, subordinate clauses, adverbial and prepositional modifiers etc.



4. Summary and Conclusion

We have demonstrated that an analysis of separable prefixes in German has been implemented in the MSNLP system without the need for any special mechanisms or any elaborate rule machinery. We presented the strategy of semi-automatic derivation of the necessary lexical entries from the existing entries of verbs with their prefixes. The resulting entries of this lexical processing consist of an entry for each separable prefix as a verb, and entries for the corresponding verbal stem(s) which are augmented with a list of the prefixes the stem can occur with. The newly generated lexical entries then serve as the basis for a syntactic analysis of separated prefixes as verbs. Only one new rule had to be added to the grammar, which combines two verb phrases, one headed by a verb stem, the other headed by a prefix. With few minor modifications of existing rules this analysis captures the linguistic generalization that the separated prefix occurs in the same structural position as a participle in an analytic tense.

---

[9] The actual rule as currently implemented is, of course, more complicated than this simplified schema. It contains 51 lines of code.

We believe that this analysis demonstrates two important points: First, the grammar formalism in the MSNLP system allows us to approach non-trivial linguistic problems posed by languages other than English in a straightforward and efficient way. Our analysis of separable prefixes only involved the addition of one single rule combining the stem-VP and the prefix-VP to our grammar with only minor modifications in other rules. Second, this analysis naturally captures the linguistic generalizations about the position of separated prefixes.

References

Bierwisch, Manfred. 1973. *Grammatik des deutschen Verbs*. Studia Grammatica II, Akademie-Verlag, Berlin.

Gamon, Michael; Carmen Lozano, Jessie Pinkham, Tom Reutter, 1997. Practical Experience with Grammar Sharing in Multilingual NLP. Paper presented at the "From Research to Commercial Applications" workshop of the ACL/EACL97.

Grewendorf, Günther. 1988. *Aspekte der deutschen Syntax*. Tübingen, Narr.

Grewendorf, Günther. 1990. Verb-Bewegung und Negation im Deutschen. In *Groninger Arbeiten zur Germanistischen Linguistik* 30. pages 57-125.

Jensen, Karen, George Heidorn, Steve Richardson, eds. 1993. *Natural Language Processing: The PLNLP Approach.*. Boston, Kluwer Academic Publishers.

Jensen, Karen. 1987. Binary rules and non-binary trees: Breaking down the concept of phrase structure. In *Mathematics of language*, ed. Alexis Manaster-Ramer, pages 65-86. Amsterdam, John Benjamins Publishing Company

Pinkham, Jessie. 1996. Grammar Sharing between English and French. In *Proceedings of the NLP-IA conference*, Moncton, Canada, June 4-6, 1996.

Russell, Graham. 1985. A GPS Grammar for German Word Order. In *Kontextfreie Syntaxen und verwandte Systeme*. ed. Ursula Klenk. pages 19-32. Tübingen, Niemeyer.

Uszkoreit, Hans Jürgen. 1984. *Word Order and Constituent Structure in German*. Dissertation, University of Texas at Austin.

Uszkoreit, Hans Jürgen. 1982. German Word Order in GPSG. In *Proceedings of the First West Coast Conference on Formal Linguistics*. ed. Flickinger, D., Macken, M. Wiegand, N. pages 137-148.

Volk, Martin. 1988. *Parsing German with GPSG: The Problem of Separable-Prefix Verbs*. Master's thesis, University of Georgia.

Weinrich, Harald. 1993. *Textgrammatik der deutschen Sprache*. Mannheim, Dudenverlag.

Wunderlich, Dieter. 1983. On the Compositionality of German Prefix Verbs. In *Meaning, Use, and Interpretation of Language*. ed. Bäuerle,R., Schwarze, C., Stechow, A. von. pages 452-465. Berlin, Walter de Gruyter.