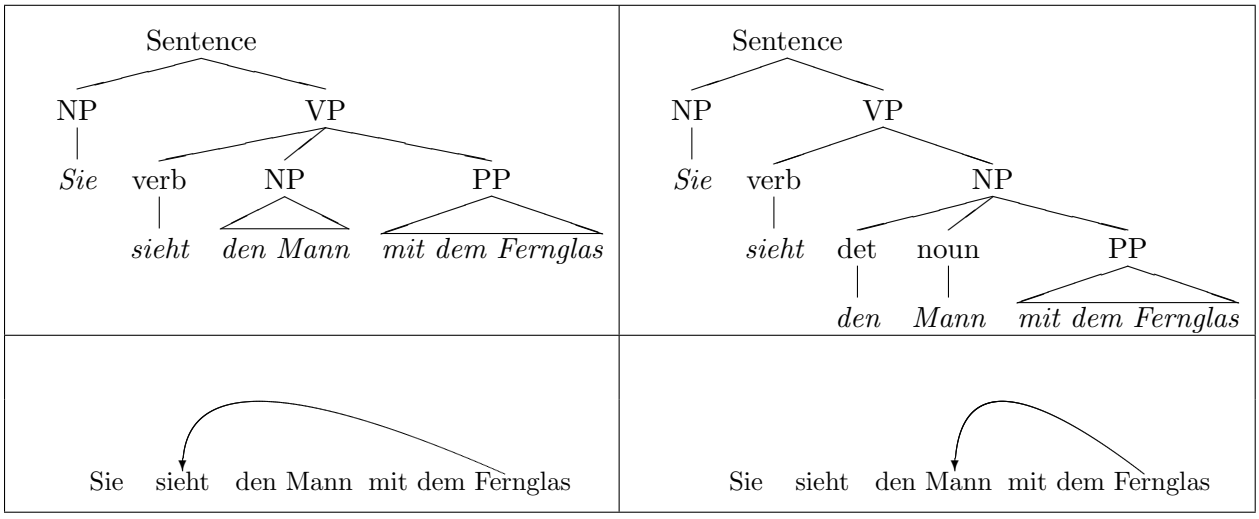
The Automatic Resolution of Prepositional Phrase - Attachment Ambiguities in German

Martin Volk
University of Zurich
Seminar of Computational Linguistics
Winterthurerstr. 190
CH-8057 Zurich
volk@ifi.unizh.ch

Habilitationsschrift
submitted to the
University of Zurich
Faculty of Arts

April 14, 2002
(version 1.1 with minor corrections)



Contents

Acknowledgements

Numerous people have influenced and supported my work. I am particularly indebted to Michael Hess who has provided an excellent work environment at the University of Zurich. He has offered guidance when I needed it, and, even more important, he granted me support and freedom to explore my own ideas.

Heartfelt thanks also go to my office mate Simon Clematide, who has given valuable advice on technical and linguistic matters in so many puzzling situations. His broad knowledge, even temper and his humour made it a joy to work with him.

This work has profited greatly from the Computational Linguistics students at the University of Zurich who I could lure into one branch or another of the project. Thanks go to Toni Arnold who has written an exceptional Little University Information System. Jeannette Roth worked on product name recognition, Gaudenz Lügstenmann on the clause boundary detector, Stefan Höfler on the recognition of temporal expressions, and Julian Käser on lemmatization. Dominic A. Merz wrote the first NP/PP chunker. Marianne Puliafito, Carola Kühnlein and Charlotte Merz annotated thousands of sentences with syntactic structures.

Charlotte Merz was my student assistant in the final phase of the project. She has delved into areas as diverse as bibligraphical searches, LaTeX formatting, hunting for example sentences, and proof-reading the book. She has been of immense help.

Special thanks also to Hagen Langer (then at the University of Osnabrück) and Stephan Mehl (then at the University of Duisburg) who in the early phases of the project joined to form an inspiring team to start investigating PP attachments.

Special thanks to Maeve Olohan (UMIST) for her help in correcting and improving my English and to Gerold Schneider (University of Geneva), Rolf Schwitter (Macquarie University, Sydney) and Andreas Wagner (Universität Tübingen) who have provided valuable comments on earlier versions of this book.

While I believe that all those mentioned have contributed to an improved final manuscript, none is, of course, responsible for remaining weaknesses.

I also would like to acknowledge all who have shared programs and resources. Special thanks to Thorsten Brants and Oliver Plaehn (University of Saarbrücken) for the Annotate treebanking tool and the NEGRA treebank. Annotate is certainly one of the most useful tools for Computational Linguistics in the last decade.

Thanks to Brigitte Krenn for the list of German support verb units, to Hagen Langer for the list of person names, and to Helmut Schmid for the TreeTagger. Thanks to Erhard Hinrichs, Andreas Wagner and the GermaNet team at the University of Tübingen for making the GermaNet thesaurus available to us.

A project like this is impossible without administrative and personal support. Thanks to Corinne Maurer and Lotti Kuendig, the excellent departmental secretaries for supporting me in administrative and personal matters. Thanks also to Beat Rageth and Rico Solca for providing a stable computer network and for technical support. I am also indebted to the Swiss National Fund for its financial support under grant 12-54106.98.

I wish to thank my sisters Anke and Birgit for discussions and encouragement and their friendship. Thanks to my father who has instilled in me a sense of investigation and commitment when I was young. I wish I could return some of this to him now.

And finally, heartfelt thanks to my wife Bettina Imgrund for her loving support. She has pushed me on when I was tempted to stop and slowed me down when I ran too fast. My apologies to her for many lone weekends.

This book has been written in English so that the methods are accessible to the research community. But in the application of the methods it focuses on German. The book contains numerous German example sentences to illustrate the linguistic phenomena. Most of the examples were extracted from the Computer-Zeitung corpus (discussed in detail in section 3.1). I assume that the reader has a basic knowledge of German in order to understand the example sentences. I therefore present the sentences without English glossing.

Zurich, April 14, 2002 Martin Volk

Abstract

Any computer system for natural language processing has to struggle with the problem of ambiguities. If the system is meant to extract precise information from a text, these ambiguities must be resolved. One of the most frequent ambiguities arises from the attachment of prepositional phrases (PPs). A PP that follows a noun can be attached to the noun or to the verb. In this book we propose a method to resolve such ambiguties in German sentences based on cooccurrence values derived from a shallow parsed corpus.

Corpus processing is therefore an important preliminary step. We introduce the modules for proper name recognition and classification, Part-of-Speech tagging, lemmatization, phrase chunking, and clause boundary detection. We processed a corpus of more than 5 million words from the Computer-Zeitung, a weekly computer science newspaper. All information compiled through corpus processing is annotated to the corpus.

In addition to the training corpus, we prepared a 3000 sentence test corpus with manually annotated syntax trees. From this treebank we extracted over 4000 test cases with ambiguously positioned PPs for the evaluation of the disambiguation method. We also extracted test cases from the NEGRA treebank in order to check the domain dependency of the method.

The disambiguation method is based on the idea that a frequent cooccurrence of two words in a corpus indicates binding strength. In particular, we measure the cooccurrence strength between nouns (N) and prepositions (P) and on the other hand between verbs (V) and prepositions. The competing cooccurrence values of N+P versus V+P are compared to decide whether to attach a prepositional phrase (PP) to the noun or to the verb. A variable word order language like German poses special problems for determining the cooccurrence value between verb and preposition since the verb may occur at different positions in a sentence. We tackle this problem with the help of a clause boundary detector to delimit the verb's access range.

Still, the cooccurrence values for V+P are much stronger than for N+P. We need to counterbalance this inequality with a noun factor which is computed from the general tendency of all prepositions to attach to verbs rather than to nouns. It is shown that this noun factor leads to the optimal attachment accuracy.

The method for determining the cooccurrence values is gradually refined by distinguishing sure and possible attachments, different verb readings, idiomatic and non-idiomatic usage, deverbal versus regular nouns, as well as the head noun from the prepositional phrase. In parallel we increase the coverage of the method by using various clustering techniques: lemmatization, core of compounds, proper name classes and the GermaNet thesaurus.

In order to evaluate the method we used the two test sets. We also varied the training corpus to determine its influence on the cooccurrence values. As the ultimate corpus, we tried cooccurrence frequencies from the WWW.

Finally, we compared our method to another unsupervised method and to two supervised methods for PP attachment disambiguation. We show that intertwining our cooccurrence-based method with the supervised Back-off model leads to the best results: 81% correct attachments for the Computer-Zeitung test set.

Chapter 1

Introduction

In recent years vast amounts of texts in machine-readable form have become available through the internet and on mass storage devices (such as CD-ROMs or DVDs). The texts represent a large accumulation of human knowledge. However, the appropriate information to a given question can only be found with the help of sophisticated computer tools. Our central goal is the improvement of retrieval tools with linguistic means so that a user querying a collection of textual data in natural language - in our case German - is guided to the answer that best fits her needs. Our prototype system is described in [Arnold et al. 2001]. Similar systems for English are FAQfinder [Burke et al. 1997] and ExtrAns [Aliod et al. 1998].

Nowadays, information retrieval is mostly organized as document retrieval. The relevance of a document to a given query is computed via a vector of mathematically describable properties (cf. [Schäuble 1997]). We want to move from document retrieval to answer extraction. In answer extraction we are not only interested in the relevant documents but also in the precise location of the relevant information unit (typically a sentence or a short passage) within a document. This requires higher retrieval precision which, we believe, can only be achieved by combining the information retrieval relevance model with a linguistic model. To increase retrieval precision we use linguistic analysis methods over the textual data. These methods include morphological analysis, part-of-speech tagging, and syntactic parsing as well as semantic analysis. We will briefly survey the relevant natural language processing modules and point to their limitations.

1. As an early step in analysis, the words of a natural language text must be **morphologically analysed**. Inflectional endings and stem alterations must be recognized, compounded and derived word forms segmented, and the appropriate base form, the lemma, must be determined. Morphological analysis is especially important for German due to its strong inflectional and compounding system. Such morphology systems are now available (e.g. Gertwol [Lingsoft-Oy 1994] or Word Manager [Domenig and ten Hacken 1992]). These systems work solely on the given word forms. They do not take the words' contexts into account.

2. Complementary, a tagger assigns **part-of-speech tags** to the words in a sentence in accordance with the given sentence context. This enables a first line of word sense disambiguation. If a word is homographic between different parts-of-speech, inappropriate readings can be eliminated. For instance, the tagger can determine if the German word

Junge is used as adjective or noun. Of course, part-of-speech tags do not help in disambiguation if alternative readings belong to the same word class. Current part-of-speech taggers work with context rules or context statistics. They achieve 95% to 97% accuracy (cf. [Volk and Schneider 1998]).

3. The ultimate goal of syntactic analysis is the identification of a sentence's structure. State-of-the-art **parsers** suffer from two problems. On the one hand the parser often cannot find a complete sentence structure due to unknown words or complex grammatical phenomena. Many systems then back-off to partial structures such as clauses or phrases (such as noun phrases (NPs), adverbial phrases or prepositional phrases (PPs)). If a bottom-up chart parser is used, such phrases are often in the chart even if the sentence cannot be completely parsed [Volk 1996b].

   On the other hand the parser often cannot decide between alternatives and produces a multitude of sentence structures corresponding to different interpretations of the sentence. This is often due to a lack of semantic and general world knowledge. Recently, statistical models have been employed to alleviate this problem [Abney 1997]. Parsing with probabilistic grammars helps to rank competing sentence structures [Langer 1999].

4. Finally, syntactic structures need to be mapped into **semantic representations** (logical formulae). During answer extraction this representation allows to match a query to the processed documents.

The two parsing problems (unknown elements and ambiguities) make the sentence analysis task very hard. We believe that only a combination of rule-based and statistical methods will lead to a robust and wide coverage parsing system. Towards this goal we have investigated the attachment of prepositional phrases in German sentences. Prepositional phrases pose a major source of syntactic ambiguity when parsing German sentences. A linguistic unit is ambiguous if the computer (or the human) assigns more than one interpretation to it given its knowledge base.

A more formal definition of ambiguity pointing in the same direction is given by [Schütze 1997] (p. 2):

> A surface form is *ambiguous* with respect to a linguistic process p if it has several process-specific representations and the outcome of p depends on which of these representations is selected. The selection of a process-specific representation in context is called *disambiguation* or *ambiguity resolution*.

We would like to stress that ambiguity is relative to the level of knowledge. A sentence that is ambiguous for the computer is often not ambiguous for the human since the human brain has access to especially adapted knowledge. The goal of research in Computational Linguistics is to enrich the computer's knowledge so that its performance approximates human understanding of language.

From a computational perspective, ambiguities are pervasive in natural language. They occur on all levels.

Word level ambiguities comprise homographs (*Schloss, Ton, Montage*) and homophons (*Meer* vs. *mehr*) on the level of base forms or inflected forms (*gehört* can be a form

of the verb *hören* or *gehören*). They also comprise inflectional ambiguities (*Häuser* can be nominative, genitive or dative plural) and compound segmentation ambiguities (*Zwei#fels#fall* vs. *Zweifel-s#fall*).

Sentence level ambiguities include syntactic and semantic ambiguities. A frequent syntactic ambiguity in German concerns the mix-up of nominative and accusative NPs especially for feminine and neuter nouns (*Das Gras frisst die Kuh*). The ordering preference of subject < object is a hint for disambiguation but it can be overridden by topical constraints or emphatic usage leading to the ambiguity. A second frequent syntactic ambiguity concerns coordination. The scope of the coordinated elements can often be inferred only with knowledge of the situation. In example 1.1 the negation particle *nicht* modifies either the adjective *starr* or both *starr* and *unabhängig*. In 1.2 the *was*-relative clause modifies either only the last verb or both coordinated verbs. In 1.3 the adverb *neu* modifies one or two verbs.

(1.1) *Das erfaßte Wissen wird also* ***nicht starr und unabhängig von realen Fakten*** *verarbeitet ...*

(1.2) *Da nicht ständig jemand neben mir stand, der* ***angab und aufpaßte****, was zu tun sei ...*

(1.3) *... wenn man das von der Bedeutung für den Menschen her* ***neu interpretiere und formalisiere****.*

The third frequent syntactic ambiguity concerns the attachment of prepositional phrases which is exemplified on the title page and will be dealt with in this book.

In some sense all syntactic ambiguities are also semantic ambiguities. They represent different meaning variants. True semantic ambiguities arise if the syntactic structure is evident but meaning variants still persist. This often happens with quantifier scoping. In example 1.4 the syntactic structure is clear. But the quantifiers *alle* and *einer* can be interpreted in a collective reading (all take-overs depend on one and the same strategy) or a distributive reading (all take-overs depend on different strategies).

(1.4) *Alle Übernahmen und Partnerschaften basieren auf einer Strategie des qualitativen Wachstums.*

Text level ambiguities involve inter-sentence relations such as pronominal references. If, for example, two masculine nouns are introduced in a discourse, the pronoun *er* can refer to either of them.

(1.5) *Neben Corollary-Präsident George White steht Mitbegründer Alan Slipson:* ***Er*** *ist der Unix-Experte, der heute die Software-Entwicklung bei Corollary leitet.*

(1.6) *Peter Scheer (31) leitet zusammen mit Andreas S. Müller die Beratung der Münchner ASM Werbeagentur GmbH. Vorher war* ***er*** *für die internationalen Marcom-Aktivitäten von Softlab, München, verantwortlich.*

1.1 Prepositions and their Kin

Prepositions in German are a class of words relating linguistic elements to each other with respect to a semantic dimension such as local, temporal, causal or modal. They do not inflect and cannot function by themselves as a sentence unit (cf. [Bußmann 1990]). But, unlike other function words, a preposition governs the grammatical case of its argument (genitive, dative or accusative). As the name indicates, a preposition is positioned in front of its argument. Typical German prepositions are *an, für, in, mit, zwischen*.

Prepositions are among the central word classes in modern grammatical theories such as Generalized Phrase Structure Grammar (GPSG) and Head-Driven Phrase Structure Grammar (HPSG). In GPSG, prepositions together with nouns, verbs and adjectives are defined by the basic features N and V (cf. [Gazdar et al. 1985] p. 20). In HPSG these four word classes plus relativizers are in the same class of the sort hierarchy as the partition of "substantive" objects (cf. [Pollard and Sag 1994] p. 396).

Prepositions are considered to be a closed word class. Nevertheless it is difficult to determine the exact number of German prepositions. [Schröder 1990] speaks of "more than 200 prepositions", but his "Lexikon deutscher Präpositionen" lists only 110 of them. In this dictionary all entries are marked with their case requirement and their semantic features. For instance, *ohne* requires the accusative and is marked with the semantic functions instrumental, modal, conditional and part-of.[1]

The lexical database CELEX [Baayen et al. 1995] contains 108 German prepositions with frequency counts derived from corpora of the "Institut für deutsche Sprache". This results in the arbitrary inclusion of *nördlich, nordöstlich, südlich* while *östlich* and *westlich* are missing.

Searching through 5.5 million words of our tagged computer magazine corpus we found around 540,000 preposition tokens corresponding to 100 preposition types.[2] These counts do not include contracted prepositions. The 20 most frequent prepositions are listed in the following table, the complete list can be found in appendix A.

[1] See also [Klaus 1999] for a detailed comparison of the range of German prepositions as listed in a number of recent grammar books.

[2] These figures are based on automatically assigned part-of-speech tags. If the tagger systematically mistagged a preposition, the counting procedure does not find it. In the course of the project we realized that this happened to the prepositions *a*, *via* and *voller* as used in the following example sentences.

(1.7) *Derselbe Service in der Regionalzone (bis zu 50 Kilometern) kostet 23 Pfennig* ***a 60 Sekunden****.*

(1.8) *Master und Host kommunizieren* ***via IPX****.*

(1.9) *Windows steckt* ***voller eigener Fehler****.*

rank	preposition	frequency		rank	preposition	frequency
1	*in*	84662		11	*aus*	13949
2	*von*	71685		12	*durch*	12038
3	*für*	64413		13	*bis*	11253
4	*mit*	61352		14	*unter*	10129
5	*auf*	49752		15	*um*	9880
6	*bei*	27218		16	*vor*	9852
7	*über*	19182		17	*zwischen*	5079
8	*an*	18256		18	*seit*	4194
9	*zu*	17672		19	*pro*	4175
10	*nach*	15298		20	*ohne*	3007

An early frequency count for German by [Meier 1964] lists 18 prepositions among the 100 most frequent word forms. 17 out of these 18 prepositions are also in our top-20 list. Only *gegen* is missing which is on rank 23 in our corpus. This means that the usage of the most frequent prepositions is stable over corpora and time.

All frequent prepositions in German have some homograph serving as

- separable verb prefix (e.g. *ab, auf, mit, zu*),
- clause conjunction (e.g. *bis, um*)[3],
- adverb (e.g. *auf, für, über*) in often idiomatic expressions (e.g. *auf und davon, über und über*),
- infinitive marker (*zu*),
- proper name component (*von*), or
- predicative adjective (e.g. *an, auf, aus, in, zu* as in *Die Maschine ist an/aus. Die Tür ist auf/zu.*).

The most frequent homographic functions are separable verb prefix and conjunction. Fortunately, these functions are clearly marked by their position within the clause. A clause conjunction usually occurs at the beginning of a clause, and a separated verb prefix mostly occurs at the end of a clause (*rechte Satzklammer*). A part-of-speech tagger can therefore disambiguate these cases.[4]

Typical (i.e. frequent) prepositions are monomorphemic words (e.g. *an, auf, für, in, mit, über, von, zwischen*). Many of the less frequent prepositions are derived or complex. They have become prepositions over time and still show traces of their origin. They are derived from other parts-of-speech such as

- nouns (e.g. *angesichts, zwecks*),
- adjectives (e.g. *fern, unweit*),

[3] [Jaworska 1999] (p. 306) argues that "clause-introducing preposition-like elements are indeed prepositions".

[4] Note the high degree of ambiguity for *zu* which can be a preposition *zu ihm*, a separated verb prefix *sie sieht ihm zu*, the infinitive marker *ihn zu sehen*, a predicative adjective *das Fenster ist zu*, an adjectival or adverb marker *zu gross, zu sehr*, or the ordinal number marker *sie kommen zu zweit*.

- participle forms of verbs (e.g. *entsprechend, während; ungeachtet*), or
- lexicalized prepositional phrases (e.g. *anhand, aufgrund, zugunsten*).

Prepositions typically do not allow compounding. It is generally not possible to form a new preposition by concatenation of prepositions. The two exceptions are *gegenüber* and *mitsamt*. Other concatenated prepositions have led to adverbs like *inzwischen, mitunter, zwischendurch*.

[Helbig and Buscha 1998] call the monomorphemic prepositions **primary prepositions** and the derived prepositions **secondary prepositions**. This distinction is based on the fact that only primary prepositions form prepositional objects, pronominal adverbs (cf. section 1.1.2) and prepositional reciprocal pronouns (cf. section 1.1.3).

In addition, this distinction corresponds to different case requirements. Governing grammatical case is typical for German prepositions. The primary prepositions govern accusative (*durch, für, gegen, ohne, um*) or dative (*aus, bei, mit, nach, von, zu*) or both (*an, auf, hinter, in, neben, über, unter, vor, zwischen*). Most of the secondary prepositions govern genitive (*angesichts, bezüglich, dank*). Some prepositions (most notably *während*) are in the process of changing from genitive to dative. Some prepositions do not show overt case requirements (*je, pro, per*; cf. [Schaeder 1998]).

Some prepositions show other idiosyncracies. The preposition *bis* often takes another preposition (*in, um, zu* as in 1.10) or combines with the particle *hin* and a preposition (as in 1.11). The preposition *zwischen* is special in that it requires a plural argument (as in 1.12), often realized as a coordination of NPs (as in 1.13).

(1.10) *Portables mit 486er-Prozessor werden* ***bis zu 20 Prozent*** *billiger.*

(1.11) *... und berücksichtigt auch Daten und Datentypen* ***bis hin zu Arrays*** *oder den Records im VAX-Fortran.*

(1.12) *Die Verbindungstopologie* ***zwischen den Prozessoren*** *läßt sich als dreidimensionaler Torus darstellen.*

(1.13) *Durch Microsoft Access müssen sich die Anwender nicht mehr länger* ***zwischen Bedienerfreundlichkeit und Leistung*** *entscheiden.*

1.1.1 Contracted Prepositions

Certain primary prepositions combine with a determiner to contracted forms. This process is restricted to *an, auf, ausser, bei, durch, für, hinter, in, neben, über, um, unter, von, vor, zu*. Our corpus contains about 89,000 tokens that are tagged as contracted prepositions (14% of all preposition tokens). The contracted form stands usually for a combination of the preposition with the definite determiner *der, das, dem*.[5] If a contracted preposition is available, it will not always substitute the separate usage of preposition and determiner but rather compete with it. For example, the contracted preposition *beim* (example 1.14) is used in its separate forms with a definite determiner in 1.15. Example 1.16 shows a sentence with *bei* plus an indefinite determiner. But the usage of the contracted preposition would also be possible (*Beim Ausfall*

[5] [Helbig and Buscha 1998] (p. 388) mention that it is possible to build contracted forms with the determiner *den*: *hintern, übern, untern*. But these forms are very colloquial and do not occur in our corpus.

einer gesamten CPU), and we claim that it would not change the meaning. This indicates that sometimes the contracted preposition might stand for a combination of the preposition with the indefinite determiner *einer, ein, einem.*

(1.14) *Detlef Knott, Vertriebsleiter* ***beim Softwarehaus Computenz GmbH*** *...*

(1.15) *Eine adäquate Lösung fand sich* ***bei dem indischen Softwarehaus CMC****, das ein Mach Plan-System bereits ... in die Praxis umgesetzt hatte:*

(1.16) ***Bei einem Ausfall*** *einer gesamten CPU springt der Backup-Rechner für das ausgefallene System in die Bresche.*

For the most frequent contracted prepositions (*im, zum, zur, vom, am, beim, ins*), a separate usage indicates a special stress on the determiner. The definite determiner then almost resembles a demonstrative pronoun.

The less frequent contracted prepositions sound colloquial (e.g. *aufs, überm*). The frequency overview in appendix B shows that these contracted prepositions are more often used in separated than in contracted form in our newspaper corpus. [Helbig and Buscha 1998] (p. 388) claim that *ans* is unmarked ("völlig normalsprachlich"), but our frequency counts contradict this claim. In our newspaper corpus *ans* is used 199 times but *an das* occurs 611 times. This makes *ans* the borderline case between the clearly unmarked contracted prepositions and the ones that are clearly marked as colloquial in written German.

Some contracted prepositions are required by specific constructions in standard German. Among these are (according to [Drosdowski 1995]):

- *am* with the superlative: *Sie tanzt am besten.*
- *am* or *beim* with infinitives used as nouns: *Er ist am Arbeiten. Er ist beim Kochen.*
- *am* as a fixed part of date specifications: *Er kommt am 15. Mai.*

1.1.2 Pronominal Adverbs

In another morphological process primary prepositions can be embedded into pronominal adverbs. A pronominal adverb is a combination of a particle (*da(r), hier, wo(r)*) and a preposition (e.g. *daran, dafür, hierunter, woran, wofür*).[6] In colloquial German pronominal adverbs with *dar* are often reduced to *dr*-forms (e.g. *dran, drin, drunter*), and we found some dozen occurrences of these in our corpus.

Pronominal adverbs are used to substitute and refer to a prepositional phrase. The forms with *da(r)* are often used in place holder constructions, where they serve as (mostly cataphoric) pointers to various types of clauses.

(1.17) **Cataphoric pointer to a *daß*-clause:** *Es sollte* ***darauf*** *geachtet werden, daß auch die Hersteller selbst vergleichbar sind.*

[6] This is why pronominal adverbs are sometimes called prepositional adverbs (e.g. in [Zifonun et al. 1997]) or even prepositional pronouns (e.g. in [Langer 1999]).

(1.18) **Cataphoric pointer to an *ob*-clause:** *Die Qualitätssicherung von Dokumentationen richtet sich bei dem vorrangig zu betrachtenden Vollständigkeitsaspekt **darauf**, ob Aufbau und Umfang im vereinbarten Rahmen gegeben sind.*

(1.19) **Cataphoric pointer to an infinitive clause:** *Die Praxis der Software-Nutzungsverträge zielt **darauf** ab, den mitunter gravierenden Wandel in den DV-Strukturen eines Unternehmens nicht zu behindern ...*

(1.20) **Cataphoric pointer to a relative clause:** *Im Grunde kommt es **darauf** an, was dann noch alles an Systemsoftware hinzukommt.*

(1.21) **Anaphoric pointer to a noun phrase:** *Vielmehr können sich /36-Kunden, die den Umstieg erst später wagen wollen, mit der RPG II 1/2 **darauf** vorbereiten.*

The following table shows the most frequent pronominal adverbs in our computer magazine corpus (the complete list can be found in appendix C):

rank	pronominal adverb	frequency		rank	pronominal adverb	frequency
1	*damit*	6333		11	*wobei*	687
2	*dabei*	5861		12	*darin*	685
3	*dazu*	3099		13	*darunter*	587
4	*dafür*	2410		14	*danach*	531
5	*darüber*	1752		15	*daraus*	432
6	*davon*	1713		16	*hierbei*	381
7	*dagegen*	1397		17	*darum*	367
8	*dadurch*	1385		18	*hierzu*	348
9	*darauf*	1267		19	*daneben*	331
10	*daran*	737		20	*hierfür*	309

It is striking that the frequency order of this list does not correspond to the frequency order of the preposition list. The most frequent prepositions *in* and *von* are represented only on ranks 13 and 6 in the pronominal adverb list. Obviously, pronominal adverbs behave differently from prepositions. Pronominal adverbs can only substitute prepositional complements (as in 1.22) with the additional restriction that the PP noun is not an animate object (as in 1.23). Pronominal adverbs cannot substitute adjuncts. Those will be substituted by adverbs that represent their local (*hier, dort*; see 1.24) or temporal character (*damals, dann*).

(1.22) *Die Wasserchemiker warten **auf solche Geräte / darauf** ...*

(1.23) *Absolut neue Herausforderungen warten **auf die Informatiker / *darauf / auf sie** beim Stichwort "genetische Algorithmen" ...*

(1.24) *Daher wird **auf dem Börsenparkett / *darauf / dort** heftig über eine mögliche Übernahme spekuliert.*

We restrict pronominal adverbs to combinations of the above-mentioned particles (*da, hier, wo*) with prepositions. Sometimes other combinations with prepositions are included as well. The STTS [Schiller et al. 1995] includes combinations with *des* and *dem*.

- *deswegen*; *deshalb*[7]
- *ausserdem, trotzdem*; also with postpositions: *demgemäss, demzufolge, demgegenüber*

On the other hand the STTS separates the combinations with *wo* into the class of adverbial interrogative pronouns. This classification is appropriate for the purpose of part-of-speech tagging. The distributional properties of *wo*-combinations are more similar to other interrogative pronouns like *wann* than to regular pronominal adverbs. But for the purpose of investigating prepositional attachments, we will concentrate on those pronominal adverbs that behave most similar to PPs.

In this context we need to mention preposition stranding, a phenomenon that is ungrammatical in standard German but acceptable in northern German dialects and some southern German dialects. It is the splitting of the pronominal adverb into discontinuous elements (as in 1.25).[8]

(1.25) ***Da*** *weiss ich nichts* ***von****.*

1.1.3 Reciprocal Pronouns

Yet another disguise of primary prepositions is their combination with the reciprocal pronoun *einander*.[9] The preposition and the pronoun constitute an orthographic unit which substitutes a prepositional phrase. Reciprocal pronouns are a powerful abbreviatory device. The reciprocal pronoun in a schema like *A und B P-einander* stands for *A P B und B P A*. For instance, *A und B spielten miteinander* stands for *A spielte mit B und B mit A*.

A reciprocal pronoun may modify a noun (as in example 1.26) or a verb (as in 1.27). Most reciprocal pronouns can also be used as nouns (see 1.28); some are nominalized so often that they can be regarded as lexicalized (e.g. *Durcheinander, Miteinander, Nebeneinander*).

(1.26) *... und damit eine Modellierung von Objekten der realen (Programmier-) Welt und ihrer Beziehungen* ***untereinander*** *darstellen können.*

(1.27) *Ansonsten dürfen die Behörden nur die vom Verkäufer und vom Erwerber eingegangenen Informationen* ***miteinander*** *vergleichen.*

(1.28) *Chaos ist in der derzeitigen Panik- und Krisenstimmung nicht nur ein Wort für wildes* ***Durcheinander****, sondern ...*

In our corpus we found 16 different reciprocal pronouns with prepositions. The frequency ranking is listed in appendix D. It is striking that some of the P+*einander* combinations are more frequent than the reciprocal pronoun itself.

---

[7] Of course, *halb* is not a preposition but rather a preposition building morpheme: *innerhalb, ausserhalb; oberhalb, unterhalb.*

[8] The phenomenon was discussed in the LINGUIST list as contribution 11.2688, Dec. 12, 2000.

[9] Sometimes the word *gegenseitig* is also considered to be a reciprocal pronoun. Since the preposition *gegen* in this form cannot be substituted by any other preposition, we take this to be a special form and do not discuss it here.

1.1.4 Prepositions in Other Morphological Processes

Some prepositions are subject to conversion processes. Their homographic forms belong to other word classes. In particular, there are P + conjunction + P sequences (*ab und zu, nach wie vor, über und über*) that are idiomized and function as adverbials (cf. example 1.29). They are derived from prepositions but they do not form PPs. As long as they are symmetrical, they can easily be recognized. All others need to be listed in a lexicon so that they are not confused with coordinated prepositions.

Some such coordinated sequences must be treated as N + conjunction + N (*das Auf und Ab, das Für und Wider*; cf. 1.30) and are also outside the scope of our research. Finally, there are few prepositions that allow a direct conversion to a noun such as *Gegenüber* in 1.31.

(1.29) *Eine Vielzahl von Straßennamensänderungen wird **nach und nach** noch erfolgen.*

(1.30) *Nachdem sie das **Für und Wider** gehört haben, können die Zuschauer ihre Meinung ... kundtun.*

(1.31) *Verhandlungen enden häufig in der Sackgasse, weil kein Verhandlungspartner sich zuvor Gedanken über die Situation seines **Gegenübers** gemacht hat.*

Prepositions are often used to form adverbs. We have already mentioned that P+P compounds often result in adverbs (e.g. *durchaus, nebenan, überaus, vorbei*). Even more productive is the combination with the particles *hin* and *her*. They are used as suffix *nachher, vorher; mithin, ohnehin* or as prefix *herauf, herüber; hinauf, hinüber*. These adverbs are sometimes called prepositional adverbs (cf. [Fleischer and Barz 1995]). They can also combine with pronominal adverbs (*daraufhin*).

In addition, there is a limited number of preposition combinations with nouns (*bergauf, kopfüber, tagsüber*) and adjectives (*hellauf, rundum, weitaus*) that function as adverbs if the preposition is the last element. Sometimes the preposition is the first element, which leads to a derivation within the same word class (*Ausfahrt, Nachteil, Vorteil, Nebensache*).

Finally, most of the verbal prefixes can be seen as preposition + verb combinations. Some of them function only as separable prefix (*ab, an, auf, aus, bei, nach, vor, zu*), others can be separable or inseparable (*durch, über, um, unter*). Note that the meaning contribution of the preposition to the verb varies as much as the semantic functions of the preposition. Consider for example the preposition *über* in *überblicken* (to survey; literally: to view over), *übersehen* (to overlook, to disregard, to realize; literally: to look over or to look away), and *übertreffen* (to surpass; literally: to aim better).

The preposition *mit* can also serve as a separable prefix (see 1.32), but it shows an idiosyncratic behaviour when it occurs with prefixed verbs (be they separable as in 1.33 or inseparable as in 1.34).[10] In this case *mit* does not combine with the verb but rather functions as an adverb.

(1.32) *Die künftigen Bildschirmbenutzer wirken an der Konzeption nicht **mit**.*

(1.33) *Schröder ist seit 22 Jahren für die GSI-Gruppe tätig und hat die deutsche Dependance **mit** aufgebaut.*

[10] A detailed study of the preposition *mit* can be found in [Springer 1987].

(1.34) *Die Hardwarebasis soll noch erweitert werden und andere Unix-Plattformen* ***mit*** *einbeziehen.*

This analysis is shared by [Zifonun et al. 1997] (p. 2146). *mit* can function like a PP-specifying adverb (see 1.35). And in example 1.36 it looks more like a stranded separated prefix (cf. *an Bord mitzunehmen*). [Zifonun et al. 1997] note that the distribution of *mit* differs from full adverbs. It is rather similar to the adverbial particles *hin* and *her*. All of them can only be moved to the *Vorfeld* in combination with the constituent that they modify (cf. examples 1.37 and 1.38).

(1.35) *... und deren Werte* ***mit*** *in die DIN 57848 für Bildschirme eingingen.*

(1.36) *... geht man dazu über, Subunternehmer* ***mit*** *an Bord zu nehmen.*

(1.37) ***Mit*** *auf der Produktliste standen noch der Netware Lanalyzer Agent 1.0, ...*

(1.38) ****Mit*** *standen noch der Netware Lanalyzer Agent 1.0 auf der Produktliste, ...*

1.1.5 Postpositions and Circumpositions

In terms of language typology German is regarded as a preposition language while others, like Japanese or Turkish, are postposition languages. But in German there are also rare cases of postpositions and circumpositions. Circumpositions are discontinuous elements consisting of a preposition and a "postpositional element". This postpositional element can be an adverb (as in example 1.39) or a preposition (as in example 1.40). Even pronominal adverbs can take postpositional elements to form circumpositional phrases (see example 1.41).

The case of postpositions is similar. There are few true postpositions (e.g. *halber, zufolge*; see 1.42), but others are homographic with prepositions (see examples 1.43 and 1.44).

(1.39) *Beispielsweise können Werte und Grafiken in ein Textdokument exportiert oder Messungen* ***aus einer Datenbank heraus*** *parametriert und gestartet werden.*

(1.40) *... oder* ***vom Programm aus*** *direkt gestartet werden.*

(1.41) *Die Messegesellschaft hat* ***darüber hinaus*** *globale Netztechnologien und verschiedene Endgeräte in dieser Halle angesiedelt.*

(1.42) *Über die Systems in München werden* ***Softbank-Insidern zufolge*** *Gespräche geführt.*

(1.43) *Das größte Potential für die Branche steckt* ***seiner Ansicht nach*** *in der Verknüpfung von Firmen.*

(1.44) *Und das bleibt auch* ***die Woche über*** *so.*

Because of these homographs the correct part-of-speech tagging for postpositions and postpositional elements of circumpositions is a major problem. It works correctly if the subsequent context is prohibitive for the preposition reading (e.g. when the postposition is followed by a verb). But in other pre-post ambiguities the tagger often fails since the preposition reading is so dominant for these words. Special correction rules will be needed.

1.2 Prepositional Phrases

Usually, a preposition introduces a prepositional phrase (PP). A PP is a phrasal constituent typically consisting of a preposition and a noun phrase (NP) or a pronominal term.[11] The pronominal term is either a pronoun or a subclass of adjectives and adverbs that can function as an adverbial. [Langer 1999] even creates a special word class called "prepositional complement particles" since there are some words that occur only in this position (e.g. *jeher* in *seit jeher*).

A PP can be realized with the following internal constituents:

preposition + NP	*durch den Garten, mit viel Geld*
contracted prep. + NP (without determiner)	*im Garten, beim alten Fritz*
preposition + pronoun	*auf etwas, durch ihn, mit dem*[12]
preposition + adjective	*auf deutsch, für gut*
preposition + adverb	*bis morgen, von dort*

Within a sentence a PP can take over many functions which is the reason for the PP attachment ambiguities. A PP may serve as:

Prepositional object. In this case the verb subcategorizes for the PP as it does for other complements such as accusative or dative objects. The specific preposition is determined by the verb. Only primary prepositions (like *auf, mit, zu*) are used with prepositional objects. Secondary prepositions like *infolge, anstelle* will only serve in adverbials. According to [Zifonun et al. 1997] (p. 1093) the prepositional complement is third in the usage frequency of complements after subject and accusative object. A detailed discussion of prepositional objects can be found in [Breindl 1989].

(1.45) *Das Kommunikationsprotokoll TCP/IP sorgt* ***für einen reibungslosen Datenfluß*** *in heterogenen Netzwerken.*

(1.46) *Der 56jährige Spitzenmanager will sich nach eigener Aussage nun verstärkt* ***um seine eigenen Interessen*** *kümmern.*

Attribute to a noun. The PP is either a complement or an adjunct of a noun. Prepositional attributes in German are discussed in detail in [Schierholz 2001].

(1.47) *Großen Zuspruch* ***bei EC-Karten-Besitzern*** *erhofft sich die Kreditwirtschaft von der Integration der Telefonkartenfunktion.*

(1.48) *PC-Software versteht sich nicht mehr als Synonym* ***für totale Austauschbarkeit****.*

Attribute to a predicative or attributive adjective. The PP is dependent on an adjective.

(1.49) *Wir können* ***mit dem Geschäft*** *absolut nicht zufrieden sein.*

[11] [Langer 1999] reports that the grammar rule $PP \rightarrow P + NP$ accounts for 78% of all German PPs.

[12] As noted above, the reciprocal pronoun forms an orthographic unit with the determiner. Similarly, the preposition *wegen* combines with personal pronouns: *meinetwegen, seinetwegen, Ihretwegen.*

(1.50) *Das erste Quartal 93 brachte dem Add-on-Board-Hersteller mit 14 Millionen Dollar einen* ***um 53 Prozent*** *höheren Umsatz als im Vorjahreszeitraum.*

Adverbial adjunct. The PP is not necessary for the grammaticality of the sentence. It contains clause-modifying information.

(1.51) *Wir haben das Paket* ***bei Ihnen in der Neuen Rabenstraße*** *gestern* ***um 14.30 Uhr*** *abgeholt.*

Predicative. The PP and the verb *sein* are the predicate of the sentence. Most predicative PPs sound idiomized.

(1.52) *Fast alle sind* ***mit von der Partie****.*

(1.53) *Der Siegeszug des Japan-Chips ist* ***zu Ende****.*

(1.54) *Sind Frauen nach Ihren Erfahrungen bei der Jobsuche im DV-Arbeitsmarkt* ***im Nachteil****?*

[Jaworska 1999] claims that a PP can also function as the subject of a sentence and she quotes the English example 1.55. An analogous German example would be 1.56. [Zifonun et al. 1997] (p. 1331) mention sentences with the expletive *es* subject and a PP (as in 1.57) which often lead to "secondary subjects" (as in 1.58). We think that example 1.56 contains an invisible *es* subject and that the PP is not the subject. Examples like 1.56 are very rare and will not be further explored in this book.

(1.55) ***Between six and seven*** *suits her fine.*

(1.56) ***Um 6 Uhr*** *geht mir gut.*

(1.57) ***Im letzten Herbst*** *war es regnerisch und kalt.*

(1.58) ***Der letzte Herbst*** *war regnerisch und kalt.*

In principle, prepositions can be coordinated even if they govern different grammatical cases. The last preposition in the conjoined sequence will then determine the case of the PP.

(1.59) *Dafür werden Pentium-Prozessoren* ***mit oder ohne*** *den Multimediabefehlssatz MMX ab August im Preis sinken.*

(1.60) *... und LDAP-Operationen* ***mit oder anstelle*** *des DCE Call Directory Services zu nutzen.*

(1.61) *Insellösungen wie CAD- oder Qualitätssicherungsapplikationen laufen oft* ***neben und nicht mit*** *dem PPS-System.*

Some prepositions can also be combined. The most notable example is *bis* which is often used with other prepositions (e.g. *bis am nächsten Freitag, bis um 3 Uhr, bis zu diesem Tag*). But it also works for some other prepositions (e.g. *seit nach dem Krieg*). [Jaworska 1999] describes this phenomenon as a preposition taking a PP argument. There are also combinations with *über* and *unter* like *seit über 20 Jahren, mit über 600 Seiten, für unter*

10.000 Mark, but it is doubtful whether *über* and *unter* function as prepositions in these expressions. We think they should rather be regarded as specifier in the measurement phrase.

Combinations of secondary prepositions with *von* (like *jenseits von Afrika, westlich von Rhein und Mosel*) look similar. But in these combinations the genitive argument (e.g. *westlich des Rheins und der Mosel*) is only substituted by a *von*-PP if the case is not marked by a determiner or an adjective. This is illustrated in the following examples for the preposition *innerhalb*.

(1.62) ***Innerhalb von anderthalb Jahren*** *mauserte sich W. Industries ...*

(1.63) ****Innerhalb anderthalb Jahre*** *mauserte sich W. Industries ...*

(1.64) *Die Software soll* ***innerhalb der nächsten drei Jahre*** *geliefert werden.*

If a PP does not function as object, it can take a specifier. The specifier modulates the adverbial contribution of the PP to the sentence. In example 1.65 the adverb *schon* modifies the temporal PP, and in 1.66 the adverb *fast* relativizes the strict exclusion of *ohne Ausnahme*. It is difficult to automatically recognize such PP specifiers. The adverb might as well modify the verb as in 1.67. [Zifonun et al. 1997] also mention adverbs like *morgens* (cf. 1.68) as post-PP specifiers.

(1.65) *Zum einen will der Telekom-Riese die Unix-Schmiede* ***schon seit 1991*** *an den Mann bringen.*

(1.66) *Gleichzeitig ist sie* ***fast ohne Ausnahme*** *mit Überkapazitäten belastet, ...*

(1.67) *... sind viele der noch vor einem Jahr angebotenen Peer-to-Peer-Produkte fast* ***vom Markt*** *verschwunden.*

(1.68) *Die Abonnenten von Chicago Online können parallel zur gedruckten Ausgabe ihres Blattes* ***ab 8.00 Uhr morgens*** *... nach Artikeln suchen.*

1.2.1 Comparative Phrases

Comparative phrases are borderline cases of PPs. The comparative particles (*als, wie*) function as relation operator in much the same way as a preposition, but they do not determine the grammatical case of the dependent NP.

Comparative phrases can attach to the verb or to a preceding noun. They vary considerably with regard to the meaning relation of their reference element. Examples 1.69 and 1.70 contain noun-attached *als*-phrases with the meaning relation "functioning as". In contrast, example 1.71 contains an *als*-phrase that is the complement to the reflexive verb. The comparative sense is almost lost in this function. Unlike regular PPs, comparative phrases that follow a noun can also be attached to the comparative adjective within the NP. In example 1.72 the *als*-phrase is attached to the adjective phrase *ganz andere* and in 1.73 it complements the indefinite pronoun *mehr*.

(1.69) *Eine zweite befaßt sich mit der Sprache* ***als Steuermedium*** *für PCs.*

(1.70) *... beschreiben die Autoren Architektur, Technologie und Protokolle im FDDI und dessen Einsatz* ***als Backbone****.*

(1.71) *Dafür erweist sich die CD-ROM* ***als höchst flexibles Medium****.*

(1.72) *Speziell die Gestaltung der Interaktivität bedingt ganz andere Qualitäten der Aufbereitung* ***als beispielsweise das Drehen eines Films*** *...*

(1.73) *... und IBM war immer schon mehr eine Geisteshaltung* ***als eine Firma****.*

Similarly, the comparative particle *wie* can attach to nouns, adjectives and verbs. As noun-attached phrase it stands for the meaning relation "as exemplified by" (1.74). As verb- or adjective-attached phrase the relation is "in the same way as" (1.75, 1.76).

(1.74) *Die Folge sind häufige Über- oder Unterzuckerwerte mit akuten Komplikationen* ***wie Bewußtlosigkeit und Vergiftungserscheinungen*** *...*

(1.75) *Juristen beispielsweise könnten die gespeicherten Daten* ***wie ihre herkömmlichen Informationsquellen*** *als Basisinformation für ihre Arbeit verwenden.*

(1.76) *Der Empfänger ist mit einem PIN-Photodetektor ausgestattet und ähnlichen Bauelementen* ***wie der Sender****.*

Sometimes these comparative particles are considered to be conjunctions (cf. [Schaeder 1998] p. 216) which is evident since both of them can introduce subordinate sentences. Since comparative phrases behave differently from regular PPs, we exclude them from the general investigation and discuss them separately in section 4.11.

1.2.2 Frozen PPs

PPs are frequent components of German idioms (as exemplified in 1.77). Within these idioms the PP is (often) frozen in the sense that the lexical items cannot be interchanged without hurting the idiomatic reading. No additional lexemes can intervene, in particular no attributes can be added. We will look at idiomatic PPs in more detail in section 4.6.

(1.77) *Mit einem Datenbankprogramm könnte Lotus* ***zwei Fliegen mit einer Klappe schlagen****:*

Moreover, there are PPs that function similar to prepositions (*mit Blick auf, mit Hilfe, unter dem Druck*). [Schröder 1990] lists 96 PPs of this sort. Most of them are of the two patterns. Either they occur as fixed P+N+P triple (as in 1.78) or with a determiner as P+Det+N+P (as in 1.79).

(1.78) *Dabei modifizieren sie* ***mit Hilfe*** *von Algorithmen die Stärke der Verbindungen zwischen den Knoten.*

(1.79) *Demgegenüber werden die Gewinnmargen ... in diesem Jahr antizyklisch steigen und erst mit Verzögerung* ***unter dem Druck*** *von Open-Systems-Technologien und preiswerteren Hardwarelösungen sinken.*

We therefore searched our corpus for patterns of this sort and manually inspected all sequences that occurred more than 50 times. We added 31 PPs to Schröder's list so that we can employ them in corpus processing (e.g. *nach Ansicht, mit Blick auf*).

More difficult for syntactic analyses are N+P+N sequences in which the same noun is repeated. This pattern is restricted to the prepositions *an, auf, für, nach, über, um*. Our corpus contains 260 patterns (tokens) of this type with *Schritt für Schritt* being by far the most frequent (52 times). Other examples are:

(1.80) *Der Rechner tastet sich **Tag für Tag** in die Zukunft vor.*

(1.81) *Der angeschlagene Multi setzt **Zug um Zug** seine Umstrukturierung fort, ...*

Some of these patterns sound almost idiomatic, especially the ones standing for time expressions like *Stunde um Stunde, Tag für Tag, Jahr für Jahr*. But as can be seen in example 1.82, the pattern is productive and allows to express repetition and duration. Similar to these is the special pattern N+*im*+N to express the contained-in relation (cf. 1.83).

(1.82) *Auf diese Art konnte DEC kurzfristig die Lücke nach unten füllen und beginnt nun, **Maschinchen für Maschinchen** aus der eigenen Entwicklung nachzuschieben.*

(1.83) *Getragen von der Idee, Hierachien abzuflachen, wird das "**Unternehmen im Unternehmen**" konstituiert.*

[Langer 1999] suggests to treat these patterns as NPs with modifying PPs and we will follow this suggestion: e.g. *(NP Stunde PP(für Stunde))*. Since such patterns are rare in comparison to the occurrence frequencies of the involved prepositions, we will leave them in our training corpus but make sure that we do not use them in our test corpus (cf. chapter 3).

1.2.3 Support Verb Units

A support verb unit (*Funktionsverbgefüge*) is a combination of a PP (or NP) and a semantically weak verb (e.g. *in Besitz nehmen*). The support verb unit functions as a full verb and increases the verb's variability to express phases of processes and states (cf. [Krenn and Volk 1993]). Support verb units can be seen as a special type of collocation [Krenn 2000]. They are subject to grammatical restrictions with regard to determiners and passivizability and also with respect to lexical selection. They are distinct from idioms in that their meaning can be derived by combining the meaning of the PP (or NP) with the weakened meaning of the verb (cf. 1.84). Idioms (as in 1.85) require another meaning transfer.

(1.84) *Eine Neuordnung der zeitraubenden Bearbeitung von Geschäftsunterlagen **steht** in zahlreichen Firmen und Behörden **zur Diskussion**.*

(1.85) *Die Deutsche Bank hat vor kurzem ebenfalls ein Unternehmen **aus dem Hut gezaubert**, ...*

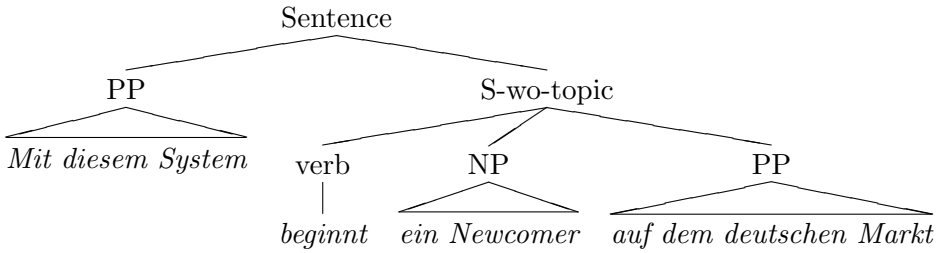
For our purposes a clear distinction between support verb units and similarly structured idioms is not necessary. In both cases the PP must be attached to the verb. For details on our treatment of idioms and support verbs see sections 4.6.1 and 4.6.2.

1.3 The Problem of PP Attachment

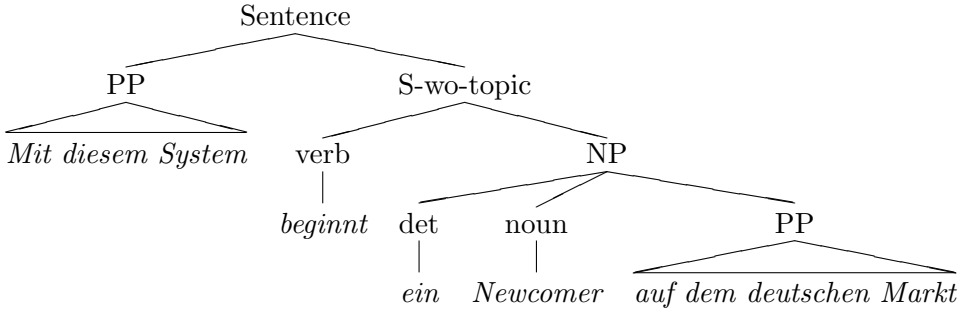
Any system for natural language processing has to struggle first and foremost with the problem of ambiguities. On the syntactic level ambiguities lead to multiple syntactic structures that can be assigned to most sentences. [Agricola 1968] has identified more than 60 different types of syntactic ambiguities for German which are results of ambiguous word forms or of ambiguous word order or constituent order.

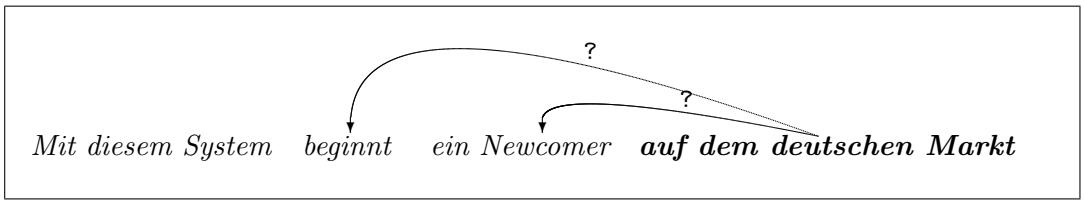
Among these structural ambiguities the problem of prepositional phrase attachment (PP attachment) is most prominent. The most frequent PP ambiguity arises between the attachment to a verb (as prepositional object or adverbial) or to a noun (as an attribute). More precisely, attachment to a verb means positioning the local tree of the PP as a sister node under the same parent node as the verb (as in example tree 1.86). And attachment to a noun means positioning the PP as a sister node of a noun (as in example tree 1.87).[13] The attachment difference corresponds to a meaning difference. In the first case the PP modifies the verb: there is a *Newcomer* who begins on the German market. In the second case the PP modifies the noun: there is a *Newcomer* on the German market who starts with this system on the German market or somewhere else.[14]

(1.86)



(1.87)





[13] It is a matter of debate whether the determiner should also be a sister node to the noun (as in 1.87) or whether it should attach one level up. This matter is not relevant for our research and will be ignored here.

[14] These syntax structures follow the idea that German sentences do not have a verb phrase. A main clause rather consists of a topic position and the remainder without the topic (cf. [Uszkoreit 1987]). S-wo-topic stands for 'Sentence without topic'.

The PP attachment ambiguity arises in German whenever a PP follows immediately after a noun in the *Mittelfeld* of a clause. In this position the PP is accessible to both the verb and the noun. So, when we talk about a PP in an "ambiguous position", we will always refer to such a position in an NP+PP sequence in the *Mittelfeld*. In addition, the head noun of the NP will be called the reference noun, whereas the noun within the PP will be called the core noun or simply the "PP noun". In the above example *Newcomer* is the reference noun and *Markt* is the core noun of the PP.

Vorfeld	left bracket finite verb	*Mittelfeld* ... NP PP ...	right bracket rest of verb group
Mit diesem System	*beginnt*	*ein Newcomer auf dem dt. Markt*	
Mit diesem System	*hat*	*ein Newcomer auf dem dt. Markt*	*begonnen*
Wann	*wird*	*ein Newcomer auf dem dt. Markt*	*beginnen*
	Wird	*der Newcomer auf dem dt. Markt*	*beginnen*

If the NP+PP sequence occurs in the *Vorfeld*, it is generally assumed that the PP is attached to (= is part of) the NP since only one constituent occupies the *Vorfeld* position.

We will illustrate the PP attachment problem with some more corpus examples. If we want to parse the following sentences, we have the problem of attaching the prepositional phrases introduced by *mit* (which in most cases corresponds to the English preposition *with*[15]) either to the preceding noun or to the verb.

(1.88) *Die meisten erwarten, dass der Netzwerk-Spezialist* ***mit einem solchen strategischen Produkt*** *verantwortungsvoll umgehen wird.*

(1.89) *Schon vor zwei Jahren wurde ein Auftragsvolumen von 20 Milliarden Mark* ***mit langlaufenden Währungsoptionen*** *abgesichert.*

(1.90) *Gegenwärtig entsteht ein System* ***mit 140 Prozessoren****.*

The *mit*-PP in example 1.88 needs to be attached to the verb *umgehen* rather than to the preceding noun, since the verb subcategorizes for such a prepositional object. Examples 1.89 and 1.90 are less clear. Neither the verb *absichern* nor *entstehen* strictly subcategorize for a *mit*-PP. From language and world knowledge a German speaker can decide that the *mit*-PP in 1.89 needs to be attached to the verb, whereas in 1.90 it needs to go with the noun.

The occurrence of a post-nominal genitive attribute or another PP will aggravate the attachment problem. Due to the genitive NP in 1.91 there are three possible attachment sites for the PP, the verb *vorschlagen*, the noun *Erweiterung*, and the noun within the genitive NP *CLI-Standards*. 1.92 illustrates the problem with a sequence of two PPs.

(1.91) *... wollen die vier Hersteller gemeinsam eine entsprechende Erweiterung des bestehenden CLI-Standards* ***mit der Bezeichnung NAV/CLI*** *vorschlagen.*

(1.92) *So hat beispielsweise ein bekannter Lebensmittelkonzern seine Filialen* ***in*** *den neuen Bundesländern* ***mit*** *gebrauchten SNI-Kassen ausgestattet.*

[15] See [Schmied and Fink 1999] for a discussion of *with* and its German translation equivalents.

(1.93) *... daß Compaq die Lieferschwierigkeiten **mit** ihrer ProLinea-Reihe **trotz** einem monatlichen Ausstoß **von** 200,000 Stück **in** diesem Quartal **in** den Griff bekommen wird.*

The problem of automatic attachment gets worse the longer the sequence of PPs is. Example 1.93 contains a sequence of five PPs. Still, this sentence does not pose any problem for human comprehension. In fact, only the PP *in diesem Quartal* is truly ambiguous for the human reader; it could be attached to *Ausstoß* or the verb. The other PP attachments are obvious due to the idiomatic usage *in den Griff bekommen* and noun-preposition requirements.

Our approach (and most of the approaches described in the literature) ignores the distinction between adjunct or object (i.e. complement) function of a PP although we are aware that this distinction sometimes causes very different interpretations. In 1.94 the PP will function as prepositional object but it could also be interpreted as temporal adjunct (not least because of the noun *Ende*).

(1.94) *In einem kniffligen Spiel müssen die hoffnungslos naiven Nager **vor dem sicheren Ende** bewahrt werden.*

In most cases the human reader does not realize the inherent syntactic ambiguity in natural language sentences. But they can be made perceivable in humour or in advertising. Currently, the city of Zurich is pestered with advertising posters by an internet provider that deliberately use an ambiguous PP:

(1.95) *Check Deine E-Mails in der Badehose.*

Adjective attachment

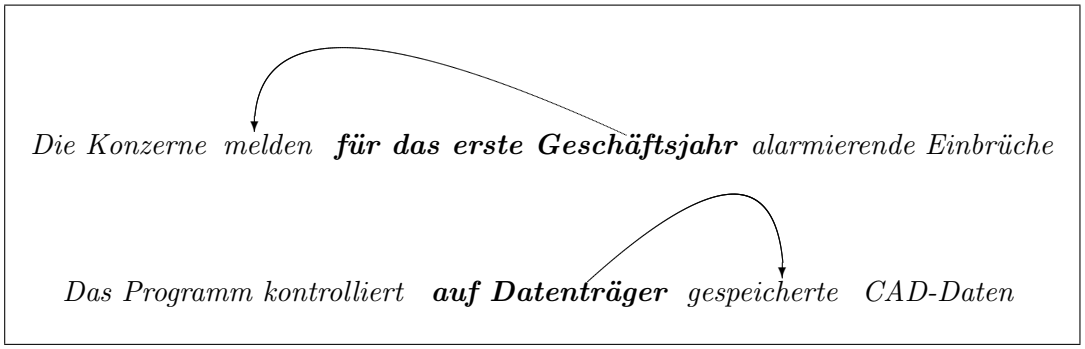
There are other types of difficult PP attachment ambiguities. For example, a PP can be ambiguous between verb attachment and adjective attachment if it occurs immediately preceding an adjective in an NP that lacks a determiner (as in the following examples). The ambiguity is striking for deverbal adjectives (present participle or past participle forms used as noun attributes) since they carry a weakened valency requirement of the underlying verb (as in 1.97). But sometimes this ambiguity pops up with other adjectives as well (cf. 1.98). Overall, these adjective-verb ambiguities are very rare compared to the number of noun-verb ambiguities, and we will not explore them in this book.

(1.96) *Die japanischen Elektronikkonzerne melden **für das erste Geschäftshalbjahr** alarmierende Gewinneinbrüche.*

(1.97) *Das Programm DS-View kontrolliert **auf Datenträger** gespeicherte CAD-Daten auf Korrektheit und Syntaxfehler.*

(1.98) *Diese als BUS bezeichneten Kommunikationsstränge erfordern **im Hintergrund** leistungsfähige schnelle Mikroprozessoren.*

The attachment difference between the PPs in the example sentences 1.96 and 1.97 can best be illustrated by dependency graphs.



Systematically ambiguous PPs

Finally, there are PPs that are systematically ambiguous. An attachment to either noun or verb does not alter the meaning of the sentence (except perhaps for the focus). Most of these indeterminate PPs fall into two classes.

Systematic Locative Ambiguity. If an action is performed involving an object in a place, then both the action and the object are in the place.

(1.99) *Die Modelle der Aptiva-S-Serie benötigen weniger Platz* ***auf dem Arbeitstisch****.*

Systematic Benefactive Ambiguity. If something is arranged for someone (or something), then the thing arranged is also for them (or it).

(1.100) *Das Bundespostministerium hat drei Frequenzen* ***für den Kurzstreckenfunk mit Handsprechfunkgeräten*** *freigegeben.*

[Hindle and Rooth 1993] (p. 113) define that "an attachment is semantically indeterminate if situations that verify the meaning associated with one attachment also make the meaning associated with the other attachment true."

In the 70s and 80s the problem of PP attachment has been tackled mostly by using syntactic and semantic information. With the renaissance of empiricism several statistical methods have been proposed. In chapter 2 we will look at these competing approaches in detail and we will then develop and evaluate our own approach in the subsequent chapters.

1.4 The Importance of Correct PP Attachments

The correct attachment of PPs is important for any system that aims at extracting precise information from unrestricted text. This includes NP-spotting and shallow parsing for information retrieval. The correct attachment of PPs can make the indexing of web-pages more precise by detecting the relationship between PPs and either nouns or verbs. With this information internet search engines can be tuned to higher precision in retrieval. And machine translation (MT) systems can avoid some incorrect translations.

It is often argued that one does not need to resolve PP attachment ambiguities when translating between English and German. And indeed certain ambiguous constructions can be transfered literally preserving the ambiguity. The often quoted example is:

(1.101) *He sees the man with the telescope.*
Er sieht den Mann mit dem Fernglas.

But there are numerous counterexamples that show that both the position of the PP in the target sentence and the selection of the target preposition depend on the correct analysis of the PP in the source text. Consider the German sentence in 1.102 that contains the noun-modifying PP *mit dem blauen Muster* and a location complement realized as the PP *auf den Tisch*. We had this sentence translated by the MT system Langenscheidts T1, one of the leading PC-based MT systems for German - English translation. The system misinterprets the *mit*-PP as a verb modifier and reorders the two PPs which results in the incorrect machine translation.

(1.102) *Er stellt die Vase* ***mit dem blauen Muster*** *auf den Tisch.*
Machine translation: He puts the vase on the table with the blue model.
Correct translation: He puts the vase with the blue pattern on the table.

Langenscheidts T1 has the nice feature of displaying the syntax tree for a translated sentence. The tree for sentence 1.102 is depicted in figure 1.1.[16] We see that both PPs are sister nodes to the accusative object. The noun modifying *mit*-PP is not subordinate to the accusative object NP as it should be.

Since T1 aims at offering a translation for any input sentence, it needs to find exactly one syntax tree for each sentence. If it does not have enough information for attachment decisions, it builds a flat tree and leaves nodes as siblings. This is visible for the ambiguous example sentence in 1.103. T1 follows the analysis in tree 1.86 as can be seen in figure 1.2 on page 23. This analysis results in one of two possible correct translations. The subject NP *a newcomer* was moved to the front while the PP remained in sentence final position.

(1.103) *Mit diesem System beginnt ein Newcomer* ***auf dem deutschen Markt****.*
Machine translation: A newcomer begins with this system in the German market.

Sometimes T1 also errs on the noun attachment side. Sentence 1.105 contains the temporal PP *im letzten Monat* between the accusative object and the prepositional object. In English such a temporal PP needs to be positioned at the beginning or at the end of the sentence. Somehow T1 is misled to interpret this PP as a noun modifier as can be seen in the syntax tree 1.3 on page 24 which results in the incorrect ordering of the temporal PP in the translation.[17]

(1.105) *Der Konzern hat seine Filialen* ***im letzten Monat*** *mit neuen Kassen ausgestattet.*

[16] Most of the node labels for the T1 trees are explained in "Langenscheidts T1 Professional 3.0. Der Text-Übersetzer für PCs. Benutzerhandbuch. Langenscheidt. Berlin. 1997.", in section 19.5 "Abkürzungen in den Analyse- und Transferbäumen", p. 272-273. Note that inflectional suffixes for verbs and adjectives as well as separated verbal prefixes are omitted in the tree display.

[17] The T1 behaviour seems somewhat ad hoc. The same sentence in past tense rather than present perfect is correctly translated with respect to constituent ordering and PP attachment:

(1.104) *Der Konzern stattete seine Filialen im letzten Monat mit neuen Kassen aus.*
Machine translation: The combine equipped its branches with new cash boxes in the last month.

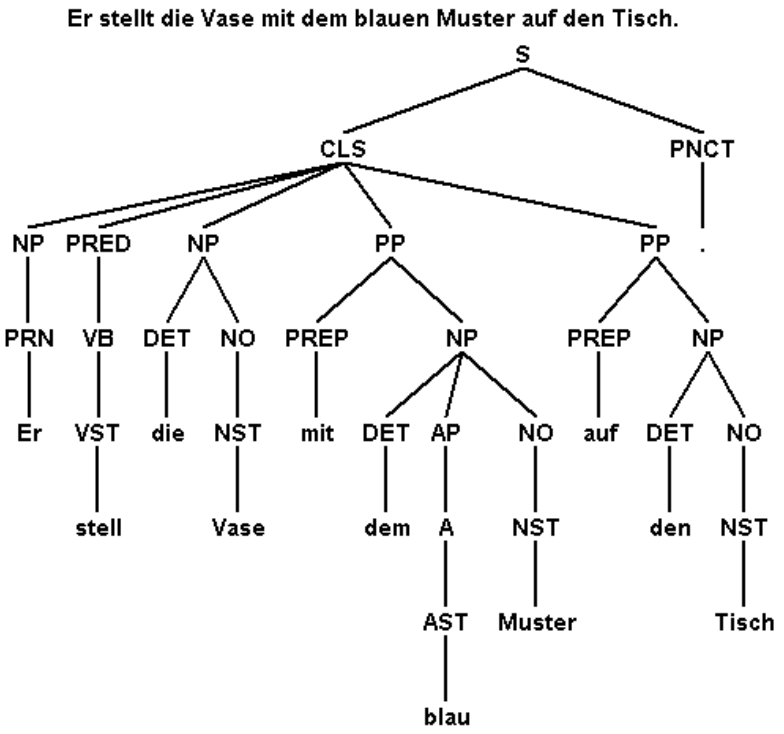


Figure 1.1: T1 tree with incorrectly verb-attached *mit*-PP

Machine translation: The combine equipped its branches in the last month with new cash boxes.
Correct translation: The group equipped its branches with new cash boxes last month.

These examples demonstrate that correct PP attachment is required for any system doing in-depth natural language processing.[18]

1.5 Our Solution to PP Attachment Ambiguities

The present project has grown out of our work on grammar development [Volk et al. 1995, Volk and Richarz 1997]. We have built a grammar development environment, called GTU, which has been used in courses on natural language syntax at the universities of Koblenz and Zurich. In the context of this work we have specialized in the testing of grammars with test suites [Volk 1992, Volk 1995, Volk 1998].

When building a parser for German, we realized that a combination of phrase-structure rules and ID/LP rules (immediate dominance / linear precedence rules) is best suited for a variable word order language like German. It serves best the requirements for both engineering

[18]The separated verb prefix is not shown in tree 1.3. The contracted preposition is divided into preposition and determiner.

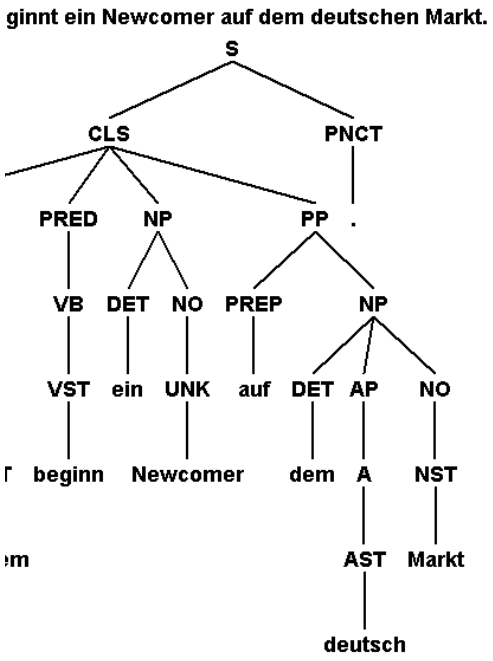


Figure 1.2: T1 tree with the ambiguous *auf*-PP

clarity and processing efficiency. We have therefore built such a parser ([Volk 1996b]) based on an algorithm first introduced by [Weisweber 1987].

Our parser will be integrated into a text-based retrieval system. It must therefore be able to find the best parse for a given input sentence. This entails that it must resolve structural ambiguities as far as possible. Since PP attachment ambiguities are among the most frequent ambiguities, we have looked at various ways of tackling this problem. Although the resolution of PP attachment ambiguities is an "old" area of investigation within the field of natural language processing (see [Schweisthal 1971] for an early study), there are few publications that address this issue for German. We will summarize these in detail in section 2.4.

We started our research by investigating the role of valency information in resolving PP attachments [Volk 1996a]. We surveyed various resources that contain valency information for German verbs ([Wahrig 1978, Schumacher 1986]). It turns out that valency information is a necessary but not a sufficient prerequisite for the resolution of PP attachment.

This has been confirmed by [Mehl 1998]. He observed that many PP complements to verbs are only optional complements. He selected verbs that have multiple readings (according to [Götz et al. 1993]), one of which with an optional PP complement with the preposition *mit* (e.g. *begründen, füttern, drohen, winken*). He manually inspected 794 corpus sentences that contained one of these verbs in the relevant reading. He found that only 38.7% of these sentences realized the optional complement. But only 2.6% of *mit*-PPs in these sentences were not complements. That is good news. If we know that a verb takes a certain PP complement and we find a PP with the required preposition, then the PP is most likely a complement to

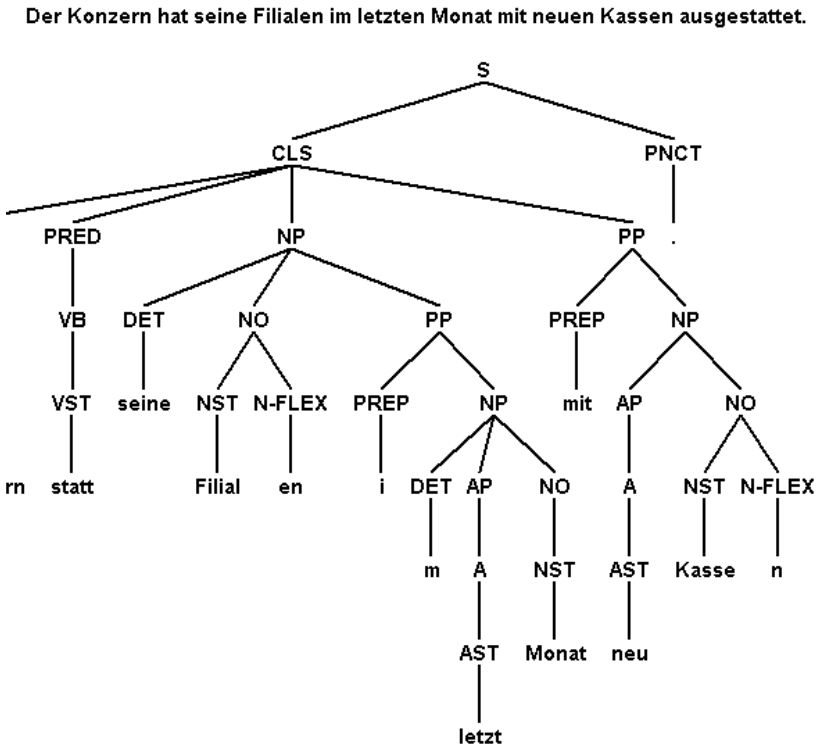


Figure 1.3: T1 tree with incorrectly noun-attached *im*-PP

the verb.

But what if the verb is not listed as taking a PP complement? And what about nouns for which valency lists do not exist (at least not in the above mentioned dictionaries)? And finally, what about the cases when both verb and noun ask for the same PP or none of them does?

Our approach relies on the hypothesis that verb and noun compete for every PP in ambiguous positions. Whichever word has a stronger demand for the PP gets the PP attachment. Strict subcategorization is a special case of this. But often both verb and noun 'subcategorize' for the PP to a certain degree. This degree of subcategorization is what we try to capture with our notion of cooccurrence strength derived from corpus statistics.

Our method for determining cooccurrence values is based on using the overall frequency of a word against the frequency of that word cooccurring with a given preposition. For example, if some noun N occurs 100 times in a corpus and this noun cooccurs with the preposition P 60 times, then the cooccurrence value $cooc(N, P)$ will be $60/100 = 0.6$. The general formula is

$$cooc(W, P) = freq(W, P)/freq(W)$$

in which W can be either a noun N or a verb V, $freq(W)$ is the number of times that the word W occurs in the corpus, $freq(W, P)$ is the number of times that the word W cooccurs with the preposition P in the corpus, and $cooc(W, P)$ is the resulting cooccurrence value. For example, the N+P cooccurrence value is the bigram frequency of a noun + preposition sequence divided by the unigram frequency of the noun. The cooccurrence value is discussed in more detail in section 4.2.

In a pilot project (reported in [Langer et al. 1997]) we have extracted cooccurrence values from different German corpora. We have focussed on one preposition (*mit*) and investigated N+P and V+P cooccurrences. Table 1.1 gives the top of the noun + *mit* cooccurrence list derived from one annual volume of our computer magazine corpus [Konradin-Verlag 1998]. The frequency counts are based on word forms. That is why the noun *Gespräch* appears in this list in three different forms. The cooccurrence values are intuitively plausible but their usefulness needs to be experimentally tested.

noun N	$freq(N, mit)$	$freq(N)$	$cooc(N, mit)$
Umgang	147	155	0.94
Zusammenarbeit	256	575	0.44
Zusammenhang	93	239	0.38
Gesprächen	13	35	0.37
Auseinandersetzung	19	53	0.35
Beschäftigung	11	32	0.34
Interview	23	74	0.31
Kooperation	126	424	0.29
Partnerschaft	31	106	0.292
Gespräche	42	144	0.291
Verhandlungen	36	142	0.253
Kooperationen	43	172	0.250
Gespräch	30	123	0.243
Verbindung	133	572	0.232

Table 1.1: Cooccurrence values of German noun forms + the preposition *mit*

Computing cooccurrences is much more difficult for German than for English because of the variable word order and because of morphological variation. In particular it is difficult to find the V+P cooccurrences since the verb can have up to four different stems and more than a dozen inflectional suffixes. In addition, German full verbs are located at first position (in questions and commands), second position (in matrix clauses in present or past tense and active mood), or clause final position (in the remaining matrix clauses and in all subordinate clauses). If the verb is in first or second position, it may have a separated verb prefix in clause final position.

Past linguistic methods for the resolution of PP attachment ambiguities have been limited to handcrafted features for small sets of verbs and nouns. Statistical approaches with supervised learning required syntactically annotated and manually disambiguated corpora (so called treebanks). Our approach combines unsupervised learning with linguistic resources. It offers a wide coverage method that, in its pure form, requires only a text corpus and special corpus processing tools. These tools are partly available in the research community (such as

tagger and lemmatizer), or they were developed and improved as part of this research (such as a clause boundary detector and a proper name classifier).

In a first approximation we assume that every PP that immediately follows a noun raises a PP attachment ambiguity. In our computer magazine corpus we find 314,000 sequences of a noun followed by a preposition (in 420,000 sentences). This illustrates how widespread the problem of PP attachment is.

The task of finding criteria for PP attachment (as discussed in this book) is similar to the automatic recognition of subcategorization frames. The cooccurrence values that are the basis for PP attachment can be seen as specialized subcategorization frames with varying degrees of strength.

Therefore we see a great degree of similarity of our approach to [Wauschkuhn 1999], who worked on the automatic extraction of verbal subcategorization frames from corpora. His idea was to determine verbal complements, group them into complement patterns, and differentiate the relative frequencies for different verb readings. This presupposes a sentence processing similar to ours in corpus preparation. For every sentence Wauschkuhn determined the clause structure and phrases like NPs (including multiword proper names), PPs and the verb group. Subcat frame extraction then worked on chosen clause types (matrix clauses, *zu* infinitives). Passive clauses were turned into active clauses. The resulting constituents were grouped based on the most frequent constituents or based on association discovery methods. Optional complements were distinguished from obligatory complements if the system determined two complement patterns that differed only in one complement C. The two patterns were then unified with the additional information that C is optional.

The automatically computed subcategorization frames of seven verbs (out of more than 1000 listed in the book's appendix) were manually compared to the valency information in [Helbig and Schenkel 1991]. The overall evaluation scores are 73% precision and 57% recall. Wauschkuhn notes that PPs pose special problems in his analysis because he has no means to decide between verb and noun attachment.

Positioning our Approach

Our approach to PP attachment resolution is based on shallow corpus analysis. It is thus positioned at the intersection of Computational Linguistics and Corpus Linguistics. In the last decade the working methods in Computational Linguistics have changed drastically. Fifteen years back, most research focused on selected example sentences. Nowadays the access to and exploitation of large text corpora is commonplace. This shift is reflected in a renaissance of work in Corpus Linguistics and documented in a number of pertinent books in recent years (e.g. the introductions by [Biber et al. 1998, Kennedy 1998] and the more methodologically oriented works on statistics and programming in Corpus Linguistics by [Oakes 1998, Mason 2000]).

The shift to corpus-based approaches has entailed a focus on naturally occurring language. While most research in the old tradition was based on constructed example sentences and self-inspection, the new paradigm uses sentences from machine-readable corpora. In parallel the empirical approach requires a quantitative evaluation of every method derived and every rule proposed.

Our work follows the new paradigm in both the orientation on frequent phenomena and in rigorous evaluation. We have developed and adapted modules for corpus annotation. The

corpus is the basis for the learning algorithms that derive cooccurrence frequencies for the disambiguation of PP attachments. The disambiguated PPs will be used for improved corpus annotation or for other tasks in natural language processing.

Corpus Linguistics, in the sense of using natural language samples for linguistics, is much older than computer science. The dictionary makers of the 19th century can be considered Corpus Linguistics pioneers (e.g. James Murray for the Oxford English Dictionary [Murray 1995] or the Grimm brothers for the Deutsches Wörterbuch). But the advent of computers changed the field completely.

Linguists started compiling collections of raw text for ease of searching. In a next step, the texts were semi-automatically annotated with lemmas and later with syntactic structures. First, corpora were considered large when they exceeded one million words. Nowadays, large corpora comprise more than 100 million words. In relation, our training corpora of 5 to 7 million words need to be ranked as middle size corpora. But we have also experimented with the world wide web (WWW) which can be seen as the largest corpus ever with more than one billion documents.

The current use of corpora falls into two large classes. On the one hand, they serve as the basis for intellectual analysis, as a repository of natural linguistic data for the linguistic expert. On the other hand, they are used as training material for computational systems. The program computes statistical tendencies from the data and derives or ranks rules which can be applied to process and to structure new data. For example, [Black et al. 1993] describe the use of a treebank to assign weights to handcrafted grammar rules. Our work also falls in the second class.

The developments in computer technology with the increase in processing speed and the access to ever larger storage media has revolutionized Corpus Linguistics. [Eroms 1981], twenty years ago, did an empirical study of German prepositions. He searched through the LIMAS-Corpus and through a corpus at the "Institut für deutsche Sprache" for example sentences with the preposition *mit*. But he notes (p. 266):

> Wegen der bei den Suchprogrammen anzugebenden Zeitlimits ist manchmal das Programm abgebrochen worden, bevor die Bänder vollständig abgefragt worden waren. ... Verben mit weit überdurchschnittlicher Häufigkeit wie *geben* eignen sich weniger gut für rechnergestützte Untersuchungen, weil die hohe Belegzahl bald zum Programmabbruch führt.

Since then, working conditions for corpus linguists have changed. Many have access to powerful interfaces to query large corpora (such as the Corpus Query Workbench at Stuttgart) not least through the internet.[19]

Corpus Linguistics methods are actively used for lexicography, terminology, translation and language teaching. It is evident that these fields will profit from annotated corpora (rather than raw text corpora). Lexicons can be enriched with frequency information for different word readings, subcategorization frames (as done by [Wauschkuhn 1999] described above) or collocations (as explored by [Lemnitzer 1997] or [Heid 1999]). [Gaussier and Cancedda 2001] show how the resolution of PP attachment is relevant to automatic terminology extraction.

[19] See `http://corpora.ids-mannheim.de/~cosmas/` to query the new versions of the corpora at the "Institut für deutsche Sprache".

1.6 Overview of this Book

The overall goal of our research is to find methods for the resolution of PP attachment ambiguities in German. The most promising wide-coverage approach is the utilization of statistical data obtained from corpus analysis. The central questions are

1. To what degree is it possible to use linguistic information in combination with statistical evidence?
2. How dependent on the domain of the training corpus are the statistical methods for PP attachment?
3. How is it possible to combine unsupervised and supervised methods for PP attachment? Will the combination lead to improved results over the single use of these methods?
4. Will statistical approaches to PP attachment lead to similar results for German as have been reported for English?

In **chapter 2** we will survey the approaches to PP attachment disambiguation reported in the literature. We differentiate between linguistic and statistical approaches. The latter will be subclassified into supervised methods (based on manually controlled training data such as treebanks) and unsupervised methods (based on raw text corpora or at most automatically annotated corpora). Most of the literature is on PP attachment for English, but we have also tracked down some material for German.

Our modules for corpus preparation are described in **chapter 3**. We detail the steps for shallow parsing our training corpus including proper name classification, part-of-speech tagging, lemmatization, phrase chunking and clause boundary detection. The tagger determines the part-of-speech tags for every word form in the input sentence. The clause boundary detector uses these tags to split the sentence into single verb clauses. In addition to the automatic annotation of the training corpus we have compiled two test sets with over 10,000 test cases. We will describe how the sentences were selected, manually annotated with syntactic structures, and how the test cases were extracted.

Chapter 4 sets forth the core experiments. We start by computing a base line using only linguistic information, subcategorization frames from CELEX and a list of support verb units. We then delve into a number of statistical experiments, starting with frequency counts over word forms. It turns out that our way of counting the bigram frequencies leads to a bias for verb attachment. This needs to be counterbalanced by a noun factor which is derived as the ratio of the general tendency of prepositions to cooccur with verbs rather than nouns.

From this starting point we follow two goals. On the one hand, we increase the coverage, the number of test cases that can be decided based on the training corpus. We use various clustering techniques towards this goal: lemmatization, decompounding of nouns, proper name classes, and the GermaNet thesaurus. In addition we propose to use partial information in threshold comparisons rather than to insist on both cooccurrence values for verbs and nouns to be present. On the other hand, we attempt to increase the attachment accuracy, the number of correctly attached cases from our test sets. We explore the distinction of sure vs. possible attachments in training, the use of support verb units, deverbal vs. regular nouns, reflexive verbs, local vs. temporal PPs, and the core noun of the PP.

For example, deverbal nouns may reuse cooccurrence information taken from the respective verbs. But since nouns do not subcategorize as strongly as verbs, the statistical measures need to be adjusted. See, for example, the German verb *warnen* which has a high probability of occurring with the preposition *vor*. Then we predict that the derived noun *Warnung* will also frequently cooccur with this preposition, but this probability will be lower than the probability for the verb (cf. section 4.7).

Intuitively, the cooccurrence measure described above should distinguish between the different readings of the verbs. It sometimes happens that a verb has a strong requirement for some preposition in one reading, and it does not have any requirement in another. The German verb *warten* meaning either *to wait* or *to maintain/repair* may serve as an example. In the first sense it strongly asks for a prepositional object with *auf*, whereas in the second sense it does not have any strong prepositional requirement. In general, it is very difficult to distinguish different verb readings short of doing a complete syntactic and semantic analysis. One special case in German, though, is the relatively clear distinction between reflexive and non-reflexive usage and we will look into this in section 4.8.

In chapter 4 we stick to a specific training corpus. We explore the influence of other training corpora in chapters 5 and 6. In **chapter 5** we exchange our domain-specific training corpus with a general newspaper corpus, and in **chapter 6** we use frequency counts from the WWW as the basis for the computation of cooccurrence values.

With our disambiguation method well-established we evaluate it against another unsupervised method (Lexical Association score by [Hindle and Rooth 1993]) and two supervised methods (Back-off by [Collins and Brooks 1995] and Transformation-based by [Brill and Resnik 1994]) in **chapter 7**. We compensate the lack of a large treebank by cross-validation. Based on the accuracies of the different decision levels in the Back-off supervised method and in our own method, we suggest an intertwined model of combining the two approaches. This model leads to the best overall attachment results.

Chapter 8 summarizes the results and contributions of this work, and points out some directions for improvements in corpus processing and future research on automatic disambiguation.

Chapter 2

Approaches to the Resolution of PP Attachment Ambiguities

Before reporting on our own research we will survey the approaches to PP ambiguity resolution that have been attempted elsewhere. We broadly distinguish between linguistic and statistical means.

2.1 Ambiguity Resolution with Linguistic Means

Syntactic approaches use the structural properties of parse trees to decide on attachment ambiguities. Numerous principles have been suggested to best capture these properties. Most of these principles are derived from studies on human sentence processing. The best known principles have been proposed by [Frazier 1978]:

Minimal Attachment. A new constituent is attached to the parse tree using as few non-terminal nodes as possible. In other words: Avoid all unnecessary nodes in the parse tree.

Late Closure. If permitted by the grammar, attach new items into the most recent phrase. This corresponds to Kimball's principle of Right Association [Kimball 1973] except that it is extended from terminal symbols to constituents.

These two principles are ordered, meaning that Minimal Attachment dominates in cases of conflict. In the case of PP-attachment, Minimal Attachment predicts that the PP will always be attached to the verb. Obviously this is not an adequate solution.

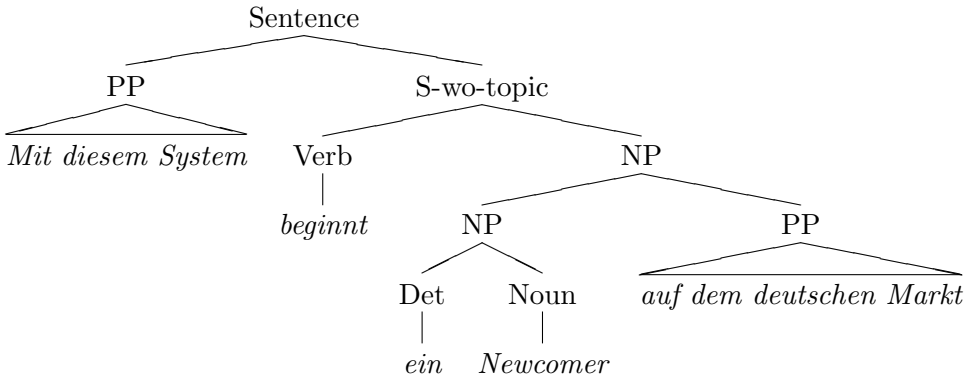
Furthermore, [Konieczny et al. 1991] point out that the Minimal Attachment principle is dependent on the underlying grammar. Consider the example rules `r1` through `r3`. Minimal Attachment will predict verb attachment for a PP if we assume a flat rule for a simple NP (`r2`) and the recursive rule `r3` for an NP combining with a PP (as in tree 2.1). This results in one more node for the noun attachment than for the verb attachment (cf. the tree in figure 1.86 on page 17).

```
(r1a) VP  --> V   NP
(r1b) VP  --> V   NP PP
```

```
(r2)  NP  --> Det N
(r3)  NP  --> NP  PP
(r3b) NP  --> Det N  PP
```

If, on the contrary, we assume a flat rule like `r3b` for the NP combining with the PP, there is no difference in the number of nodes (compare the trees 1.86 and 1.87 on page 17).

(2.1)



[Konieczny et al. 1991] therefore propose a Head Attachment principle which we will discuss in section 2.4.

[Schütze 1995] argues for a different generalization. Following [Abney 1989] he suggests that argument attachment is always prefered over modifier attachment. He quotes the following example.

(2.2) *I thought about his interest in the Volvo.*

Even though sentence 2.2 is ambiguous, people prefer the interpretation in which the PP describes what he was interested in rather than the location of the thinking. This entails that the distinction between arguments and modifiers must be made operational. First, Schütze defines it as follows ([Schütze 1995] p. 100).

> An argument fills a role in the relation described by its associated head, whose presence may be implied by the head. In contrast, a modifier predicates a separate property of its associated head or phrase.
>
> A phrase P is an argument of a head H if the semantic contribution of P to the meaning of a sentence ... depends on the particlar identity of H. Conversely, P is a modifier if its semantic contribution is relatively constant across a range of sentences in which it combines with different heads.

Then he presents a number of tests for argumenthood. He divides them into semantic tests (e.g. optionality, head-dependence, copular paraphrase) and syntactic tests (e.g. pro-form replacement, pseudo-clefting, extraction). The main argument is that if you know the arguments of a verb or a noun then you can decide the PP attachment. But discussing the tests Schütze concedes that none of them gives a clear-cut binary decision for all cases, rather that there are degrees of argumenthood. And this is exactly what we try to capture using a statistical measure.

Schütze's work followed [Britt 1994]. She had found that attachment decisions "interacted with the obligatory/optional nature of verb arguments". Her experiments furthermore supported a limited influence of discourse semantics.

This line of research was continued by [Boland 1998] with studies on human processing of ambiguous PPs. Boland used mostly sentences in which the verb and the noun call for the same PP argument and both PPs are given.

(2.3) *John gave a letter to his son to a friend earlier today.*

Experiments measured word by word sensibility judgements and reading times. The results can be summarized as "lexically based thematic constraints guide PP attachment in dative sentences" (p.27), "immediate commitments are made when the evidence for a particular analysis is very strong". These findings are good news for computational linguistics. If lexical constraints dominate pragmatic constraints in human sentence processing, this implies that such lexical constraints will also solve most attachment problems computationally. Most pragmatic constraints are out of the reach of current computer systems anyhow.

SEMANTIC APPROACHES to the resolution of PP-attachment ambiguities vary widely, ranging from selectional restrictions to semantic heuristics. Selectional restrictions are based on semantic features such as ANIMATE or ABSTRACT that can be used to select from among the possible complements of a verb. [Jensen and Binot 1987] is an early example of this approach. They determine PP attachments by searching for the function of the PP. They demonstrate their approach for the preposition *with* and example sentence 2.4. For this sentence they automatically determine the function INSTRUMENT in contrast with the function PART-OF in 2.5.

(2.4) *I ate a fish with a fork.*

(2.5) *I ate a fish with bones.*

In these pre-WordNet days [Jensen and Binot 1987] suggested that hyponym relations be extracted by parsing the definitions from online dictionaries (Webster's and Longman). They searched these definitions for specific patterns that point to a semantic function (*X is used for Y* or *X is a means for Y* points to the instrument relation). The attachment decision is then based on heuristics like:

> If some instrument pattern exists in the dictionary definition of the prepositional complement *fork* and if this pattern points to a link with the head noun *fish*, then attach the PP to the noun.

Another semantic approach is presented by [Chen and Chang 1995]. They also take the semantic classes from a dictionary and use them for conceptual clustering and subsequent ranking with information retrieval techniques. Semantic features are certainly helpful for the disambiguation task but they can only be put to a large scale use if machine-readable dictionaries or large semantic networks such as WordNet [Miller 1995] are available.

Other semantic approaches have become known as case-frame parsing [Carbonell and Hayes 1987]. Parsers use domain specific knowledge to build up an expectation frame for a

verb. Constituents are then assigned to the frame's slots according to their semantic compatibility. An extended version of this approach is used by Hirst's ABSITY parser [Hirst 1987] with a frame representation based on Montague's higher order intensional logic.

[Hirst 1987] (p. 173) describes a detailed decision algorithm for PP attachment:

```
If NP attachment gives referential success
  then attach to NP
else if VP attachment is implausible
  then attach to NP
else if NP attachment is implausible
  then attach to VP
else if verb expects a case that the preposition could be flagging
  then attach to VP
else if the last expected case is open
  then attach to NP
else if NP attachment makes unsuccessful reference
  then attach to VP
else [sentence is ambiguous]
  then attach to VP
```

Thus Hirst uses lexical preferences (i.e. preferences about prepositional complements triggered by the verb), semantic plausibility checks (a refined notion of selectional restrictions), and pragmatic plausibility checks (checking for an instance of the object or action in the knowledge base). Hirst points out that such plausibility checks go back to [Winograd 1973]. When processing sentence 2.6, Winograd's SHRDLU system checked whether there existed a **block in the box** or a **box on the table** in the model.

(2.6) *Put the block in the box on the table.*

[Crain and Steedman 1985] have called this technique "the principle of referential success". And they hypothesize that it can be generalized as a kind of presupposition satisfaction. The reading that satisfies the most presuppositions is the one to be preferred. This works along the following lines (p. 170).

1. A definite NP presupposes that the object or event it describes exists and that it is available in the knowledge base for unique reference.

2. The attachment of a PP to an NP results in new presuppositions for the NP, but cancels its uniqueness.

3. The attachment of a PP to a VP creates no new presuppositions but rather indicates new information.

This predicts that if attachment to a definite NP leads to an unknown entity, verb attachment will win. On the other hand, if NP attachment results in a definite reference the number of presuppositions remains the same and therefore noun attachment will win. In this way definiteness is one feature to be used for deciding on PP attachment. Obviously, such a detailed knowledge representation is only possible for limited domains. On the other hand,

we have to concede that there are ambiguous PPs that can only be correctly attached with such detailed information.

In a similar way semantic features are used in the research on word-expert or word-agent parsing [Small and Rieger 1982, Helbig et al. 1994, Schulz et al. 1997]. The analysis by [Helbig et al. 1994] is based on multiple principles, three of which deal with attachment problems. Most important is the valency principle which consists of compatibility checking and priority checking. Compatibility checking examines the semantic compatibility of a prospective complement. This means that the semantic content of every constituent must be determined. Their system contains semantic rules for every preposition. For the preposition *über* there are, among others, the following two rules [Schulz et al. 1995] which account for the example sentences 2.7 and 2.8 respectively:[1]

```
IF semantics = geographical-concept
  AND case = accusative
    THEN semantic sort = location; semantic relation = via

IF semantics = quantity
  AND case = accusative
    THEN semantic relation = greater
```

(2.7) *Er flog* ***über die Alpen****.*

(2.8) *Er hat* ***über 50 Bücher*** *geschrieben.*

Approaches like this use semantic knowledge almost to the fullest extent possible today. But building up the respective knowledge bases requires extensive manual labor, which prohibits the large scale usage of this approach.

Nevertheless, using deep semantic knowledge remains popular, as can be seen with HPSG parsing [Pollard and Sag 1994, Müller 1999, Richter and Sailer 1996]. In HPSG, complex feature structures are used to encode semantic features. These semantic features are employed in parallel with syntactic features when parsing a sentence. This works well for limited domains, but it is much too brittle for wide coverage parsing.

Therefore, others have set up general semantic heuristics. This approach has been called Naive Semantics by [Dahlgren 1988]. It is based on commonsense semantic primitives, three of which work on PP-attachment (quoted from [Franz 1996a]):

Lexical level commonsense heuristics. This includes rules of the form "If the prepositional object is temporal, then the PP modifies the sentence."

Lexical knowledge. An example of a syntactic disambiguation rule is "certain intransitive verbs require certain prepositions, e.g. *depend on, look for*."

Preposition-specific rules. An example of a preposition-specific rule is, "if the preposition is the word *at*, and the prepositional object is abstract or ... a place, then attach the PP to the sentence. Otherwise, attach it to the NP."

[1] Example 2.8 is taken from [Schulz et al. 1995]. In this example the sequence *über 50 Bücher* looks like a PP but in fact *über* functions as a complex comparative particle. The sequence should be considered an NP built from an adjective phrase and a noun (in analogy to *mehr als 50 Bücher*).

This model again depends on semantic features that help the program to identify whether a PP is temporal or local etc. In addition, its "lexical knowledge" principle depends on a verb's subcategorization requirement. It is well known that some verbs require a prepositional complement with a specific preposition. In German this even extends to the case requirement within the PP. For example the verb *warten* requires the preposition *auf* with an NP in accusative case (whereas this preposition could also occur with a dative NP). Such subcategorization knowledge should certainly be used for disambiguation and is available for many German verbs in the lexical database CELEX [Baayen et al. 1995].

Another elaborate rule-based approach that also requires a semantic dictionary is presented by [Chen and Chen 1996]. They distinguish between four types of PPs: predicative PPs (including verb complement PPs), sentence modifying PPs, verb modifying PPs and noun modifying PPs. No clear definition is given to tell apart the first three of these types which all involve some degree of verb modification. The majority of test cases (92%) is classified as verb modifying (43%) or noun modifying (49%). Their algorithm for the resolution of PP attachment is as follows:

1. Check if the PP is a predicative PP according to the predicate-argument structure of the clause.

2. Check if the PP is a sentence modifying PP according to one of 21 specific rule templates involving the preposition and the semantic classification of the PP. Example templates:

```
<'after'   (time)                >
<'at'      (location | time)     >
<'out of'  (abstract | location) >
```

3. Check if the PP is a verb modifying PP according to one of 46 specific rule templates involving the semantic features of the verb (optional), of the reference noun (optional) and of the PP as well as the preposition itself. Example templates:

```
<motion, _,     'about',  (object, location)   >
<action, event, 'after',  (concrete)           >
<motion, _,     'out of', (concrete, location) >
```

4. Otherwise it is a noun modifying PP.

This entails that on the one hand the predicate-argument structure and on the other hand the semantic class for verbs and nouns must be determined before the disambiguation rules can be applied. [Chen and Chen 1996] use an NP parser and a "finite-state mechanism" to decide on one of the 32 verb frame patterns from the Oxford Advanced Learner's Dictionary as the appropriate predicate-argument structure.

The semantic features for all verbs and nouns are extracted from Roget's thesaurus and mapped to a medium scale ontology (maximally 5 levels deep) developed by the authors. No information is given on how they resolve sense ambiguities.

The algorithm is evaluated over a large set (14,759 PPs) from the Penn Treebank. From the example given in the paper we gather that the authors included ambiguously and non-ambiguously positioned PPs. They report on 100% "correctness" for noun modifying PPs, sentential PPs and predicative PPs. Verb modifying PPs are allegedly 77% correct. Obviously these figures do not describe the precision of the algorithm. If they did describe precision, the missing 23% of verb modifying PPs would need to show up as false negatives in at least one of the other classes. But even with this restriction the 100% figures are unbelievable. Based on our own experiments and on the other experiments described in the literature we doubt that it is possible to achieve perfect attachment for several hundred sentence modifying PPs based on 21 rules.

Linguistic PP ambiguity resolution is used today in some commercial NLP tools. [Behl 1999] describes how PowerTranslator, a machine translation system developed by Globalink and L&H[2], decides PP attachments. She argues that translating from English to German requires reordering of semantic units which can only be performed correctly if such units (complex phrases including PP attributes) are moved as a whole. A semantic unit is a sequence of NPs and PPs that serve the same function within a sentence (complement or adjunct of time, place or manner).

(2.9) *He gave a talk* ***on the new bridge in City Hall****.*

(2.10) *Er hielt eine Rede* ***auf*** *der neuen Brücke im Rathaus.*

(2.11) *Er hielt eine Rede* ***über*** *die neue Brücke im Rathaus.*

(2.12) *Er hielt im Rathaus eine Rede* ***über*** *die neue Brücke.*

(2.13) *Er hielt* ***auf*** *der neuen Brücke im Rathaus eine Rede.*

Literal translation of 2.9 leads to a problem in preposition selection as in 2.10 or 2.11. PowerTranslator used to incorrectly translate 2.9 as 2.13 since it ordered the adjuncts preceding the complements. By using a newly added subcategorization requirement of the noun *talk* requiring a PP complement with *on*, the system finds a correct translation as in 2.12.[3] PowerTranslator also has rules to decide conflicting requirements between verb and noun as in 2.14. Both the verb *talk* and the noun *information* require an *on*-PP.

(2.14) *He relied in his talk* ***on wombats on the latest information on marsupials****.*

These rules work with the type of the semantic unit, the subcategorization requirement and the definiteness of the article. This, of course, requires a reliable identification of the type of the semantic unit.

If we compare PowerTranslator's strategy with other MT systems, we see that these systems employ some PP attachment disambiguation strategies. Langenscheidts T1 (Version 3.3) correctly attaches *on the bridge* to the preceding noun as we can observe with the help of the T1 tree drawing module (cf. tree 2.1). T1 still produces the translation 2.15 leaving the clause-attached PP *in City Hall* in its original position. It finds the correct translation for the preposition *on* but ends up with an incorrect translation of the verb.

[2]See `www.lhsl.com/powertranslator/`.

[3]Translation 2.11 could also be regarded as a correct translation.

(2.15) *T1 translation: Er führte einen Vortrag über die neue Brücke im Rathaus auf.*

(2.16) *Personal Translator translation: Er hielt eine Rede über die neue Brücke in Rathaus.*

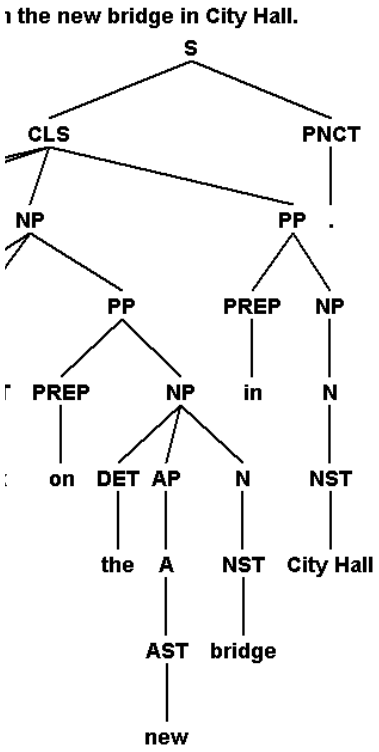


Figure 2.1: T1 tree with correctly attached PPs

Personal Translator 2001 Office Plus[4] translates sentence 2.9 as 2.16 with the correct translation of *on* but with the non-contracted and thus incorrect form of the preposition *in*.

In a more recent study [Fang 2000] describes a large scale experiment using linguistic rules to automatically determine the syntactic function of PPs in the International Corpus of English (ICE), a million-word corpus that has been annotated at the syntactic level. In this corpus there are 248 different prepositions including 160 complex ones (e.g. *in terms of, according to, in accordance with*). Fang notes that close to 90% of prepositional use in the corpus can be attributed to the 15 most frequent atomic prepositions (with *of* and *in* being by far the most frequent). The English preposition *of* leads to PPs that are most likely attached to an immediately preceding noun or adjective.

(2.17) *For most countries the ICE project is stimulating the first systematic investigation* ***of the national variety****.*

(2.18) *This new and important audience is largely ignorant* ***of the idiosyncrasies of legal research****.*

[4]Personal Translator is marketed by linguatec in Munich. See `www.linguatec.de`.

[Bowen 2001] found that 98% of English nouns that take a PP complement take an *of*-PP complement (many of them take other PP complements as well).

This is a systematic difference to German where most English *of*-PPs will be rendered as genitive NPs. Including *of*-PPs in a study on English PP attachment thus gives an advantage to English over German since this preposition is by far the most frequent and in most cases its attachment is evident.

[Fang 2000] extracted 80,393 PPs from the ICE treebank with 42% noun attachment, 55% verb attachment and 3% adjective attachment. He manually compiled rules for the disambiguation of these PPs. The rules for noun attachment are

1. Treat as noun modifying any PP headed by *of*.
2. Treat as noun modifying any PP following a sentence-initial NP.
3. Treat as noun modifying any PP whose preposition collocates with the head of the antecedent NP (based on a large collocation lexicon). For deverbal nouns collocations of the underlying verb are used.
4. Treat as noun modifying any PP that follows an NP governed by a copula antecedent VP.

The rules for adjective attachment are similar, and every PP that does not match any of the noun or adjective attachment rules is regarded as verb attachment. That means that Fang mixes preposition-specific rules (as for the *of*-PPs), collocations from a large lexical database, and structural constraints (e.g. using the information that the PP follows a sentence initial NP). Fang claims that this rule system correctly attaches 85.9% of the PPs in his test set.

This result is to be regarded with caution since he does not distinguish between ambiguously and non-ambiguously positioned PPs. As a fact, most PPs will occur in non-ambiguous positions and are thus not subject to disambiguation. A more interesting figure is the 76.3% accuracy that he reports for noun attachment. This figure is relative to the set of all PPs that were manually attached to the noun. It says that if the system looks at all the PPs (of which you know that they are attached to the noun) it can replicate the human judgement in 76.3% of the cases based on the above rules. Fang's results cannot be compared to the accuracy percentages in the next section where we look only at the set of ambiguous PPs.

2.2 Ambiguity Resolution with Statistical Means

This line of research was initiated by [Hindle and Rooth 1993]. They tackle the PP-attachment ambiguity problem (for English) by computing **lexical association scores** from a partially parsed corpus. If a sentence contains the sequence V+NP+PP, the triple V+N+P is observed with N being the head noun of the NP and P being the head of the PP. From example 2.19 they will extract the triple (*access, menu, for*).

(2.19) *The right mouse button lets you access pop-up menus for cycle options.*

The lexical association score LA is computed as the log_2 of the ratio of the probabilities of the preposition attached to the verb and of the preposition attached to the preceding noun.

$$LA(V, N_1, P) = log_2 \frac{prob(verb\_attach\ P|V, N_1)}{prob(noun\_attach\ P|V, N_1)}$$

A lexical association score greater 0 leads to a decision for verb attachment and a score less than 0 to noun attachment. The probabilities are estimated from co-occurrence counts. Although the partially parsed corpus contains the PPs unattached, it provides a basis for identifying sure-verb attachments (e.g. a PP immediately following a personal pronoun) and sure-noun attachments (e.g. a PP immediately following a noun in subject position). In an iterative step, lexical association scores greater than 2.0 or less than -2.0 that indicate clear attachments are used to assign the preposition to the verb or to the noun. The remaining ambiguous cases are evenly split between the two possible attachment sites.

Hindle and Rooth evaluated their method on 880 manually disambiguated verb-noun-preposition triples (586 noun attachments and 294 verb attachments). It results in 80% correct attachments (with V attachment being worse than N attachment).[5] We have reimplemented this method and tested it on our German data. These experiments are described in section 7.1.1.

While [Hindle and Rooth 1993] did not use any linguistic resource, except for the shallow parser, subsequent research first focussed on learning the attachment decisions from the Penn Treebank, a corpus of 1 million words which are manually annotated with their syntactic structure.[6] The sentences are bracketed with their phrase structure. Each node is labeled with a constituent name (NP, PP etc.) and with a function symbol (subject, adverbial etc.). Automatically learning preferences from manually disambiguated data is usually called supervised learning.

2.2.1 Supervised Methods

[Ratnaparkhi et al. 1994] used a **Maximum Entropy** model considering V+N_1+P+N_2. N_1 is the head noun of the NP, the possible reference noun of the PP. N_2 is the head noun of the NP governed by the preposition. The principle of Maximum Entropy states that the correct distribution maximizes entropy ("uncertainty"), based on constraints which represent evidence. Maximum entropy models can be explained under the maximum likelihood framework. Using a Maximum Entropy model serves to solve statistical classification problems. In a training phase the system determines a set of statistics to capture the behavior of the process. In the application phase the model predicts the future output of the process. The difficulty lies in determining the features for the classification task at hand.

[Ratnaparkhi et al. 1994] employed n-grams of words as features (i.e. the nouns, verbs and prepositions as they occur in the training corpus) and a class hierarchy derived from mutual information clustering. They established a training set of 20,801 and a test set of 3097 quadruples from the Penn Treebank Wall Street Journal material (which became sort of a benchmark, reused in subsequent experiments by other researchers.[7]) For ease of reference

[5] An easily accessible overview of the [Hindle and Rooth 1993] method with an explanation of some of the mathematics involved can be found in section 8.3 of [Manning and Schütze 2000].

[6] See `www.cis.upenn.edu/~treebank/`.

[7] The training and data sets are available from `ftp://ftp.cis.upenn.edu/pub/adwait/PPattachData/`. [Pantel and Lin 2000] remark that this test set is far from perfect: "For instance, 133 examples contain the word *the* as N_1 or N_2."

we will call the training material the Penn training set, the test material the Penn test set, and the collection of both the **Penn data set**.

[Ratnaparkhi et al. 1994] report on 81.6% attachment accuracy when applying their data set for training and testing. They compared their result to the attachment accuracy of 3 expert human annotators (on 300 randomly selected test events). If humans are given only the 4-tuple (V, N_1, P, N_2) without context, they achieve 88.2% accuracy, but if they are given the complete sentence their performance improves to 93.2%. This means that there is information outside the extracted 4-tuple that helps the disambiguation. [Ratnaparkhi et al. 1994] also tested 2 non-expert human annotators on 200 test events and obtained results that were 10% below the experts' judgements.

[Collins and Brooks 1995] used a statistical approach, called the **Back-off model**, in analogy to backed-off n-gram word models for speech recognition. The model uses attachment probabilities for the quadruple (V, N_1, P, N_2) computed from the Penn training set. However, it often happens that a quadruple in the application text has not been seen in the training set. In fact, 95% of the quadruples in the Penn test set are not in the training set. Therefore Collins and Brooks increase the model's robustness by computing the attachment probabilities for all triples out of each quadruple as well as all pairs. Both triples and pairs are restricted to those including the preposition. In the application of these probabilities the algorithm "backs off" step by step from quadruples to triples and to pairs until it finds a level for decision. If even the pairs do not provide any clue, the attachment probability for the preposition is used. Since the algorithm is crisp and clear, it is repeated here.

1. If $freq(V, N_1, P, N_2) > 0$

$$prob(N_{att}|V, N_1, P, N_2) = \frac{freq(N_{att}, V, N_1, P, N_2)}{freq(V, N_1, P, N_2)}$$

2. Else if $freq(V, N_1, P) + freq(V, P, N_2) + freq(N_1, P, N_2) > 0$

$$prob(N_{att}|V, N_1, P, N_2) = \frac{freq(N_{att}, V, N_1, P) + freq(N_{att}, V, P, N_2) + freq(N_{att}, N_1, P, N_2)}{freq(V, N_1, P) + freq(V, P, N_2) + freq(N_1, P, N_2)}$$

3. Else if $freq(V, P) + freq(N_1, P) + freq(P, N_2) > 0$

$$prob(N_{att}|V, N_1, P, N_2) = \frac{freq(N_{att}, V, P) + freq(N_{att}, N_1, P) + freq(N_{att}, P, N_2)}{freq(V, P) + freq(N_1, P) + freq(P, N_2)}$$

4. Else if $freq(P) > 0$

$$prob(N_{att}|V, N_1, P, N_2) = \frac{freq(N_{att}, P)}{freq(P)}$$

5. Else $prob(N_{att}|V, N_1, P, N_2) = 1.0$ (default is noun attachment)

The attachment decision is then: If $prob(N_{att}|V, N_1, P, N_2) >= 0.5$, choose noun attachment, else choose verb attachment. Collins and Brooks reported on 84.1% correct attachments, a better accuracy than in all previous research.

The application condition on each level says that the quadruple or a triple or a pair or the preposition has been seen in the training data (a minimum threshold > 0). Collins and Brooks had also experimented with setting this threshold to 5 (instead of 0), but this resulted in worse performance (81.6%). Selecting a higher threshold means cutting out low frequency

counts on a particular level and leaving the decision to a less informative level. The decrease in performance showed that low counts on a more informative level are more important than higher frequencies on lower levels.

Collins and Brooks also experimented with some simple clustering methods: replacing all 4-digit numbers by 'year' and all capitalized nouns by 'name'. These modifications resulted in a slight increase in performance (84.5%).

[Franz 1996a], [Franz 1996b] used a method based on a **loglinear model** that takes into account the interdependencies of the category features involved. The model was trained over two treebanks on all instances of PPs that were attached to VPs or NPs. Franz extracted 82,000 PPs from the Brown corpus and 50,000 PPs from the Penn Treebank (Wall Street Journal articles). Verbs and nouns were lemmatized if the base forms were attested in the corpus. Otherwise the inflected form was used. This restriction on the lemmatization helps to avoid incorrect lemmas. Another 16,000 PPs from the Penn Treebank found in a sequence V+NP+PP were reserved as test set.

Franz tested features including the preposition and its association strengths with the verb and the preceding noun as well as the noun-definiteness (introduced by [Hirst 1987] from Crain and Steedman's principle of presupposition minimization) and the type of the noun within the PP (e.g. full noun vs. proper noun vs. four-digit number interpreted as a year). The association strengths were computed as mutual information scores. It turned out that the features "preposition", its association strengths and "noun-definiteness" gave the best results. In contrast to [Hindle and Rooth 1993], Franz' algorithm learns these feature values from the Penn Treebank, but surprisingly the results were not much better. The median accuracy was 82% while his reimplementation of the Hindle and Rooth method resulted in a median accuracy of 81%.

But [Franz 1996a] also shows that his model can be extended from two to three possible attachment sites, as is the case in a sequence V+NP+NP+PP. More generally, Franz evaluated the pattern V, N_1, N_2, P, N_3. This covers sequences of a dative and an accusative NP followed by a PP but also sequences of one NP followed by two PPs. Franz reimplemented [Hindle and Rooth 1993]'s lexical association method for this case and reports a median accuracy of 72% after this method had been adapted to the particular properties of the extended case. But here Franz' loglinear model obtained a superior median accuracy of 79%, this time using only the features based on association strength (V+P, N_1+P, and N_2+P with N_2 being the noun from the second NP/PP). Note that this figure does not mean that 79% of all complete sequences were correctly structured but only that in 79% of the cases the second PP was assigned to the correct attachment site.

[Merlo et al. 1997] show that the Back-off model can be generalized to more than two attachment sites. The backing-off strategies obviously become much more complex and the sparse data problem more severe. Therefore [Merlo et al. 1997] omit the head noun in every PP and recycle the probabilities derived for the first NP for subsequent attachment sites. With this strategy they achieve 84.3% correct attachments for the first PP, replicating the result of [Collins and Brooks 1995].[8] For the second PP they achieve 69.6% correct attachments which is slightly worse than the result reported by [Franz 1996b] for this case. For the third PP the accuracy drops to 43.6%, which is still a good result considering that this PP has 4 attachment options.

[8]This is a surprising result since [Merlo et al. 1997] did not use the noun within the PP.

[Zavrel et al. 1997] employ a **memory-based learning** technique. This means storing positive examples in memory and generalizing from them using similarity metrics. The technique is a variant of the k-nearest neighbor (k-NN) classifier algorithm. The PP training instances are stored in a table with the associated correct output, i.e. the attachment decision. When a test instance is processed, the k nearest neighbours of the pattern are retrieved from the table using the similarity metric. If there is more than one nearest neighbor, the attachment decision is determined by majority voting.

The most basic metric is the Overlap metric given in the following equation. $\Delta(X, Y)$ is the distance between patterns X and Y, represented by n features. w_i is a weight for feature i and δ is the distance per feature.

$$\Delta(X,Y) = \sum_{i=1}^{n} w_i \, \delta(x_i, y_i) \qquad \text{where:} \quad \delta(x_i, y_i) = 0 \quad \text{if } x_i = y_i, \quad \text{else } 1$$

This metric counts the features that do not match between the stored pattern and the application pattern. Information Gain weighting is used to measure how much each feature contributes to the recognition of the correct attachment decision. In addition a lexical similarity measure is used to compute distributional similarity of the tokens over a corpus (3 million words). With this measure they find, for example, that the word *Japan* is similar to *China, France, Britain, Canada* etc. The similarity measure thus serves a similar purpose to a thesaurus. With this method [Zavrel et al. 1997] replicated the results from [Collins and Brooks 1995] of 84.4% correct attachments on the Penn test set. A comparison based on computational cost would thus favor the Back-off method.

[Wu and Furugori 1996] introduce a **hybrid method** with a combination of cooccurrence statistics and linguistic rules. The linguistic rules consist of syntactic or lexical cues (e.g. a passive verb indicates verb attachment for the following PPs), semantic features (e.g. a PP denoting time or date indicates verb attachment), and conceptual relations. These are relations like IMPLEMENT and POSSESSOR which are derived from the EDR Electronic Dictionary, a large property inheritance network. The cooccurrence data are derived from two large treebanks, the EDR English Corpus (160,000 sentences) and the Suzanne Corpus (130,000 words). These treebanks provide a pool of 228,000 PPs. The cooccurrence data are computed in the spirit of [Collins and Brooks 1995], backing off from triplets to pairs.

Wu and Furugori's hybrid disambiguation algorithm first tries to apply strong linguistic rules (e.g. if N_2 repeats N_1 as in *step by step* then it is a fixed expression). Second, the algorithm applies the cooccurrence data on triplets and subsequently on pairs. If the ambiguity is still not resolved, the algorithm uses concept-based disambiguation. It maps the nouns to their concept sets and applies hand-crafted rules for these sets (e.g. if motion(N_1) AND direction(N_2) then noun attachment for the PP). The authors admit that the mapping of the words to the concepts is error prone, still they report an accuracy of 84% for this step. Finally, if none of the above rules is triggered, the default attachment is determined by the general tendency of the preposition. If it attaches to the noun in more than half of the observed cases, the algorithm decides on noun attachment and else on verb attachment.

[Wu and Furugori 1996] have ordered their disambiguation steps according to decreasing reliability. The strong linguistic rules apply to 17% of their test cases but, with 96% accuracy, these rules are very reliable. Triplets cooccurrence with 92% and pair cooccurrence with 85% accuracy account for the bulk of the test cases (54%). Another 20% are handled by the conceptual rules (84% accuracy) and only 7% are left to default attachment with a low

accuracy of 70%. Overall this hybrid approach results in 86.9% attachment accuracy and is thus among the best reported figures.

[Stetina and Nagao 1997] work with an approach that is similar to [Wu and Furugori 1996] except for the hand-crafted linguistic rules. They start from the observation made by [Collins and Brooks 1995] that quadruples are more reliable than triples and pairs, but that often quadruples are not seen in the training data. They work with the Penn data set, 20,801 training and 3097 testing quadruples (V, N_1, P, N_2).

In a first step they use WordNet senses to cluster the nouns and verbs into semantically homogeneous groups. The measure of semantic distance is based on a combination of the path distance between two nodes in the WordNet graph and their depths. The problem is that many words have multiple senses in WordNet. Therefore [Stetina and Nagao 1997] used the context given by the other words in the quadruple and a similarity measure between the quadruples for sense disambiguation. An evaluation of a set of 500 words showed that their word sense disambiguation was 72% correct.

From the sense-tagged training data they induced a **decision tree** for every preposition based on the WordNet sense attributes. In addition some specific clustering was done on the training and test data (e.g. all four digit numbers were replaced by 'year', all upper case nouns not contained in WordNet were assigned the senses 'company' and 'person'). The disambiguation of test cases was done in the same way as for the training data. This approach results in 88.1% correct attachments, the best reported accuracy on the Penn test set.

Finally, there is the approach of **transformation-based learning** which uses statistical means to learn ambiguity resolution rules from a treebank [Brill and Resnik 1994].[9] The learning algorithm assigns a default attachment to every PP in the input and then derives rules based on rule-templates to reach the correct assignment as given by the parsed corpus. The rule leading to the best improvement is learned. Note that [Brill and Resnik 1994] also extend the scope of investigation to the noun N_2 within the PP. That means that they are looking at (V, N_1, P, N_2). Some examples of the rules learned by their system:

```
change attachment from N to V  if P  is 'at'
change attachment from N to V  if N2 is 'year'
change attachment from V to N  if P  is 'of'
```

The rule learning procedure is repeated until a given threshold is reached. The application phase starts with a default attachment and then applies the learned rules for modifications. [Brill and Resnik 1994] report on a rate of 80.8% correct attachments which makes their method comparable to some of the purely statistics-based methods. By adding WordNet word classes the result was improved to 81.8%. These results were achieved by training over 12,766 4-tuples from the Penn Treebank and 500 test tuples. They were confirmed by Collins and Brooks who tested the method against the Penn data set which resulted in 81.9% attachment accuracy.

The transformation-based method was extended by [Yeh and Vilain 1998]. They used an engineering approach to PP attachment, i.e. finite state parsing. They extended the scope from the V+NP+PP case to all occurring PPs. The system can look at the head-word and also at all the semantic classes the head-word can belong to (from WordNet). In addition the

[9] Transformation-based learning has successfully been employed to learn rules for part-of-speech tagging [Brill 1992].

system uses subcategorization requirements from COMLEX including prepositional requirements of verbs.

The original transformation-based disambiguation system chose between two possible attachment sites, a verb and a noun. And the method as implemented by [Yeh and Vilain 1998] resulted in 83.1% attachment accuracy on the Penn data set. Their extensions include as possible attachment sites every group that precedes the PP and result in 75.4% accuracy.

[Roth 1998] presented a unified framework for disambiguation tasks. Several language learning algorithms (e.g. backed-off estimation, transformation-based learning, decision lists) were regarded as learning linear separators in a feature space. He presented a sparse network of linear separators utilizing the Winnow learning algorithm. He modelled PP attachment as linear combinations of all 15 sub-sequences of the quadruple (V, N_1, P, N_2). Roth's method performs comparable to [Collins and Brooks 1995] on the Penn data set (83.9%).

[Abney et al. 1999] apply **boosting** to part-of-speech tagging and PP attachment. Boosting is similar to transformation-based learning. The idea is to combine many simple rules in a principled manner to produce an accurate classification. Boosting maintains an explicit measure of how difficult particular training examples are.

In the PP attachment task the boosting method learns attachment hypotheses for any combination of the features, i.e. any combination of the words in each training set. This means that it finds a hypothesis for the preposition by itself, for the preposition with N_1, for preposition, N_1 and N_2 and so on. In the experiments [Abney et al. 1999] found that the preposition *of* has the strongest attachment preference to a noun whereas *to* has the strongest preference for verb attachment. The strongest evidence for attachment decisions was provided by 4-tuples (V, N_1, P, N_2) which corresponds to our intuitions. The boosting experiments resulted in the same attachment accuracy as [Collins and Brooks 1995] on the Penn data set (84.5%).

Table 2.1 summarizes the results of the supervised methods. The 84% result seems to be the maximum performance for supervised methods without employment of a thesaurus. This result was first achieved by [Collins and Brooks 1995] and later replicated by [Zavrel et al. 1997], [Roth 1998] and [Abney et al. 1999]. We will report on our experiments with the Back-off method for German in section 7.2.1. Accessing a thesaurus for clustering of the nouns improves the performance by up to 4%, as has been demonstrated by [Wu and Furugori 1996] and [Stetina and Nagao 1997].

2.2.2 Unsupervised Methods

The statistical methods introduced in the previous section learned their attachment preferences from manually controlled data, mostly from the Penn Treebank. Following [Hindle and Rooth 1993] more unsupervised learning methods have been proposed. Unsupervised learning exploits regularities from raw corpora or automatically annotated corpora.

[Ratnaparkhi 1998] uses heuristics to extract unambiguous PPs with their attachments from a large corpus (970,000 sentences from the Wall Street Journal). The extraction procedure uses a part-of-speech tagger, a simple chunker and a lemmatizer. The heuristics are based on the fact that in English "the attachment site of a preposition is usually located only a few words to the left of the preposition". This means roughly that a PP is considered as unambiguous verb attachment if the verb occurs within a limited number of words to the left of the preposition and there is no noun in between. This is obviously a good criterion.

Author	Method	Resource	Scope	Results
[Ratnaparkhi et al. 1994]	Maximum entropy model	treebank	V+N_1+P+N_2	81.6%
[Brill and Resnik 1994]	Transformation rules	treebank	V+N_1+P+N_2	80%
[Collins and Brooks 1995]	Quadruple, triple, pair probabilities with Back-off model	treebank	V+N_1+P+N_2	84.5%
[Franz 1996a]	Lexical Association plus noun-definiteness in a loglinear model	treebank	V+N_1+P V+N_1+P+P	82% 79%
[Wu and Furugori 1996]	Quadruple, triple, pair probabilities with Back-off model, combined with linguistic rules	treebank, EDR electronic dictionary	V+N_1+P	86.9%
[Zavrel et al. 1997]	Memory-based learning	treebank	V+N_1+P+N_2	84.4%
[Merlo et al. 1997]	Quintuple, quadruple etc. probabilities with generalized Back-off model	treebank	V+N_1+P V+N_1+P+P V+N_1+P+P+P	84.3% 69.6% 43.6%
[Stetina and Nagao 1997]	Decision tree	treebank, WordNet	V+N_1+P+N_2	88.1%
[Yeh and Vilain 1998]	Transformation rules	treebank, WordNet, Comlex	V+N_1+...+N_n+P+N_m	75.4%
[Roth 1998]	Learning linear separators	treebank	V+N_1+P+N_2	83.9%
[Abney et al. 1999]	Boosting	treebank	V+N_1+P+N_2	84.5%

Table 2.1: Overview of the supervised statistical methods for PP attachment

Finding unambiguous noun attachments is more difficult. It is approximated by an analogous rule stating that the PP is considered as unambiguous noun attachment if the noun occurs within a limited number of words to the left of the preposition and there is no verb in between. These heuristics lead to 69% correct attachments as measured against the Penn Treebank. The noise in these data is compensated by the abundance of training material.

From the extracted material Ratnaparkhi computes bigram counts and word counts and uses them to compute the cooccurrence statistics. His disambiguation algorithm marks all *of*-PPs as noun attachment and follows the stronger cooccurrence value in all other cases. This approach results in 81.9% attachment accuracy (evaluated against the Penn test set). In a second set of experiments the same procedure was used for a small Spanish test set (257 test cases). It resulted in even better accuracy (94.5%). In our experiments for German in chapter 4 we will use a variant of the Ratnaparkhi cooccurrence measure.

[Li and Abe 1998] discuss PP attachment in connection with their work on the acquisition

of case frame patterns (subcategorization patterns). Case frame pattern acquisition consists of two phases: extraction of case frame instances from a corpus and generalization of those instances to patterns. Obviously, generalization is the more challenging task that has not been solved completely to date. [Li and Abe 1998] employ the Minimal Description Length principle from information theory.

In order to increase the efficiency they use WordNet to focus on partitions that are cuts in the thesaurus tree. Their algorithm obtains the optimal tree cut model for the given frequency data of a case slot in the sense of Minimal Description Length.

We first assumed that [Li and Abe 1998] will use PP complements identified in case frames as predictors for PP attachment. But that is not so. Instead they estimate $P(N_2|V,P)$ and $P(N_2|N_1,P)$ from the training data consisting of triples. If the former exceeds the latter (by a certain margin), they decide in favor of verb attachment. Analogously they decide on noun attachment. For the remaining cases that are ruled out by the margin they use verb attachment as default.

The triples were extracted from the Penn Treebank (Wall Street Journal corpus), and 12 heuristic rules were applied to cluster and simplify the data. All word forms were lemmatized, four digit integers in the range 1900 to 2999 were replaced by the word *year* and so on. Finally, noun N_2 was generalized using WordNet and the Minimal Description Length principle. In the disambiguation process they compared $P(class_1|V,P)$ and $P(class_2|N_1,P)$ where $class_1$ and $class_2$ are classes in the tree cut model dominating N_2. The result is 82.2% attachment accuracy.[10]

[Pantel and Lin 2000] use a collocation database, a corpus-based thesaurus and a 125-million word newspaper corpus. The newspaper corpus is parsed with a dependency tree parser. Then unambiguous data sets consisting of (V,N_1,P,N_2) are extracted.

Attachment scores for verb attachment and noun attachment are computed by using linear combinations of prior probabilities for $prob(P), prob(V,P,N_2), prob(N_1,P,N_2)$ and conditional probabilities for $prob(V,P|V), prob(N_1,P|N_1), prob(P,N_2|N_2)$. For example, the prior probability $prob(V,P,N_2)$ is computed as

$$prob(V,P,N_2) = log\ \frac{freq(V,P,N_2)}{freq(\text{all unambiguous triples})}$$

and the conditional probability $prob(V,P|V)$ is computed as:

$$prob(V,P|V) = log\ \frac{freq(V,P)}{freq(V)}$$

The attachment scores are then defined as:[11]

$$VScore(V,P,N_2) = prob(V,P,N_2) + prob(V,P|V)$$

$$NScore(N_1,P,N_2) = prob(N_1,P,N_2) + prob(N_1,P|N1)$$

[10]The approach by [Li and Abe 1998] is similar to the approach described in [Resnik 1993] yields better results.

[11]In the paper both score formulae contained $prob(P)$ and $prob(P,N_2|N_2)$. Since these values are not influenced by V and N_1, they will be identical and can thus be omitted.

For each test case "raw" attachment scores are computed for the words occurring in the quadruple. In addition, contextually similar words are computed for the verb V, and for N_1 and N_2 using the collocation database and the thesaurus, both of which had been automatically computed from the corpus. Using the similar words, another pair of attachment scores is computed for each test case based on the above formula. This attachment score represents the average attachment score of all the words in the word class.

Finally, the raw and the average scores are combined both for verb attachment and noun attachment. The attachment decision is won by the higher score (all *of*-PPs are noun attachments). [Pantel and Lin 2000] report on 84.3% correct attachments when testing on the Penn test set.

The unsupervised approaches are summarized in table 2.2. It should be noted that the comparison of the results is difficult if the test sets differ. [Ratnaparkhi 1998], [Li and Abe 1998], and [Pantel and Lin 2000] have evaluated their methods against the Penn test set whereas [Hindle and Rooth 1993] used a smaller test set.

Author	Method	Resource	Scope	Results
[Hindle and Rooth 1993]	Lexical Association	shallow parsed corpus	V+N_1+P	80%
[Ratnaparkhi 1998]	Pair cooccurrence values over unambiguous PPs	shallow parsed corpus	V+N_1+P	81.9%
[Li and Abe 1998]	Triple cooccurrence values with generalization of N_2	corpus, WordNet	V+N_1+P+N_2	82.2%
[Pantel and Lin 2000]	Attachment scores over unambiguous PPs, contextually similar words	collocation database, thesaurus, large dependency parsed corpus	V+N_1+P+N_2	84.3%

Table 2.2: Overview of the unsupervised statistical methods for PP attachment

2.3 Ambiguity Resolution with Neural Networks

[Alegre et al. 1999] use multiple neural networks to resolve PP attachment ambiguities. As usual the neural networks were used for supervised learning. They work with the Penn training set (20,801 4-tuples) and test set (3,097 4-tuples). The input was divided into 8 slots: (1-4) the quadruples from the data set, (5) the prepositions that the verb subcategorized (taken from COMLEX and the training set), (6) the prepositions that the noun N_1 subcategorized (from the training set), (7) WordNet classes, and (8) information on whether N_1 and N_2 are proper nouns. Since "Using words alone ... floods the memory capacity of a neural network", [Alegre et al. 1999] build word classes. All numbers were replaced by the string "whole_number". All verbs and nouns were reduced to their base form. Proper nouns were replaced by WordNet class names like PERSON or BUSINESS_ORGANIZATION. Rare prepositions were omitted. With these somewhat cleaned data they achieve 86% accuracy, comparable to the supervised statistical approaches that exploit WordNet.

2.4 PP Ambiguity Resolution for German

An early book on German prepositions in natural language processing is [Schweisthal 1971]. He started with a linguistic classification of the prepositions into temporal, local and others. In addition he collected a small lexicon of German nouns which he sorted into the same semantic classes. Local nouns comprise names of cities and countries but as subclasses also institutions (*Post, Polizei, Universität*) and materials (*Gold, Kupfer, Öl, Butter*). Temporal nouns are names of days, months and seasons as well as public holidays (*Ostern, Weihnachten, Neujahr*). Schweisthal showed that this semantic classification makes possibles the computation of one piece of information given the other two pieces out of:

1. the preposition
2. the semantic noun class (*Nomeninhaltsfunktionsklasse*)
3. the representation of the semantic content of the PP

[Schweisthal 1971] demonstrated his approach by automatically generating PPs for the prepositions *vor* and *nach* with all nouns in his lexicon. He also showed that this semantic classification serves to disambiguate PPs in machine translation. His experimental system correctly translated *nach dem Spiel* as *after the game* and *nach Köln* as *to Cologne*.

The book also touches on the subject of PP attachment. Schweisthal tackled this problem with

1. a list of 4000 verbs with their prepositional complements (*Verbbindungen*). The complements were classified as primary, which corresponds to true complements (graded by Schweisthal as necessary, implied, and expected), and secondary, which more or less corresponds to obligatory and optional adjuncts.
2. a list of 1400 nouns with prepositional complements (most of them deverbals).
3. a list of 4000 idioms which contain prepositions.
4. a list of support verb units.

These lists represented a large-scale collection for these early days of natural language processing. Unfortunately, our attempts to get hold of these resources from the University of Bonn were not successful. The data seem to have been lost over the years.

Since then there have been few publications that specifically address PP ambiguity resolution for German. We suspect that this is in large part due to the fact that until recently there was no German treebank available. Without a treebank, testing an approach to ambiguity resolution was cumbersome, and supervised learning of attachment decisions was impossible. In 1999 the NEGRA project at the University of Saarbrücken published its German treebank with 10,000 sentences from general newspaper texts. The sentences are annotated with a flat syntactic structure [Skut et al. 1997] and are thus a valuable resource for testing, but this corpus is still too small for statistical learning. In section 3.2.1 we describe how we extract the appropriate information from this treebank to establish a test set for PP attachments.

Some papers tackling the PP ambiguity problem for German are compiled in [Mehl et al. 1996]: There, [Hanrieder 1996] describes a method for integrating PP attachment in

a unification-based left-associative grammar. Syntactic and semantic information is hand-coded into complex nested feature structures that are unified during parsing according to the grammar rules. PP ambiguities are resolved based on the head-attachment principle (as proposed by [Konieczny et al. 1991]) which states that a constituent will be attached to a head that is already present in left to right processing. Head-attachment predicts noun attachment for PPs in the *Mittelfeld* of German matrix clauses (as in example 2.20) if the full verb is located in the right clause bracket (i.e. behind the *Mittelfeld*) and thus becomes available for attachment after the processing of the PP. We assume the same attachment prediction for the separated prefix case in which the truncated verb is in the left bracket position but the full verb can only be "assembled" after the prefix is found in the right bracket (as in 2.21).

According to [Konieczny et al. 1991] head-attachment does not predict the attachment for the corresponding sentences with the full verb in the left bracket position (as in 2.22). But [Hanrieder 1996] interprets it as suggesting a preference for verb attachment in this case. This is not convincing.

(2.20) *Sony hat auf einem Symposium* ***in San Francisco*** *eine neuartige Zelltechnologie vorgestellt.*

(2.21) *Sony stellt auf einem Symposium* ***in San Francisco*** *eine neuartige Zelltechnologie vor.*

(2.22) *Sony präsentiert auf einem Symposium* ***in San Francisco*** *eine neuartige Zelltechnologie.*

In addition, the head-attachment principle will certainly be superseded by subcategorization requirements of the verb (coded as constraints in Hanrieder's feature structures). The approach shows in a nutshell the possibilities and limits of hand-crafting deep linguistic knowledge in combination with global attachment principles.

In the same collection [Langer 1996] introduces his GEPARD parser for German. It is a wide coverage parsing system based on more than 1000 hand-crafted rules. (The GEPARD project is further elaborated in [Langer 1999].) Langer points to the important role of the lexicon as a place for coding prepositional requirements including support verb units which he sees as complex requirements of the verb. He also reports on fine-grained grammatical regularities that help to decide on the correct PP attachment. For instance he notes that particles like *nur, sogar* prohibit a noun attachment of the following PP.

(2.23) *Ihr hoher Preis hat am Anfang ihren Einsatz in den USA* ***nur auf den Verteidigungsbereich*** *beschränkt.*

But GEPARD includes not only lexical and grammatical constraints but also a probabilistic model for the remaining ambiguities. Langer exemplifies for the preposition *mit* how his parser incorporates probabilistic attachment values. He uses unsupervised learning over a 10 million word newspaper corpus. He computes the attachment tendency of the preposition towards the noun as

$$np\_attach(N_i) = \frac{prob(N_i|P)}{prob(N_i)}$$

The **np_attach** measure has the neutral value 1 if the probability for the noun is equal to the conditional probability of the noun, given the preposition. If the value is 10, the noun occurs 10 times more frequently in a sequence with the particular preposition than could be expected and thus there is a tendency for noun attachment. If it is below the neutral value, the PP is attached to the verb. A small evaluation of this measure in [Langer et al. 1997] speaks of 71% correct attachments. No real size evaluation of this measure has been reported.

In [Mehl et al. 1998] we reported the first results of our experiments with the cooccurrence value. We had manually disambiguated 500 sentences that contained the preposition *mit*. The cooccurrence value method which will be discussed in detail in chapter 4 resulted in 74.2% correct attachments on this small test set.

[de Lima 1997] describes an interesting approach using pronominal adverbs to find German prepositional subcategorization information. This is obviously only one first step towards PP attachment determination, but due to its unorthodox approach, we will briefly summarize it here. The approach is based on the hypothesis that "pronominal adverbs are high-accuracy cues for prepositional subcategorization" since they substitute complements but not adjuncts.

Lima uses shallow parsing to find NPs (with their grammatical case), PPs, adjectival phrases and clause boundaries. Only "correlative construct main clauses" were considered, that is main clauses containing a pronominal adverb.

(2.24) *Und die Entwickler denken bereits* ***daran****, ...*

(2.25) *Wir haben uns zunächst* ***darauf*** *konzentriert, daß ...*

In a 36 million word newspaper corpus (*Frankfurter Allgemeine Zeitung*) she finds 16,795 such clauses. Each shallow parsed clause is mapped to one of five subcategorization templates. Ambiguously positioned pronominal adverbs (5581 out of the 16,795 sentences) are mapped to all possible templates. Passive sentences were transformed into active ones and mapped accordingly. All frames were ranked using an expectation maximization algorithm. 400 of the ambiguous sets were manually judged and resulted in 85% attachment accuracy. The errors were traced to factors such as the mixing up of reflexive and non-reflexive readings, but also to pronominal adverbs which are homographs with conjunctions and adverbs (*dabei, danach*).

Lima also compared the verbs of her "acquired dictionary" (verbs plus prepositional subcat requirement) to a broad coverage published dictionary (Wahrig). A random set of 300 verbs (each occurring more than 1000 times in the corpus) was selected and compared. For these 300 verbs both dictionaries listed 307 verbal preposition frames. But 136 of these were only in the published dictionary and 121 only in the automatically acquired dictionary. Of course, this divergence could be attributed to erroneous and missing subcat frames in the published dictionary. And therefore a true evaluation will have to employ the automatically computed frames in PP ambiguity resolution.

An interesting study on German PP attachments is [Hartrumpf 1999] who extended the work by [Schulz et al. 1997] which we described in section 2.1. Hartrumpf tries to solve the PP attachment problem together with the PP interpretation problem. PP interpretation refers to the semantic interpretation of the PP as e.g. local, temporal, or causal. Hartrumpf combines hand-crafted interpretation rules and statistical evidence. The interpretation rules use a set of feature structure constraints in their premise. The features include syntactic case and number as well as semantic sorts from a predefined ontology. The conclusion of an interpretation rule is the semantic interpretation of the PP. The approach considers all possible mothers

for a PP. The disambiguation works in three steps: application of the interpretation rules, interpretation disambiguation based on relative frequencies over semantic interpretations, and attachment disambiguation, again based on relative frequencies and a distance scoring function (the number of words between the candidate mother and the PP).

Hartrumpf uses cross validation on a small corpus of 720 sentences (120 each for 6 prepositions). Problematic cases like complex named entities, elliptic phrases, foreign language expressions and idioms were excluded from the corpus. He reports on 88.6% (preposition *auf*) to 94.4% (preposition *wegen*) both correct attachment and interpretation for binary attachment ambiguities and 85.6% to 90.8% correct attachment and interpretation overall (the average being 87.7%). These are very impressive results bought at the cost of hand-crafted semantic rules and semantic lexical entries. They show that semantic information does indeed improve the resolution of attachment ambiguities but requires a lot of time-consuming manual labor.

Disambiguation in natural language processing has been called an AI-complete task. This means that all types of knowledge required to solve AI problems will also be required for disambiguation. In the end, disambiguation in language processing requires an understanding of the meaning. The computer can only approximate the behavior of an understanding human. And in order to do so it needs all the information it can get. For PP attachment this means that the computer should have access to both linguistic (syntactic, semantic) and statistical information.

This survey has shown that the best results for the wide coverage resolution of PP attachment ambiguities are based on supervised learning in combination with semantically oriented clustering. Thesaurus relations helped to approximate human performance for this task. Since a large German treebank is not available, we will explore an unsupervised learning method in this book. But we will make sure to enrich our training corpus with as much linguistic information as is currently possible with automatic procedures.

Chapter 3

Corpus Preparation

3.1 Preparation of the Training Corpus

Our method for the disambiguation of PP-attachment ambiguities relies on competing cooccurrence strengths between the noun and the preposition (N+P) and the verb and the preposition (V+P). Therefore we have computed these cooccurrence strengths from a corpus. We chose to work on a computer magazine corpus since it consists of semi-technical texts which displays features from newspapers (some articles are very short) and from technical texts (such as many abbreviations, company and product names). We selected the Computer-Zeitung [Konradin-Verlag 1998], a weekly computer magazine, and worked with 4 annual volumes (1993-95 and 1997). The 1996 volume was left for the extraction of test material. The raw texts contain around 1.4 million tokens per year. Here are the exact figures as given by the UNIX word count function:

year	number of tokens
1993	1,326,311
1994	1,444,137
1995	1,360,569
1997	1,343,046
total	5,474,063

The Computer-Zeitung (CZ) contains articles about companies, products and people in information technology. The articles range from short notes to page-long stories. They include editorials, interviews and biographical stories. The articles are text oriented, and there are few graphics, tables or diagrams. The newspaper aims at a broad readership of computer professionals. It is not a scientific publication; there are no cross references to other publications. The general vocabulary is on the level of a well-edited daily newspaper (comparable to e.g. Süddeutsche Zeitung or Neue Zürcher Zeitung), but due to its focus on technology it additionally contains a wealth of specific words referring to hardware and software as well as company and product names.

As examples we present a typical company news article and a typical product information article in the following two text boxes. Both articles are introduced by an identifier line stating the number, the year and the page of publication. This line is followed by one or two header lines. The company news article starts with a city anchor and an author acronym in

parentheses. Both articles show some typical examples of company names, city names and product names.

CZ 39/1993, S. 1
Weniger Produkte
Debis speckt ab
Stuttgart (gw) - Bei der Daimler-Benz-Tochter Debis Systemhaus ist nach zahlreichen Firmenaufkäufen und Beteiligungen jetzt Großreinemachen angesagt. Bereits im Frühjahr war die Arbeit an einer Standardanwendungssoftware à la SAPs R/3 gestoppt worden. Jetzt wurden die eigenentwickelten Softwareprodukte für den dezentralen Bankenbereich aus der Produktpalette gestrichen. Damit will sich das Systemhaus offenbar weiter von unrentablen Einheiten trennen, die sich durch die vielen Firmenaufkäufe in der Vergangenheit angehäuft hatten, und sich verstärkt auf das Projektgeschäft konzentrieren.

CZ 39/1993, S. 11
Steckbarer Rechner
Die neuen Hochleistungscomputer von Motorola, Hamburg, arbeiten als Unix-Mehrplatzsysteme oder als Server in verteilten Client-Server-Umgebungen. Maximal 1000 Benutzer werden mit Leistung versorgt. Die Computermodelle bestehen aus einzelnen Modulen, die laut Motorola durch Einrasttechnik ohne Werkzeug "innerhalb weniger Minuten" zusammengesetzt werden, anschließend die neue Konfiguration selbständig erkennen. Als Prozessor wird der M88110 mit einer Taktfrequenz von 50 Megahertz eingesetzt, in der Einstiegsversion der 88100 mit 33 Megahertz. Geliefert werden Prozessor-, VME- sowie SCSI-Erweiterungsmodule. Insgesamt sechs verschiedene Singleprozessormodelle sind lieferbar, ein Multiprozessorsystem kommt im Oktober und die Vierprozessorversion Anfang 1994.

Articles in the CZ are on average 20.1 sentences long (including document headers; standard deviation 18.6), while the average sentence length is 15.7 words (including punctuation symbols but excluding XML tags; standard deviation 9.6). Figure 3.1 shows a plot of the sentence length distribution. There is a first peak at length 2. This includes short headers (*Hardware-Umsätze stagnieren*) and turn-taking indicators in interviews (*Niedermaier :* vs. *CZ :*). The second peak is at sentence length 14, close to the average sentence length of 15.7. In section 5 we will compare these values with the Neue Zürcher Zeitung, a general newspaper.

3.1.1 General Corpus Preparation

Our corpus documents are distributed via CD-ROM. All texts are in pure text format. There is no formatting information except for a special string that marks the beginning of an article. In order to compute the cooccurrence values, the corpora had to be processed in various steps. All programming was done in Perl.

1. **Clean-up.** The texts have been dehyphenated by the publisher before they were distributed (with few exceptions). Some internet addresses (mostly ftp and http addresses) still contained blanks. These blanks (represented here by the symbol ␣) were eliminated to recognize the addresses as one unit.

   (3.1) **Before:** *Eine Liste der erreichbaren Bibliotheken bietet "http://www.␣laum.uni-hannover.de/iln/bibliotheken/bibliotheken.␣html".*

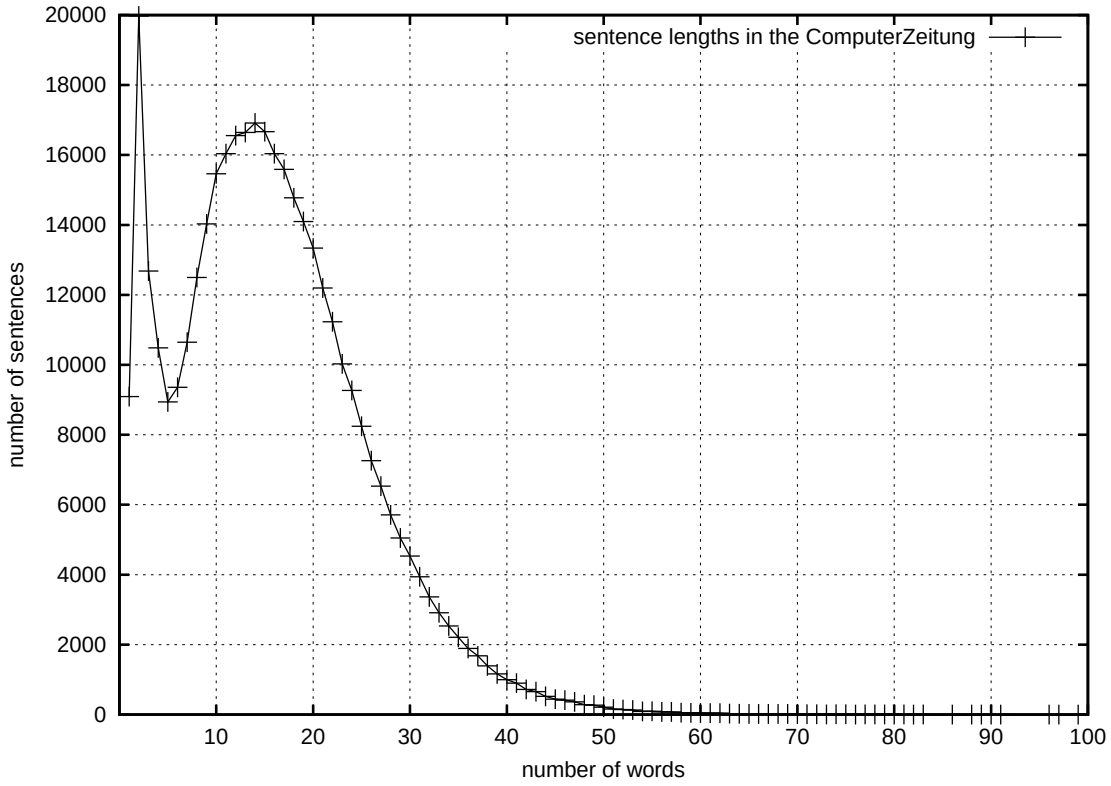


Figure 3.1: Sentence length distribution in the CZ corpus

(3.2) **After:** *Eine Liste der erreichbaren Bibliotheken bietet "http://www.laum.uni-hannover.de/iln/bibliotheken/bibliotheken.html".*

There are other blanks that are not token delimiters. Our corpus contains blanks within long sequences of digits such as numbers over 10␣000 and telephone numbers. We substitute these blanks with auxiliary symbols (e.g. a dash) to facilitate tokenization.

(3.3) **Before:** *Weitere Informationen unter der Telefonnummer 0911/96␣73-156.*

(3.4) **After:** *Weitere Informationen unter der Telefonnummer 0911/96-73-156.*

In general, there are no line breaks within or after sentences but only at the end of paragraphs. But there is a substantial number of misplaced line breaks that violate this rule. Some of these can be automatically detected and eliminated. For example, a line break after a comma will not be a correct paragraph end and can be eliminated.

2. **Recognition of text structure.** Headlines are often elliptical sentences (*Mehrwert gefunden, Arbeitsplätze gesucht*). They cause many tagging errors since the part-of-speech tagger has been trained over complete sentences. Therefore we have to recognize and mark headers, regular paragraphs, and list items in order to treat them specifically.

A header is a line that ends without a sentence-final punctuation marker. A list item starts with a '-' as the first symbol in a line. We use SGML tags to mark these items (⟨h2⟩, ⟨li⟩) and all other meta-information (e.g. document boundaries and document identifiers).

We identify newspaper-specific article starters. Most articles begin with a city name and an abbreviation symbol for the author.

(3.5) *Bonn (pg) - Bundesregierung und SPD kamen sich ...*

These can be recognized with a pattern matcher and marked with ⟨city⟩ and ⟨author⟩ tags. In this way we make this information explicit for further processing steps. For example, we may later want to delete all headers, city names and author identifiers from the texts since they do not contribute to finding PP attachment statistics.

3. **Recognition of sentence boundaries.** Sentences end at the end of a paragraph or with a sentence-finishing punctuation symbol (a full stop, an exclamation mark, a question mark). Unfortunately, a full stop symbol is identical to a dot that ends an abbreviation (*zusammen mit Dr. Neuhaus*) or an ordinal number (*Auf dem 5. Deutschen Softwaretag*). We use an abbreviation list with 1200 German abbreviations to distinguish the dot from a full stop. In addition we assume that a German word consists of at least two letters. Thus we identify one-letter abbreviations (*will es Raymond J. Lane*). If a number or an abbreviation is in sentence final position, we will miss the sentence boundary in this step. We partly correct these errors after part-of-speech tagging (cf. section 3.1.3).

4. **Verticalization of the text.** The text is then verticalized (one word per line). Punctuation marks are considered separate tokens and thus also occupy a separate line each. According to the UNIX word count function, our texts now contain more tokens than at the start. By deleting some blanks in the clean-up, some token pairs have been connected to one token, but the punctuation marks are now counted as separate tokens. The following table shows the number of tokens per annual volume.

year	number of tokens	SGML tags in tokens	ratio of SGML tags
1993	1,591,424	82,159	0.0516
1994	1,728,462	89,906	0.0520
1995	1,632,731	87,696	0.0537
1997	1,630,088	106,309	0.0652
total	6,582,705	366,070	

The SGML tags that mark the document and text structure account for 5-6% of all tokens. It is striking that the ratio of SGML tags to all tokens increases over the years. This means that the texts were structured into smaller units (more list items and shorter articles).

3.1.2 Recognition and Classification of Named Entities

At some point during corpus processing we need to recognize and classify proper names for persons, geographical locations, and companies.[1] We will use this information later on to

[1]Part of this section was published as [Volk and Clematide 2001].

form semantic classes for all name types. A class name will stand for all members of the class. It will be used to reduce the sparse data problem and subsequently the noise in our frequency counts.

One could argue that the recognition of proper names is a task for a part-of-speech tagger and consequently classification should be done after tagging. But [Volk and Schneider 1998] have shown that a tagger's distinction between proper names and regular nouns is not reliable in German. The confusion between these two types of noun is the main source of tagging errors. In German both proper names and regular nouns are spelled with an initial capital letter and their distributional properties are not distinct enough to warrant a clear tagger judgement. One would guess that proper names are less likely to occur without a determiner. But there are many cases in which regular nouns occur without a determiner (plural forms, singular forms of mass nouns, coordinations and listings). Therefore we decided to recognize and classify named entities before tagging. All recognized names will be reported to the tagger, in this way reducing the number of cases for which the tagger has to tackle the difficult task of noun classification.

Named entity recognition is a topic of active research, especially in the context of message understanding and classification. In the message understanding conference [MUC 1998] the best performing system achieved an F-measure of 93.39% (broken down as 91% precision and 90% recall). This includes the classification of person, organization, location, date, time, money and percent. The first three (person, organization, location) are the core of the task while the others can be recognized by regular expressions and short lists of keywords for month names and currencies.

The approaches described in the technical literature use internal evidence (keywords[2], name lists, gazetteers) or external evidence (the context). If a token (or a sequence of text tokens) from the text under investigation is listed in a name list, the task of name recognition is a special case of word sense disambiguation (if there is a competing reading). But general techniques for word sense disambiguation, such as using lexicon definitions or thesaurus relations (cf. [Wilks and Stevenson 1997] and [Wilks and Stevenson 1998]), can only seldom be used since in most cases a proper name will not be listed in lexicon or thesaurus.

Of course, most problematic is the classification of unknown names (as is the classification of any word not listed in lexicon). Different algorithms have been used to learn names and their classification from annotated texts and from raw corpora.

An example for the usage of internal evidence is the SPARSER system [McDonald 1996]. It does proper name classification in 3 steps: delimit (sequences of capitalized words), classify (based on internal evidence), and record (proper name in its entirety but also its constituents). No evaluation figures are given.

An example for the extensive usage of external information is described by [Cucchiarelli et al. 1999]. They use an unsupervised method to classify proper names based on context similarity. In a first step they employ a shallow parser to find elementary syntactic relations (such as subject-object). They then combine all syntactic relations found in one document with the same unknown word, using the one-sense-per-document hypothesis. They compare these combined contexts to all other contexts of known names (which are provided in a start-up gazetteer). They achieve good results for the classification of organization, location and person names (80% to 100% precision) but report problems with product names.

[2]Such keywords are sometimes called trigger words.

We will use both internal and external evidence, and hence our approach is similar to the LaSIE system [Stevenson and Gaizauskas 2000]. It combines list lookup, part-of-speech tagging, name parsing and name matching. While [Stevenson and Gaizauskas 2000] have experimented with learning name lists from annotated corpora, we will learn them from our unannotated corpus. They show that carefully compiled lists cleaned through dictionary filtering and probability filtering lead to the best results. Dictionary filtering means removing list items which also occur as entries in the dictionary. But this should only be done if a word occurs more frequently in the annotated data as non-name than as name (probability filtering).

These approaches have taken a binary decision as to whether a token is a proper name or not. [Mani and MacMillan 1996] stress the importance of representing uncertainty about name hypotheses. Their system exploits the textual structure of documents to classify names and to tackle coreference. In particular it exploits appositives to determine name categories (e.g. *X, a small Bay Area town* $\rightarrow X =$ city name). A newly introduced name leads to the generation of a normalized name, name elements and abbreviations so that these forms are available for coreference matching. The system works in two passes. It first builds hypotheses on name chunks (sequences of capitalized words). Second, it groups these name chunks into longer names if there are intervening prepositions or conjunctions. They report on 85% precision and 67% recall on 42 hand-tagged Wall Street Journal articles with 2075 names.

While most of the described approaches use a schema of 4 or 5 name types, [Paik et al. 1996] describe a system with a very elaborate classification schema. It contains 30 name classes in 9 groups. For instance, the group "organization" contains company names, government organizations and other organizations. Their name classifier works with much the same methods as the previously described ones. In addition, they make intensive use of name prefixes, infixes and suffixes. They also use a partial string match for coreference resolution. They performed a rather small evaluation on Wall Street Journal articles with a test set of 589 names. They claim to achieve 93% precision and 90% recall. This result is suprising considering the wide variety of name classes.

Most of the research on the classification of named entities is for English. In particular, there are very few publications on German name recognition. One is [Langer 1999] describing briefly the PRONTO system for person name recognition. He uses a combination of first name lists, last name lists, heuristics ("a capitalized word following a first name is a last name"), context information ("a determiner in front of a hypothetical person name cancels this hypothesis") and typical letter trigrams over last names. He reports precision and recall figures of 80%.

Recognition of person names

It is close to impossible to list the full names of everybody in the world. It is a fruitless task anyway since people change their names (e.g. when they get married) and new people are constantly born and named. Even if one could access the world's telephone book (if there were such a worldwide database), one would have to deal with different writing systems or transliterations. Therefore we need to find a more pragmatic approach to the problem of proper name recognition. One observation is that there is a rather stable set of personal first names. Second, we find that a person's last name is usually introduced in a text with either his/her first name, a title (Dr., Prof., Sir), or a word describing his/her profession or function

(manager, director, developer).[3]

Therefore we use a list of 16,000 first names and another list of a dozen titles as keywords to find such name pairs (keyword followed by a capitalized word). The name list contains mostly German and English first names with many different spelling variations (e.g. *Jörg, Joerg, Jürg, Jürgen*). It is derived from searching through machine readable telephone books. Our recognition program "learns" the last name, a capitalized word that follows the first name. The last name will then be used if it occurs standing alone in subsequent sentences.

(3.6) *Beim ersten Internet-Chat-in von EU-Kulturkommissar* ***Marcelino Oreja*** *mußten die Griechen "leider draußen bleiben".* ***Oreja****, ..., beantwortete unter Zuhilfenahme von elf Übersetzern bis zu 80 Anfragen pro Stunde.*

This approach, however, leads to two problems. First, the program may incorrectly learn a last name if e.g. it misinterprets a company name (*Harris Computer Systems*), or if there is a first name preceding a regular noun (*... weil Martin Software entwickelt*). Second, a last name correctly learned in the given context might not be a last name in all subsequent cases (consider the person name *Michael Dell* and the company name *Dell*). Applying an incorrectly learned last name in all subsequent occurrences in the corpus might lead to hundreds of erroneously recognized names.

Therefore we use the observation that a person name is usually introduced in a document in either full form (i.e. first name and last name) or with a title or job function word. The last name is thereafter primed for a certain number of sentences in which it can be used standing alone. If it is used again later in the text, it needs to be reintroduced. So, the question is, for how many sentences does the priming hold. We use an initial value of 15 and a refresh value of 5. This means that a full name being introduced is activated for 15 subsequent sentences. In fact, its activation level is reduced by 1 in every following sentence. After 15 sentences the program "**forgets**" the name. If, within these 15 sentences, the last name occurs standing alone, the activation level increases by 5 and thus keeps that name active for 5 more sentences.

```
foreach sentence {
  if match(full_name(first_name|title, last_name)) {
    activation_level(last_name) += 15;
  }
  elsif match(last_name) && (activation_level(last_name) > 0) {
    activation_level(last_name) += 5;
  }
  elsif end_of_document {
    foreach last_name {
      activation_level(last_name) = 0;
    }
  }
  else {  ## sentence without last_name
    foreach last_name {
      if activation_level(last_name) > 0 {
```

[3]Of course, we also have to take into consideration a middle initial or a honorific preposition (*von, van, de*) between the first and the last name.

```
        activation_level(last_name)--;
      }
    }
  }
}
```

We found the initial activation value by counting the number of sentences between the introduction of a full name and the subsequent reuse of the last name standing alone. In an annual volume of our corpus we found 2160 full names with a reused last name in the same document. In around 50% of the cases, the reuse happens within the following two sentences. But the reuse span may stretch up to 30 sentences. With an initial activation value of 10 we miss 7%, but with a value of 15 only 3% of reused names. We therefore decided to set this level to 15. We also experimented with a lower refresh value of 2. Against our test set we found that we are losing about 10% recall and therefore kept the refresh value at 5.

In another experiment we checked all documents of an annual volume of our corpus for recognized last names that reoccur later on in the document without being recognized as last names. For an initial activation value of 10 we found 209 such last name tokens in 6027 documents. The initial value of 15 only resulted in 98 unrecognized last name tokens (about 1% improved recall) with only 6 erroneously recognized items (a negligible loss in precision).

With this priming algorithm we delimit the effect of erroneously learned last names to the priming area of the last name. The priming area ends in any case at the end of the document. Note that this algorithm allows a name to belong to different classes within the same document. We have observed this in our corpus especially when a company name is derived from its founder's name and both are mentioned in the same document.

(3.7) *Der SAP-Konkurrent **Baan** verfolgt eine aggressive Wachstumsstrategie. ... Das Konzept des Firmengründers **Jan Baan** hat Erfolg.*

These findings contradict the one-sense-per-document hypothesis brought forward by [Gale et al. 1992]. They had claimed that it is possible to combine all contextual evidence of all occurrences of a proper name from one document to strengthen the evidence for the classification. But in our corpus we find dozens of documents in every annual volume where their hypothesis does not hold.

Included in our algorithm is the use of the genitive form of every last name (ending in the suffix *-s*). Whenever the program learns a last name, it treats the genitive as a parallel form with the same activation level. Thus the program will also recognize *Kanthers* after having learned the last name *Kanther*.

(3.8) *Wie es heißt, gewinnen derzeit die Hardliner um Bundesinnenminister **Manfred Kanther** die Oberhand. ... **Kanthers** Interesse gilt der inneren Sicherheit:*

If a learned last name is also in our list of first names, our system regards it as last name for the priming span (cf. 3.9). If it occurs standing alone, it is recognized as last name if it is not followed by a capitalized word. An immediate capitalized successor will trigger the learning of a new last name. This strategy is succesful in most cases (cf. 3.10) but leads to rare errors as exemplified in 3.11. This means that a first name - last name conflict is resolved in favor of the first name. The trigger is not applied for the genitive form since this form as such is not in the list of first names (see 3.12).

(3.9) ***Alain Walter**, adidas, geht noch tiefer in die Details bei den Schwierigkeiten, die ...*
*Die Konsequenz sieht für **Walter** so aus, daß ...*

(3.10) *"Im Juli werden die ersten Ergebnisse des San-Francisco-Projekts ausgeliefert", veranschaulicht **Julius Peter**.*
*... ergänzt Lawsons Cheftechnologe **Peter Patton**.*

(3.11) *Am Anfang war die Zukunftsvision von einem künftigen Operationssaals, die der Neurochirurg **Volker Urban** von der Dr.-Horst-Schmidt-Klinik ...*
*... räumt **Urban *Akzeptanzprobleme** ein.*

(3.12) *Als **Bruce Walter** seinen sofortigen Rücktritt von seinem Amt als Präsident der Grid Systems Corporation einreichte, ...*
*Bis ein Nachfolger nominiert ist, übernimmt der bisherige Vice President **Walters** Job.*

In an evaluation of 990 sentences from our computer magazine corpus we manually detected 116 person names. 73 of these names are full names and 43 are stand alone last names. Our algorithm achieves a recall of 93% for the full names (68 found) and of 74% for the stand alone names (32 found). The overall precision is 92%.

The algorithm relies on last names being introduced by first names or titles. It will miss a last name that occurs without this introduction. In our corpus this (rarely) happens for last names that are very prominent in the domain of discourse (*Gates*) and in cataphoric uses, mostly in headlines where the full name is given shortly after in the text.

(3.13) ⟨h2⟩***McNealy** präzisiert Vorwürfe gegen **Gates***⟨/h2⟩
*... Suns Präsident **Scott McNealy** hat auf der IT Expo ...*

This type of error could be tackled if we used all learned names not only for subsequent sentences but also for the immediately preceding headlines. And the problem with prominent names could be reduced by counting how often a name has been learned. If it is learned a certain number of times it stays in memory and will not be forgotten.

Recognition of geographical names

Names of geographical entities (cities, countries, states and provinces, mountains and rivers) are relatively stable over time. Therefore it is easy to compile such lists from resources in the WWW. In addition, we exploit the structure of our newspaper texts that are often introduced with a city name (cf. step 2). We collected all city names used in our computer magazine corpus as introductory words as well as (German) city names from the WWW into a gazetteer of around 1000 city names. We also use a list of 250 country names (including abbreviations like *USA*) and (mostly German) state names. When matching these geographical names in our corpus, we have to also include the genitive forms of these names (*Hamburgs, Deutschlands, Bad Sulzas*). Fortunately, the genitive is always formed with the suffix *-s*.

A more challenging aspect of geographical name recognition is their adjectival use, frequently as modifiers to company names or other organizations.

(3.14) *Das gleiche gilt für die zur **Londoner** Colt Telecom Group gehörende **Frankfurter** Colt Telecom GmbH, ...*

(3.15) *Die **amerikanische** Engineering Information und das **Karlsruher** Fachinformationszentrum wollen gemeinsam ...*

(3.16) *Die **japanische** Telefongesellschaft NTT drängt auf den internationalen Markt.*

We decided to also mark these adjectives as geographical names since they determine the location of the company or organization. The difficulty lies in building a gazetteer for these words. Obviously, it is difficult to find a gazetteer of derived forms in the WWW. But it is also difficult to derive these forms systematically from the base forms due to phonological deviations.

(3.17) *London → Londoner*

(3.18) *Karlsruhe → Karlsruher*

(3.19) *München → Münchner*

(3.20) *Bremen → Bremer*

(3.21) *England → englische/r*

(3.22) *Finnland → finnische/r*

As these examples show, both *-er* and *-isch* can be used as derivational suffixes to turn a geographical name into an adjective. *-isch* is the older form but it has been pushed back by *-er* since the 15th century (cf. [Fleischer and Barz 1995] p.240). While *-isch* is used to build a fully inflectional lower case adjective, *-er* is used to form an invariant adjective that keeps the capitalized spelling of the underlying noun. There is currently a strong tendency to use the *-isch* form for country names and the *-er* form for city names. Rarely, both forms are used side by side. A few country names have parallel capitalized adjective forms (*Luxemburger, Liechtensteiner, Schweizer*), and a few city names have parallel *-isch* forms (*münchnerische, römische*). If both forms exist, there is a slight but noticeable difference in usage. The *-isch* form describes a general trait of the region, whereas the *-er* form denotes an object as belonging to or being located in this place.

The lower case *-isch* adjective is available for almost any country name in the world (?*singapurisch* is one of the few debatable exceptions). Analogously, the *-er* form can be used for every city name in the German-speaking world and also for foreign city names unless they end in a vowel not used as suffix in German city names like *-i* (*Helsinki, Nairobi*) or *-o* (*Chicago, San Francisco*).

For all country names we manually compiled the list of the *-isch* base form of the adjectives. For the city names we are faced with a much larger set.

We therefore used the morphological analyzer Gertwol to identify such words. According to [Haapalainen and Majorin 1994], Gertwol comprises around 12,000 proper names out of which 2600 are geographical names. For every geographical name Gertwol derives a masculine and a feminine form for the inhabitants (*Bremer, Bremerin*) as well as the form for the adjective. The capitalized adjective form with suffix *-er* is available for all city names (*Bremer, Koblenzer*) and some state names (*Rheinland-Pfälzer, Saarländer, Thüringer*).

The capitalized geographical adjectives are therefore homographic to nouns denoting a masculine inhabitant of that city or state and also to the plural form of the inhabitants (*die*

Bremer sind ...). We use this ambiguity to identify geographical adjectives in the Gertwol output: If a capitalized word ending in *-er* is analyzed as both a proper name (the inhabitant reading) and an invariant adjective, then this word will be a geographical adjective and we can list it in a special gazetteer.

In our corpus we mark all forms of the lower case geographical adjectives. For the capitalized adjectives we mark all occurrences that are followed by a capitalized noun. Occurrences followed by a lower case word are likely to stand for the inhabitant reading (as in 3.23 and 3.24).

(3.23) *In Sachen Sicherheit kooperieren die* ***Düsseldorfer*** *mit ...*

(3.24) *Vor fünf Jahren hatten sich die* ***Redmonder*** *bei der Forschung ...*

In our evaluation of 990 sentences we manually found 173 geographical names. Out of these 159 were automatically marked (a recall of 91%). The algorithm incorrectly marked 28 geographical names (a precision of 85%). The precision is surprisingly low given the fact that the method works (mostly) with manually compiled lists. What then could be the reason for incorrectly annotated locations?

There are rare cases of ambiguities between geographical names and regular nouns (e.g. *Essen, Halle, Hof* are names of German cities as well as regular German nouns meaning *food, hall, yard*). There are also ambiguities between geographical names and person names (e.g. the first name *Hagen* is also the name of a German city). City names and geographical adjectives (e.g. *Schweizer, Deutsch*) can also be used as personal last names. But these ambiguities hardly ever occur in our corpus. Ambiguities also arise when a city name does not mark a location but rather stands for an organization (such as a government) and it happens that there are number of these in our 990 test sentences.

(3.25) *Die Argumentation ist losgelöst vom Aufbruch in die Informationsgesellschaft und unangreifbar für* ***Bonn*** *oder* ***Brüssel****.*

Recognition of company names

Company names are very frequent in our computer magazine corpus since most articles deal with news about hardware and software products and companies. Our algorithm for company name recognition is based on keywords that indicate the occurrence of a company name. Based on this, we have identified the following patterns:

1. A sequence of capitalized words after strong keywords such as *Firma*. The sequence can consist of only one such capitalized word and ends with the first lower case word. The keyword is not part of the company name.

   (3.26) *Nach einem Brandunfall bei der Firma* ***Sandoz*** *fließen bei Basel ...*

   (3.27) *... das Software-System "DynaText" der Firma* ***Electronic Book Technologies****.*

2. A sequence of capitalized words preceding keywords such as *GmbH, Ltd., Inc., Oy*. The sequence can consist of only one such capitalized word and ends to the left with the

first lower case word or with a geographical adjective or with a feminine determiner.[4] The keyword is considered to be part of the company name.

(3.28) *In Richtung Multimedia marschiert* ***J. D. Edwards & Co.*** *(JDE) mit ihrem kommerziellen Informationssystem ...*

(3.29) *... standen im Mittelpunkt der Hauptversammlung der Münchner* ***Siemens AG****.*

3. According to German orthographical standards, a compound consisting of a proper name and a regular noun is spelled with a hyphen. We exploit this fact and find company names in hyphenated compounds ending in a keyword such as *Chef, Tochter*.[5]

(3.30) *Der* ***Siemens****-Chef denkt offensichtlich an ...*

(3.31) *... ist die Zukunft der deutschen* ***France-Télécom****-Tochter geklärt.*

4. Combining evidence from two or more weaker sources suffices to identify candidates for company names. We have found two useful patterns involving geographical adjectives.

   (a) A sequence of capitalized words after a feminine determiner followed by a geographical adjective.

   (3.32) *In Deutschland ist das Gerät über die Bad Homburger* ***Ergos*** *zu beziehen.*

   (3.33) *Für Ethernet- und Token-Ring-Netze hat die Münchner* ***Ornetix*** *einen Medienserver entwickelt.*

   (3.34) *Mit Kabeln und Lautsprechern im Paket will die kalifornische* ***Media Vision*** *den PC- und Macintosh-Markt multimedial aufrüsten.*

   (b) A sequence of capitalized words after a geographical adjective and a weak keyword (like *Agentur, Unternehmen*).[6] Neither the adjective nor the keyword is part of the company name.

   (3.35) *Das Münchner Unternehmen* ***Stahlgruber*** *zählt zu den wenigen Anwendern, die ...*

Using these patterns our program "learns" simple and complex company names and saves them in a list. All learned company names constitute a gazetteer for a second pass of name application over the corpus. The learning of company names will thus profit from enlarging the corpus, while our recognition of person and geographical names is independent of corpus size.

Complex company names consist of two or more words. The complex names found with the above patterns are relatively reliable. Most problems arise with pattern 2 because it is difficult to find all possible front boundaries (cf. *das Automobilkonsortium Micro Compact Car AG*). Our algorithm sometimes includes unwanted front boundaries into the name.

Often acronyms refer to company names (*IBM* is probably the best known example). These acronyms are frequently introduced as part of a complex name. We therefore search

[4] In German, all company names are of feminine gender.

[5] We owe this observation to our student Jeannette Roth.

[6] We distinguish between strong keywords that always trigger company name recognition and weak keywords that are less reliable cues and therefore need to cooccur with a geographical adjective.

complex names for such acronyms (all upper case words) and add them to the list of found names.

(3.36) *... die **CCS Chipcard & Communications GmbH**. Tätigkeitsschwerpunkt der **CCS** sind programmierbare Chipkarten.*

Learning single-word company names is much more error prone. It can happen that a capitalized word following the keyword *Firma* is not a company name but a regular noun (*... weil die Firma Software verkauft*), or that the first part of a hyphenated compound with *Chef* is a country name (*Abschied von Deutschland-Chef Zimmer*). Therefore we need to filter these one-word company names before applying them to our corpus. We use Gertwol to analyse all one-word names. We accept as company names all words

- that are unknown to Gertwol (e.g. *Acotec, Belgacom*), or
- that are known to Gertwol as proper names (e.g. *Alcatel, Apple*), or
- that are recognized by Gertwol as abbreviations (e.g. *AMD, AT&T, Be*), and
- that are not in an English dictionary (with some exceptions like *Apple, Bull, Sharp, Sun*).

In this way we exclude all regular (lexical) nouns from the list of simple company names.

In a separate pass over the corpus we then apply all company names collected in the learning phase and cleared in the filter phase. In the application process we also accept genitive forms of the company names (*IBMs, Microsofts*).

Note that the order of name recognition combined with the rather cautious application of person names leads to the desired effect that a word can be both person name and company name in the same corpus. With the word *Dell* we get:

sentence	example	type
647	*Bei Gateway 2000 und **Dell** ...*	company
6917	*Auch IBM und **Dell** ...*	company
11991	*Michael **Dell***	person
11994	*... warnte **Dell***	person
12549	*Siemens Nixdorf, **Dell** und Amdahl ...*	company

In our evaluation of 990 sentences, the program found 283 out of 348 company name occurrences (a recall of 81%). It incorrectly recognized 89 items as company names that were not companies (a precision of 76%). These values are based on completely recognized names. Many company names, however, consist of more than one token. In our evaluation text 50 company names consist of two tokens, 13 of three tokens, 3 of four tokens and 1 of five tokens (*Challenger Gray & Christmas Corp.*). We therefore performed a second evaluation for company names checking only the correct recognition of the first token. We then get a recall of 86% and a precision of 80%.

With these patterns we look for sequences of capitalized words. That means we miss company names that are spelled all lower case (against conventions in German). We also have problems with names that contain function words such as conjunctions or prepositions. We will only partially match these names.

Investigating conjoined constructions seems like a worthwhile path for future improvements of our method. If we recognize a name within a conjoined phrase, we will likely find another name of the same type within that phrase. But since a conjunction can connect units of various levels (words, phrases, clauses), it is difficult to use them within a pattern matching approach.

(3.37) *... auf die Hilfe zahlreicher Kooperationspartner wie* ***BSP****,* ***Debis****,* ***DEC*** *oder* ***Telekurs*** *angewiesen ist.*

Recognition of product names

Proper names are defined as names for unique objects. A person name (e.g. *Bill Gates*) denotes a unique human being, a geographical name denotes a specific country, city, state or river. Although some cities are named alike (e.g. *Koblenz* is a city both in Germany and in Switzerland), a city name refers to one specific city according to the given context. Similarly, a company name refers to a specific commercial enterprise.

In this respect product names are different. When we use *Mercedes*, we might refer to a specific car, but we might also refer to the class of all cars that were produced under that name. Still, product names share many properties with person or company names. They are an open class with new names constantly being invented as new products are introduced into the market.

Product names are sometimes difficult to tell apart from company names (*Lotus, WordPerfect*). They also compete with names of programming languages (*C++, Java, Perl*), standards (*Edifact (Electronic Data Interchange for Adminstration, Commerce and Transport)*) and services (*Active-X-Technologie*). We experimented with restricting the name search to software and hardware products as exemplified in the following sentences.

(3.38) *Sie arbeiten fieberhaft an einem neuen gemeinsamen Betriebssystem namens* ***Taligent****, das ...*

(3.39) *Zur Optimierung des Datendurchsatzes unterstützt das aktuelle Release von* ***Netware*** *nun ...*

(3.40) *Die Multimedia-Ausstattung besteht aus einer Soundkarte (****Soundblaster Pro-II****) ...*

In a student project under our supervision [Roth 2001] investigated product name recognition over our corpus. She used the methods that we had explored for company name recognition. She first collected keywords that may trigger a product name in our domain (e.g. *System, Version, Release*). She identified specific patterns for these keywords (e.g. *Version* ⟨*number*⟩ *von* ⟨*product*⟩). The patterns were then used to collect product names from the corpus. Since this learned set of product names contained many words from the general vocabulary, they were filtered using the morphological analyzer Gertwol. As a novel move, Roth also used conjunction patterns to improve the recall.

```
PRODUCT (und|sowie|oder) PRODUCT
PRODUCT, PRODUCT (und|sowie|oder) PRODUCT
```

If one of the product names is learned based on the keyword patterns, then the other names in the conjunction patterns will be added to the list. If, in example 3.41, *Unix* has been learned as a product name, then *MCP* and *OS/2* will be added to the list.

(3.41) *Die A7-Openframes integrieren das proprietäre* ***MCP sowie Unix oder OS/2*** *auf Datei-, Programm- und Kommandoebene.*

Finally, all learned product names were applied to all matching strings in the corpus and marked as product names. An evaluation of 300 sentences showed that the precision in product name recognition was good (above 90%) but recall was very low (between 20% and 30%). Product names are so diverse that it is very difficult to find exact patterns to extract them. Due to the low recall we disregarded product names for the time being in our research.

3.1.3 Part-of-Speech Tagging

In order to extract nouns, verbs and prepositions we need to identify these words in the corpus. Before we decided on a part-of-speech (PoS) tagger, we performed a detailed comparative evaluation of the Brill-Tagger (a rule-based tagger) and the Tree-Tagger (a statistics-based tagger) for German. We showed that the Tree-Tagger was slightly better [Volk and Schneider 1998]. Therefore we use the Tree-Tagger [Schmid and Kempe 1996] in this research.

The Tree-Tagger uses the STTS (Stuttgart-Tübingen Tag Set; [Thielen and Schiller 1996]), a tag-set for German with around 50 tags for parts-of-speech and 3 tags for punctuation marks. The STTS distinguishes between proper nouns and regular nouns, between full verbs, modal verbs and auxiliary verbs, and between prepositions, contracted prepositions and postpositions.

The tagger works on the vertical text (each word and each punctuation mark in a separate line). In addition, in our corpus the tagger input already contains the proper name tag NE for all previously recognized names used as nouns (e.g. *München*) and the adjective tag ADJA for all recognized names in attributive use (e.g. *Münchner*). The tagger assigns one part-of-speech tag to every word in a sentence. It does not change any tag provided in the input text. Thus the prior recognition of proper names ensures the correct tags for these names and improves the overall tagging quality (cf. [Clematide and Volk 2001]).

After tagging, some missed sentence boundaries can be inserted. If, for instance, a number plus dot (suspected to be an ordinal number) is followed by a capitalized article or pronoun, there must be a sentence boundary after the number (*... freuen sich über die Werbekampagne für Windows 95. Sie steigert ihre Umsätze*). In our corpora we find between 75 and 130 such sentence boundaries per annual volume.

3.1.4 Lemmatization

In our experiments on PP attachment resolution we will use the word forms but also the base forms of verbs and nouns. We therefore decided to enrich our corpus with the base forms, also called lemmas, for all inflecting parts-of-speech. As usual, we reduced every noun to its nominative singular form, every verb to its infinitive form and every adjective to its uninflected stem form (*schönes, schönere* → *schön*).

We used the morphological analyser Gertwol [Lingsoft-Oy 1994] for this task. Gertwol is a purely word-based analyser that outputs every possible reading for a given wordform. For

instance, it will tell that *Junge* can be either an adjective (*young*) with lemma *jung* or a noun (*boy*) with lemma *Junge*. We thus have to compare Gertwol's output with the PoS tag to find the correct lemma.

All nouns, verbs and adjectives are extracted and compiled into a list of word-form tokens. With the UNIX `uniq` function we then turn the word-form token list into a word-form types list. The word-form types are analyzed and lemmatized by Gertwol.

Gertwol analyses a hyphenated compound only if it knows all components. This means that Gertwol will analyze *Software-Instituts* → *Software-Institut*, but it will not recognize *Informix-Aktien* since it does not know the word *Informix*. But the inflectional variation of such a compound word is only affected by the last component. Therefore we make Gertwol analyse the last component of each hyphenated compound so that we can construct the lemma even if one of the preceding components is unknown to Gertwol.

In addition, Gertwol is unable to analyse the upper case I-form of German nouns (e.g. *InformatikerInnen*). This form has become fashionable in German in the last decade to combine the male and female forms *Informatiker* and *Informatikerinnen*. We convert this special form into the female form so that Gertwol can analyse it. When merging the lemmas into the corpus, we convert it back to the upper case I resulting in the lemma *Inform-atik-er-In*.

When merging the Gertwol analysis into our corpus, we face the following cases with respect to the tagger output:

1. **The lemma was prespecified during name recognition.** In the recognition of proper names we included the genitive forms (cf. section 3.1.2). These are generated by adding the suffix *-s* to the learned name. Whenever we classify such a genitive name, we also annotate it with its base form.

word form	PoS tag	lemma	semantic tag
IBMs	NE	*IBM*	company
Kanthers	NE	*Kanther*	person

   These lines are not changed, the Gertwol information - if there is any - is not used.

   This increases the precision of the lemmatization step since many of the names are unknown to Gertwol. Instead of using the word form as lemma or simply chopping off any *-s* suffix, we can distinguish between names that end in *-s* in their base form (like *Paris*) and names that carry an inflectional suffix (*Schmitts, Hamburgs, IBMs*). In every annual volume of our corpus we identify around 2000 genitive names.

2. **Gertwol does not find a lemma.** Around 14% of all noun-form types in our corpus are unknown to Gertwol and therefore no lemma is found. Most of these are proper names and foreign language expressions. Moreover, around 7% of all verb-form types are unknown to Gertwol and no lemma is found. We insert the word form in place of the lemma into the corpus.

word form	PoS tag	lemma
corpus lines before lemmatization		
Cytosensor	NN	
Laboratories	NE	
corpus lines after lemmatization		
Cytosensor	NN	*Cytosensor*
Laboratories	NE	*Laboratories*

3. **Gertwol finds exactly one lemma for the given part-of-speech.** This is the desired case. The Gertwol lemma is added to the corpus.

word form	PoS tag	lemma
corpus line before lemmatization		
Technologien	NN	
Gertwol information		
Technologien	NN	*Techno|log-ie*
corpus line after lemmatization		
Technologien	NN	*Techno|log-ie*

4. **Gertwol finds multiple lemmas for the given part-of-speech.** 12% of the noun forms receive more than one lemma. The alternatives arise mostly from alternative segmentations because of dynamic undoing of compounding and derivation. We have developed a disambiguation method for these cases that relies on weighting the different segmentation boundaries [Volk 1999]. For instance, the word *Geldwäschereibestimmungen* will be analysed as both
Geld#wäsch-er#eib-e#stimm-ung and *Geld#wäsch-er-ei#be|stimm-ung*.

   It includes strong segmentation symbols (#) that mark the boundary between elements that can occur by themselves (independent morphemes). It also includes a weak segmentation symbol (|) that is used for prefixes and dependent elements. The dash indicates the boundary in front of a derivational or inflectional morpheme. By counting and weighting the segmentation symbols we determine that the latter segmentation of our example word has less internal complexity and is thus the correct lemma. This method leads to the correct lemma in around 90% of the ambiguous cases.

word form	PoS tag	lemma
corpus line before lemmatization		
Geldwäschereibestimmungen	NN	
Gertwol information		
Geldwäschereibestimmungen	NN	*Geld#wäsch-er#eib-e#stimm-ung*
Geldwäschereibestimmungen	NN	*Geld#wäsch-er-ei#be|stimm-ung*
corpus line after lemmatization		
Geldwäschereibestimmungen	NN	*Geld#wäsch-er-ei#be|stimm-ung*

   6% of the verbs receive more than one lemma. The alternatives arise mostly from different segmentations while dynamically undoing prefixation and derivation. We compute the best verb lemma with a method analogous to the noun segmentation disambiguator.

5. **Gertwol finds a lemma but not for the given part-of-speech.** This indicates that there is a tagger error, and we use the Gertwol analysis to correct these.

   (a) If a word form is tagged with PoS tag X, but Gertwol states that only PoS tag Y is possible, we substitute X with Y in our corpus and also add the corresponding lemma. This amounts to giving preference to Gertwol's judgement over the tagger's judgement. This is based on the observation that Gertwol's precision is very high.[7]

   (b) If a word form is tagged with PoS tag X, but Gertwol has more than one tag (excluding X), we have to decide on the best tag. We follow the tagger tag as closely as possible. This means we will try first to exchange ADJA with ADJD (attributive with predicative adjective form), NN with NE (regular noun with proper noun), and any verb form tag with another verb form tag. If such a matching tag within the word class is not available, our algorithm guesses and takes the first lemma offered by Gertwol.

   According to these rules, we substituted 0.74% of all the PoS tags (or 2% of the adjective, noun, verb tags). In absolute figures this means that in an annual volume of our corpus we exchanged around 14,000 tags. 85% of the exchanges are cases with exactly one Gertwol tag and only 15% are cases which the system had to guess.

word form	PoS tag	lemma
corpus lines before lemmatization		
Software	NE	
Festplatte	VVFIN	
Gertwol information		
Software	NN	*Soft|ware*
Festplatte	NN	*Fest#platt-e*
Festplatte	ADJA	*fest#platt*
corpus lines after lemmatization		
Software	NN	*Soft|ware*
Festplatte	NN	*Fest#platt-e*

We also computed the lemma for contracted prepositions (e.g. *am* → *an, ins* → *in, zur* → *zu*). Right-truncated compounds were not lemmatized. It would be desirable to lemmatize them with their full form (*Text- und Lernprogramme* → *Text#programm und Lern#programm*) since the rightmost component determines the meaning. Left-truncated compounds were lemmatized in their reduced form (*Softwarehäuser oder -abteilungen* → *Soft|-ware#haus oder -Ab|teil-ung*). All other word classes do not inflect or need not be lemmatized for our purposes (e.g. possessive or demonstrative pronouns).

[7] As a consequence, the order of application of the PoS tagger and Gertwol could be reversed, i.e. we could use Gertwol first and provide PoS tags for all words that have only one unique Gertwol tag. The tagger would then fill in only the tags for the ambiguous words. We expect that this method would improve the tagger output, but we have not yet evaluated this method.

3.1.5 Chunk Parsing for NPs and PPs

We use a pattern matcher with part-of-speech patterns to identify the most common noun phrases and prepositional phrases. These include adjective phrases as well as conjoined noun, prepositional and adjectival phrases (2 levels deep). Here are some example patterns with PoS tags from the STTS.

```
#### Adjective Phrases
# example:  sehr gross
ADV ADJA --> AP

# example:  zu gross
PTKA ADJA --> AP

#### Prepositional Phrases
# example:  auf einem hohen Level
APPR ART ADJA NN --> PP

# example:  mit den [50 erfolgreichsten] Firmen
APPR ART AP NN --> PP

# example:  vor den [technischen und politischen] Gefahren
APPR ART CAP NN --> PP
```

Similar chunk parsers for German have been described by [Skut and Brants 1998] using a statistical model (Viterbi search on the basis of trigram frequencies and a maximum-entropy technique) and [Piskorski and Neumann 2000] using weighted finite state transducers. Also similar are the corpus annotation tools described by [Kermes and Evert 2001] making use of Perl scripts and queries to a Corpus Query Processor within the University of Stuttgart's Corpus Workbench. A comparison and evaluation of the performace of these systems has never been undertaken.

The phrase information is stored in the NEGRA export format [Skut et al. 1997]. This is a line-based format using numerical identifiers for nested phrases. The NEGRA annotation format tries to keep structures as flat as possible without losing information. Towards this goal, NEGRA does not ask for an explicit NP node within a PP, since all words after the preposition always constitute the NP. Only if a subconstituent has an internal structure, such as a complex adjective phrase or conjoined nouns, is it marked with special nodes.

The following listing shows an example sentence in the NEGRA format after name recognition, lemmatization and NP/PP recognition. Figure 3.2 shows the first part of the sentence as partial trees.

```
Laut                APPR    --      AC      505     %% laut
Einschätzung        NN      --      NK      505     %% Ein|schätz~ung
von                 APPR    --      AC      504     %% von
Lutz                NE      --      PNC     500     %% Lutz <PERS1>
Meyer-Scheel        NE      --      PNC     500     %% Meyer-Scheel <PERS1>
,                   $,      --      --      0
```

```
Vorstandsvorsitzender   NN      --      --      0       %% Vor|stand\s#vor|sitzend
der                     ART     --      NK      506
Hamburger               ADJA    --      NK      506     %% Hamburg~er <GEO1>
Info                    NE      --      PNC     501     %% Info <FA1>
AG                      NE      --      PNC     501     %% AG <FA1>
,                       $,      --      --      0
werden                  VAFIN   --      --      0       %% werd~en
nach                    APPR    --      AC      503     %% nach
einer                   ART     --      NK      503
längeren                ADJA    --      NK      503     %% lang
Umstrukturierung        NN      --      NK      503     %% Um|struktur~ier~ung
künftig                 ADJD    --      --      0       %% künftig
wieder                  ADV     --      --      0
positive                ADJA    --      NK      502     %% posit~iv
Ergebnisse              NN      --      NK      502     %% Er|geb~nis
erzielt                 VVPP    --      --      0       %% er|ziel~en
.                       $.      --      --      0

#500                    MPN     --      NK      504
#501                    MPN     --      NK      506
#502                    NP      --      --      0
#503                    PP      --      --      0
#504                    PP      --      --      0
#505                    PP      --      --      0
#506                    NP      --      --      0
```

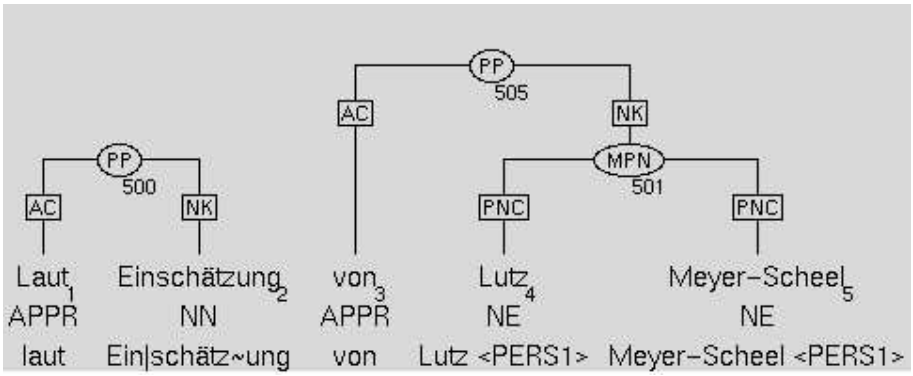


Figure 3.2: Automatically computed phrasal trees (PPs and MPN) with lemmas and proper name tags

The information in the NEGRA format is divided into two blocks. The first block holds the words and the corresponding information, the second block holds the phrase nodes. Within the first block, column 1 contains the word forms and punctuation symbols. Column 2 contains the part-of-speech tags. Column 3 is reserved for morphological information which we do not use here. Column 4 contains the function of the word within its immediately dominating node (the function symbols are documented in [Negra-Group 2000]). Column 5

holds the numerical pointers to those nodes which are spelled out in the second block. The last column (6) may hold a word comment. We use this last column for the lemma and for our semantic information on person (PERS), geographical (GEO) or company names (FA).

All constituent nodes are listed in the second block. In this example our chunk parser recognizes three PPs (*laut Einschätzung, von Lutz Meyer-Scheel, nach einer längeren Umstrukturierung*), two multiword proper nouns (MPN; *Lutz Meyer-Scheel* and *Info AG*), and two NPs (*der Hamburger Info AG, positive Ergebnisse*). The two MPNs are integrated in second level constituents. The parser does not attempt any attachments. Neither genitive NPs nor PPs are attached to a possible landing site.

We will tackle the recognition and attachment of genitive NPs in a next step. Information about grammatical case of determiners, adjectives and nouns can be obtained from Gertwol. This will be used to find the grammatical case of phrases. Genitive NPs can be attached to the preceding noun with a high certainty. Genitive NPs functioning as verbal objects are very rare, and thus ambiguities involving genitive NPs occur seldom. We also need to consider pre-nominal genitive attributes. They mostly consist of names, and we will thus profit from our proper name recognition. Examples 3.42 and 3.43 show company names as pre-nominal genitive attributes in an NP and a PP. Sentence 3.44 is an example of a genitive name that could be both a post-nominal attribute to *Technik-Manager* and a pre-nominal attribute to *Wettbewerbsfähigkeit*.

(3.42) ***IBMs*** *jüngst angekündigte RISC-Unix-Rechner der RS/6000-Linie standen auf dem Prüfstand.*

(3.43) *Mit Gelassenheit reagiert Sunsoft auf* ***Microsofts*** *neues 32-Bit-Betriebssystem.*

(3.44) *In einer Umfrage ermittelte der Verband Deutscher Elektrotechniker (VDE), wie die Technik-Manager* ***Deutschlands*** *Wettbewerbsfähigkeit bewerten.*

Still, this type of phrase recognition helps us in subsequent steps, in delimiting temporal and local PPs as well as in determining sure PP attachments (cf. 4.5).

3.1.6 Recognition of Temporal and Local PPs

Prepositional phrases fall into various semantic classes. [Drosdowski 1995] makes a rough distinction into modal, causal, temporal and local PPs. Out of these, temporal and local are easiest to classify since they denote clear concepts of point and duration of time as well as direction and position in space.

We use lists of prepositions and typical temporal and local nouns and adverbs to identify such PPs.[8] The prepositions are subdivided into

- 3 prepositions that always introduce a temporal PP: *binnen, während, zeit.*
- 30 prepositions that may introduce a temporal PP: e.g. *ab, an, auf, bis.*
- 21 prepositions that always introduce a local PP: e.g. *fern, oberhalb, südlich von.*
- 22 prepositions that may introduce a local PP: e.g. *ab, auf, bei.*

[8]The lists for the recognition of temporal PPs were compiled in a student project by Stefan Höfler.

Note that contracted prepositions like *am, ans, zur* are mapped to their base prepositions during lemmatization so that they need not be listed here.

If a preposition always introduces a temporal or local PP, the type of the preposition is a sufficient indication for the semantic classification of the PP. On the other hand, if the preposition only sometimes introduces a temporal or local PP, we require additional evidence from the core of the PP. If the core consists of a typical adverb or a typical noun, then the PP is classified.

We list 230 typical temporal adverbs like *heute, niemals, wann.* We did not make a distinction between adverbs that can occur within a PP and adverbs that can only occur standing alone. We also list 17 typical local adverbs like *dort, hier, oben, rechts.*[9]

In addition we have compiled lists of typical nouns. Examples of typical temporal nouns are names of months and weekdays, time spans (*Minute, Stunde, Tag, Woche, Monat, Jahr, Jahrhundert*), and others like *Anfang, Zeitraum, Zukunft.*

Typical local nouns are not easy to collect. We started with the city environment (*Strasse, Quartier, Stadt, Land*) and with directions (*Norden, Osten, Südosten*). But many physical location words can also be used to denote organizations (*Bank, Universität*) and make it difficult to classify them as locations. To be on the safe side, we used the previously recognized geographical entities as core to a local PP.

All temporal and local information is annotated as word comment in the NEGRA format. If preposition and core of a PP are evidence for a temporal or local PP, the complete PP (including attributes) is marked with this semantic type.

(3.45) *Angestrebt wird der Verkauf von 10,000 Geräten* ***im ersten Jahr****.*

(3.46) *... läßt sich der Traktor wahlweise im Schub- oder Zugmodus betreiben, das Papier* ***von hinten****, oben und auch* ***von unten*** *zuführen.*

In an evaluation of 990 sentences from our corpus, we found 263 temporal and 131 local PPs. The following table shows the results. We evaluated twice, checking once only the correct start token of the PP and once the correct recognition of all phrase tokens.

	in corpus	found	correct	incorrect	precision	recall
local PPs start	131	62	51	11	82%	39%
local PPs tokens	360	159	127	32	80%	35%
temporal PPs start	263	246	200	46	81%	76%
temporal PPs tokens	547	340	311	29	91%	57%

The table shows that our module for the recognition of temporal and local PPs works with a high precision but has a much lower recall especially for the local PPs (35%). Local PPs are harder to identify than temporal PPs since there is a wider spectrum of lexical material to denote a position or a direction in space compared to temporal expressions.

The annotated corpus is used as the basis for both the computation of the N+P cooccurrences and the V+P cooccurrences. We will look at these computations in chapter 4.

[9]Note that we have to consider orthographic variations of these adverbs such as *vorn, vorne; außen, aussen.*

3.1.7 Clause Boundary Recognition

A sentence consists of one or more clauses, and a clause consists of one or more phrases (i.e. noun phrases, prepositional phrases, adverb phrases und the like) [Greenbaum 1996]. A clause is a unit consisting of a full verb together with its (non-clausal) complements and adjuncts (as well as the auxiliary verbs in the verb group). An auxiliary verb or a modal verb can sometimes function as full verb if no 'regular' full verb is present. The copula verb *sein* (as in sentence 3.47) and the verb *haben* in the sense of *to possess, to own* are examples of this. Clauses constitute the unit in which a verb and an attached prepositional phrase cooccur.

(3.47) *ICL ist nun die größte Fachhandelsorganisation mit Headquarter in Großbritannien.*

(3.48) *Heute können Daten automatisch in gemeinsame MIS-Datenbasen überführt und verarbeitet werden.*

Usually a clause contains exactly one full verb. Exceptions are clauses that contain coordinated verbs. Usually this results in a complex sharing of the complements (as in 3.48). Other exceptions are clauses with a combination of a perception verb and an infinitive verb in so-called accusative with infinitive (AcI) constructions (as in the second clause of 3.49; the tag $\langle CB \rangle$ marks the clause boundary). These constructions are even more frequent with the verb *lassen* (example 3.50). Although these sentences look like active sentences (there is no passive verb form), they often express an impersonal point of view with regard to the main verb. The accusative object of *lassen* is the logical subject of the dependent verb. Reflexive usage of *lassen* is frequently used in impersonal expressions (example 3.51) with a clear passive sense.

(3.49) *Wir halten uns strikt an die Analysten, $\langle CB \rangle$ die den Markt in den nächsten drei Jahren um je 40 Prozent wachsen sehen.*

(3.50) *Der US-Flugzeughersteller Boeing läßt die technischen Handbücher sämtlicher Flugzeugmodelle auf CD-ROM übertragen.*

(3.51) *Die geforderten elektrischen Eigenschaften lassen sich chemisch durch den Einbau elektronenab- oder aufnehmender Seitenketten erzeugen.*

Clauses can be coordinated (forming a compound sentence, as in 3.52) or subordinated (resulting in a complex sentence). Subordinate clauses may contain a finite verb (as in 3.53) or a non-finite verb (as in 3.54). Subordination is signalled by a subordinator (a complementizer or relative pronoun). Clauses can be elliptical (lacking some complement, or even the verb itself). This often happens in compound sentences. Clauses with inserted clauses (marked off by hyphens as in 3.55 or parentheses) can also be seen as complex nested clauses.

(3.52) *Immer mehr Firmen und Behörden verlieren ihre Berührungsängste $\langle CB \rangle$ und greifen auf Shareware zurück.*

(3.53) *Analysten rechnen jedoch nicht damit, $\langle CB \rangle$ daß die Minderheitseigner Novell und AT&T noch einen Strich durch die Rechnung machen.*

(3.54) *Noorda bemüht sich schon seit längerem, $\langle CB \rangle$ sein Imperium zu erweitern.*

(3.55) *Leichte Startschwierigkeiten des Programmes ⟨CB⟩ - der Laserdrucker machte Probleme - ⟨CB⟩ behob der Autor innerhalb weniger Tage.*

Since verb and preposition cooccur within a clause, the sentences of our corpus need to be split up into clauses. We use a clause boundary detector that was developed in this project.[10] It consists of patterns over part-of-speech tags, most of which state some condition in connection with a comma. Currently the clause boundary detector consists of 34 patterns. If, for example, a comma is followed by a relative pronoun, there is a clause boundary between them. Or if a finite verb is followed by some other words, a conjunction, and another finite verb, then there is a clause boundary in front of the conjunction. Most difficult are those clauses that are not introduced by any overt punctuation symbol or word (as in 3.56).

(3.56) *Simple Budgetreduzierungen in der IT in den Vordergrund zu stellen ⟨CB⟩ ist der falsche Ansatz.*

The goal of clause boundary detection is to identify as many one-verb clauses as possible. Our clause boundary detector focuses on recall rather than precision. It tries to find as many clause boundaries as possible. It leaves relatively few clauses with more than one verb, but it results in many clauses without a full verb (copula sentences, article headers and clause fragments). In the CZ corpus we find:

Number of clauses with a single full verb	406,091
Number of clauses with multiple full verbs	23,407
Number of clauses without a full verb	182,000

We evaluated our clause boundary detector over 1150 sentences.[11] We manually determined all clause boundaries in these sentences. They contained 754 intra-sentential boundaries adding up to a total of 1904 clause chunks.

The clause boundary detector splits these test sentences into 1676 clause chunks including 70 false boundaries. This translates into a recall of 84.9% and a precision of 95.8%. These figures include the clause boundaries at the end of each sentence which are trivial to recognize. If we concentrate on the 754 intra-sentential clause boundaries, we observe a recall of 62.1% and a precision of 90.5%. We deliberately focused on high precision (few false clause boundaries) since we can easily identify clauses with missed clause boundaries based on multiple full verbs.

Using a PoS tagger as clause boundary detector

The clause boundary detector can be seen as a disambiguator between clause-combining tokens (mostly commas but also other punctuation symbols or conjunctions) and tokens (commas etc.) that combine smaller units (such as phrases or words). This disambiguation task is similar to the task faced by a part-of-speech (PoS) tagger for tokens belonging to two or more parts-of-speech. We therefore tested two PoS taggers as clause boundary detectors.[12]

[10] Our approach to clause boundary recognition resembles the approach described in [Ejerhed 1996].

[11] The CB detector was originally developed by the author. It was enhanced and evaluated by our student Gaudenz Lügstenmann.

[12] These experiments were for the most part organized and evaluated by my colleague Simon Clematide.

We used 75% of our manually annotated set of clauses as training corpus for the taggers. In the training corpus all clause triggering tokens were annotated as either clause boundary tokens or with their usual part-of-speech tag. All other words had been automatically tagged. Both taggers were then applied to tagging the remaining 25% of the clause set. Using 3 rounds of cross-validation, we determined 91% recall and 93% precision for the Brill tagger, and 89% recall with 89% precision for the Tree-Tagger (in both cases including sentence-final clause boundaries). If we focus solely on comma disambiguation, we get 75% recall and precision values. This means that three quarters of the commas were assigned the correct tag.

These results on using a PoS tagger for clause boundary recognition need reconfirmation from a larger training and test corpus. In particular, one needs to modify the tagger to insert clause boundaries in between words, which is a non-trivial modification.

Clause boundary recognition vs. clause recognition

Clause boundary detection is not identical to clause detection. In clause boundary detection we will only determine the boundaries between clauses, but we do not identify discontinuous parts of the same clause. The latter is much more difficult, and due to the nesting of clauses it should be done with a recursive parsing approach rather than with a pattern matcher. Example sentence 3.57 contains a relative clause nested within a matrix clause. The clause boundary detector finds the boundaries at the beginning and end of the relative clause. A clause detector will have to indicate that the matrix clause continues after the relative clause. It will therefore have to mark the beginning and end of each clause (as sketched in 3.58).

(3.57) *Nur ein Projekt der Volkswagen AG, ⟨CB⟩ die ihre europäischen Vertragswerkstätten per Satellit vernetzen will, ⟨CB⟩ stößt in ähnliche Dimensionen vor.*

(3.58) *⟨C⟩ Nur ein Projekt der Volkswagen AG, ⟨C⟩ die ihre europäischen Vertragswerkstätten per Satellit vernetzen will, ⟨/C⟩ stößt in ähnliche Dimensionen vor. ⟨/C⟩*

3.2 Preparation of the Test Sets

3.2.1 Extraction from the NEGRA Treebank

In 1999 the NEGRA treebank [Skut et al. 1998] was made available. It contains 10,000 manually annotated sentences for German (newspaper texts from the Frankfurter Rundschau). In this treebank, every PP is annotated with one of the following functions:

- 'postnominal modifier' or 'pseudo-genitive' (a *von*-PP used instead of an adnominal genitive; see example 3.59 as a variant of 3.60). We count these as noun attachments.
- 'modifier' (of a verb) or 'passivised subject' (a *von*-PP expressing the logical subject in a passive clause; see example 3.61 and the active mood variant in 3.62). We count these as verb attachments.
- seldom: some other function such as 'comparative complement' or 'measure argument of adjective'. We disregard these functions.

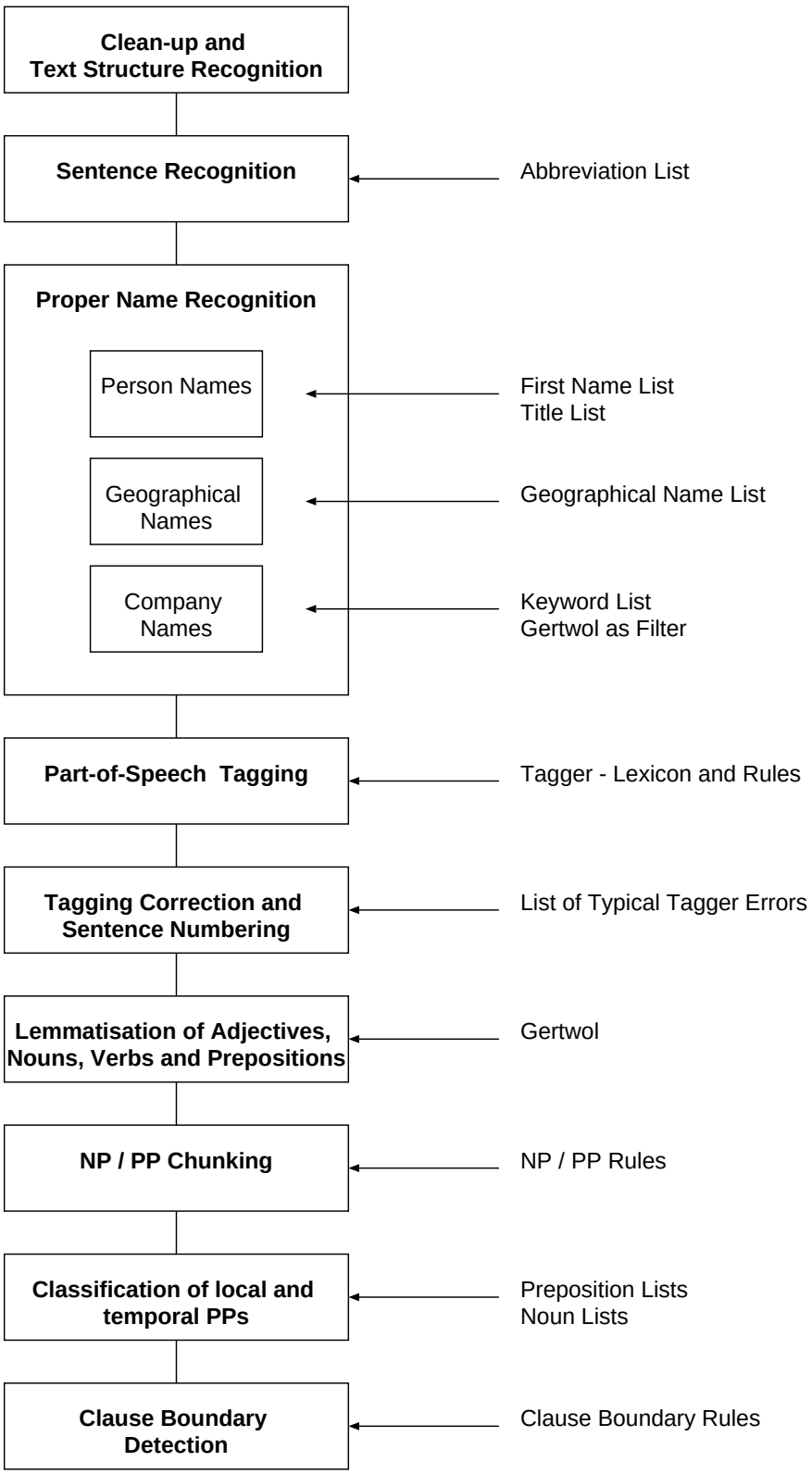


Figure 3.3: Overview of corpus preparation

(3.59) *Borland hat nach dem Rücktritt* ***von Gary Wetsel*** *einen neuen CEO gefunden.*

(3.60) *Borland hat nach* ***Gary Wetsels*** *Rücktritt einen neuen CEO gefunden.*

(3.61) *Dummerweise wird diese Einschätzung* ***von vielen innovativen kleinen Unternehmen*** *aus Nordamerika bestätigt.*

(3.62) *Dummerweise bestätigen* ***viele innovative kleine Unternehmen*** *aus Nordamerika diese Einschätzung.*

No distinction is made between complements and adjuncts.

We converted the sentences line by line from NEGRA's export format (cf. section 3.1.5) into a Prolog format. This format consists of `line/6` and `p_line/5` predicates. The arguments in a `line/6` predicate are sentence number, word number, word, part-of-speech, function and pointer to a phrasal node. The phrasal node lines contain sentence number, node number, phrase name, phrase function and a pointer to the superordinate node. This Prolog format is used to convert the line-based format into a nested structure so that it becomes feasible to access and extract the necessary information for PP attachment. Prolog was chosen for this task since it is well suited to work with nested sentence structures. Example for a sentence in the Prolog line format:

```
line(7561,  1, 'Das',         'ART',      'NK',   500).
line(7561,  2, 'Dorfmuseum',  'NN',       'NK',   500).
line(7561,  3, 'gewährt',     'VVFIN',    'HD',   505).
line(7561,  4, 'nicht',       'PTKNEG',   'NG',   504).
line(7561,  5, 'nur',         'ADV',      'MO',   504).
line(7561,  6, 'einen',       'ART',      'NK',   504).
line(7561,  7, 'Einblick',    'NN',       'NK',   504).
line(7561,  8, 'in',          'APPR',     'AC',   503).
line(7561,  9, 'den',         'ART',      'NK',   503).
line(7561, 10, 'häuslichen',  'ADJA',     'NK',   503).
line(7561, 11, 'Alltag',      'NN',       'NK',   503).
line(7561, 12, 'vom',         'APPRART',  'AC',   501).
line(7561, 13, 'Herd',        'NN',       'NK',   501).
line(7561, 14, 'bis',         'APPR',     'AC',   502).
line(7561, 15, 'zum',         'APPRART',  'AC',   502).
line(7561, 16, 'gemachten',   'ADJA',     'NK',   502).
line(7561, 17, 'Bett',        'NN',       'NK',   502).
line(7561, 18, '.',           '$.',       '--',     0).
p_line(7561, 500, 'NP',  'SB',  505).
p_line(7561, 501, 'PP',  'MNR', 503).
p_line(7561, 502, 'PP',  'MNR', 503).
p_line(7561, 503, 'PP',  'MNR', 504).
p_line(7561, 504, 'NP',  'OA',  505).
p_line(7561, 505, 'S',   '--',    0).
```

We used a Prolog program to build the nested structure and to recursively work through the annotations in order to obtain sixtuples with the relevant information for the PP classification task. The sixtuples include the following elements:

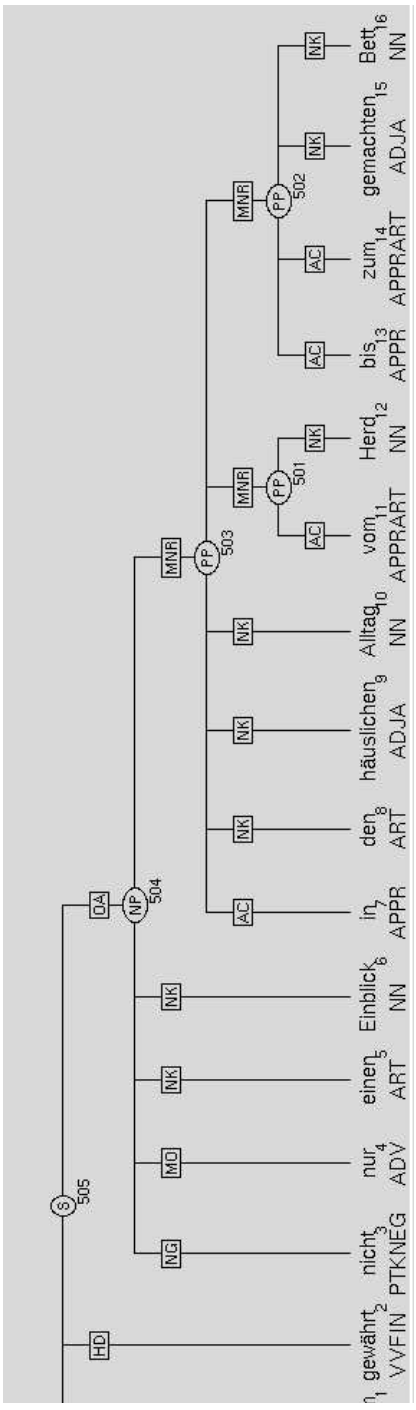


Figure 3.4: Tree from ANNOTATE tool

1. the full verb (with reflexive pronoun if there is one),
2. the real head noun (the noun which the PP is attached to),
3. the possible head noun (the noun that immediately precedes the preposition; this noun leads to the attachment ambiguity),
4. the preposition or pronominal adverb,
5. the core of the PP (noun, number, adjective, or adverb), and
6. the attachment decision (as given by the human annotators).

Let us illustrate this with some examples.

(3.63) *Das Dorfmuseum gewährt nicht nur einen Einblick* ***in den häuslichen Alltag vom Herd bis zum gemachten Bett****.*

(3.64) *... nachdem dieses wichtige Feld* ***seit 1985*** *brachlag.*

(3.65) *Das trifft auf alle Waren* ***mit dem berüchtigten "Grünen Punkt"*** *zu.*

(3.66) *Die Übereinkunft sieht die Vermarktung des Universal-Servers* ***von Informix auf den künftigen NT-Maschinen*** *vor.*

These corpus sentences will lead to the following sixtuples:

verb	real head N	possible head N	prep.	core of PP	PP function
gewährt	*Einblick*	*Einblick*	*in*	*Alltag*	noun modifier
gewährt	*Alltag*	*Alltag*	*vom*	*Herd*	noun modifier
gewährt	*Alltag*	*Herd*	*bis*	*Bett*	noun modifier
brachlag	/	*Feld*	*seit*	*1985*	verb modifier
zutrifft	*Waren*	*Waren*	*mit*	*Punkt*	noun modifier
vorsieht	*Servers*	*Servers*	*von*	*Informix*	noun modifier
vorsieht	*Vermarktung*	*Informix*	*auf*	*Maschinen*	noun modifier

Each sixtuple represents a PP with the preposition occuring in a position where it can be attached either to the noun or to the verb. Note that the PP *auf alle Waren* in 3.65 is not in such an ambiguous position and thus does not appear in the sixtuples.

In the example sentence 3.63 and 3.66 we observe the difference between the real head noun and the possible head noun. The PP *bis zum gemachten Bett* is not attached to the possible head noun *Herd* but to the preceding noun *Alltag*. In example 3.66 the PP *auf den künftigen NT-Maschinen* is not attached to the possible head noun *Informix* but to the preceding noun *Vermarktung*. Obviously, there is no real head noun if the PP attaches to the verb (as in 3.64).

We get multiple tuples from one sentence if there is more than one noun-preposition sequence with the same verb or with different verbs. We also get multiple tuples if the PP contains a coordination. The overall goal of the sixtuple extraction is to get as many test cases as possible from the manually annotated material. Therefore we do include sixtuples that are derived from PPs that form part of a sentence-initial constituent in a verb-second

clause (as in 3.67). A PP occurring in this position cannot be attached to the verb. But since this sentence could always be reordered into 3.68 due to the variable constituent ordering in German, we include this PP as a possibly ambiguous case in our test set.

(3.67) ***Die Nachfrage nach japanischen Speicherchips*** *dürfte im zweiten Halbjahr 1993 deutlich ansteigen.*

(3.68) *Im zweiten Halbjahr 1993 dürfte* ***die Nachfrage nach japanischen Speicherchips*** *deutlich ansteigen.*

There are a number of special cases that need to be considered:

Discontinuous elements

1. **Separable prefix verbs and reflexive pronouns:** If a separated prefix occurs, it is reattached to the verb (occuring in the same clause). If a reflexive pronoun occurs, it is also marked with the verb with the exception of reflexive pronouns in verb clauses that are dependent on *lassen* (as in 3.70). Those clauses are impersonal passive constructions and do not indicate a reflexivity of the main verb (cf. [Zifonun et al. 1997] p. 1854).

   (3.69) *Der Sozialistenchef und Revolutionsveteran Hocine Ait Ahmed setzte sich aus Sorge* ***um seine persönliche Sicherheit nach dem Mord an Boudjaf*** *erneut ins Ausland ab.*

   (3.70) *Ihre Speicherkapazität* ***lässt sich*** *von 150 Gigabyte* ***auf über 10 Terabyte ausbauen****.*

verb	real head N	possible head N	prep.	core of PP	PP function
sich absetzte	*Sorge*	*Sorge*	*um*	*Sicherheit*	noun modifier
sich absetzte	/	*Sicherheit*	*nach*	*Mord*	verb modifier
sich absetzte	*Mord*	*Mord*	*an*	*Boudjaf*	noun modifier
ausbauen	/	*Gigabyte*	*auf*	*Terabyte*	verb modifier

2. **Postposition or circumposition:** Postpositional PPs are omitted. But the right element of a circumposition is extracted with the preposition to form a complex entry in the preposition field. The NEGRA treebank contains 52 postposition tokens and 63 circumposition tokens.

   (3.71) *Er leitete seinen Kammerchor der Oberstufe* ***vom Klavier aus****, ...*

verb	real head N	possible head N	prep.	core of PP	PP function
leitete	/	*Oberstufe*	*vom aus*	*Klavier*	verb modifier

3. **Multiword proper noun:** Proper nouns consisting of more than one token are combined into one orthographic unit (with blanks substituted by underscores) so that the complete name is available. All proper nouns (multiword names and simple names) are specially marked so that we can distinguish them from regular nouns if need arises. The NEGRA corpus does not contain any semantic classification for proper nouns.

(3.72) *Als Resümee ihrer Untersuchungen warnten die Mediziner* ***um Gerhard Jorch*** *dringend davor ...*

verb	real head N	possible head N	prep.	core of PP	PP function
warnten	*Mediziner*	*Mediziner*	*um*	*Gerhard Jorch*	noun modifier

Coordinated elements

1. **Coordinated NPs or PPs:** If the PP is coordinated or if the core of the PP consists of a coordinated NP, we derive as many sixtuples as there are nouns in the coordination. On the other hand, right-truncated compounds are omitted since their most important component is missing.

   (3.73) *Sie bringen behinderte Menschen* ***zur Schule, zur Arbeit, zu privaten oder kulturellen Terminen****.*

   (3.74) *Weitere 200 Millionen würden durch Einzelmaßnahmen* ***bei der Gehalts- und Arbeitszeitstruktur*** *gespart.*

verb	real head N	possible head N	prep.	core of PP	PP function
bringen	/	*Menschen*	*zur*	*Schule*	verb modifier
bringen	/	*Menschen*	*zur*	*Arbeit*	verb modifier
bringen	/	*Menschen*	*zu*	*Terminen*	verb modifier
gespart	*Einzelmaßn.*	*Einzelmaßnahmen*	*bei*	*Arbeitszeitstruktur*	noun modifier

2. **Coordinated full verbs:** If two or more full verbs are coordinated or if they occur in coordinated verb phrases, we combine these verbs with all PPs.

   (3.75) *Das Bernoulli-Laufwerk "Multidisk 150"* ***liest und beschreibt*** *magnetische Wechselplatten mit einer Kapazität von 30 bis maximal 150 MB.*

verb	real head N	possible head N	prep.	core of PP	PP function
liest	*Wechselplatten*	*Wechselplatten*	*mit*	*Kapazität*	noun modifier
beschreibt	*Wechselplatten*	*Wechselplatten*	*mit*	*Kapazität*	noun modifier

3. **Coordinated prepositions and double preposition PPs:** PPs with coordinated prepositions lead to as many sixtuples as there are prepositions in the coordination. On the contrary, in double preposition PPs (like in 3.63) only the first preposition is extracted, since this preposition determines the character of the PP. This is obviously true for genitive substitution PPs as in *jenseits von Afrika*, but it also holds for *bis*-PPs.

4. **Elliptical clause without full verb:** The NEGRA annotation scheme does not annotate grammatical traces. An elliptical clause without an overt verb may nevertheless be annotated as a full sentence. These clauses are discarded during extraction.

   (3.76) *Platz 2 der Umsatzrangliste belegte Cap Gemini Sogetti mit rund 1,5 Milliarden,* ***Platz 3 Siemens Nixdorf mit 1,2 Milliarden Mark****.*

verb	real head N	possible head N	prep.	core of PP	PP function
belegte	/	*Cap Gemini Sogetti*	*mit*	*Milliarden*	verb modifier

5. **Duplicates:** Exact sixtuple duplicates are suppressed. Sentence 3.77 will give rise to the same sixtuple twice. The second item is suppressed in order not to bias the test set.

   (3.77) *... welches am 14. Juni* ***um 11 Uhr*** *und am 15. Juni* ***um 20 Uhr*** *im Großen Haus stattfindet.*

verb	real head N	possible head N	prep.	core of PP	PP function
stattfindet	*Juni*	*Juni*	*um*	*Uhr*	noun modifier

Additional elements in the PP

1. **Pre-prepositional modifier:** Sometimes a PP contains a modifier in front of the preposition. Most of these are adverbs or the negation particle *nicht*. These modifiers are disregarded during extraction. Such a modifier occurs in 809 out of 16,734 PPs (5%) in the NEGRA treebank.

   (3.78) *... wobei sich das Kunstwerk* ***schon mit seinem Entwurf*** *in diesen Prozeß der Provokation von Kritik stets selber einbezieht.*

verb	real head N	possible head N	prep.	core of PP	PP function
einbezieht	/	*Kunstwerk*	*mit*	*Entwurf*	verb modifier

2. **Postnominal apposition:** If the head noun in the PP is followed by some sort of apposition, this apposition is disregarded.

   (3.79) *Und obwohl mir die Mechanismen der freien Marktwirtschaft völlig fremd waren verlief mein Sprung* ***vom Elfenbeinturm Universität*** *hinein ins kommerzielle Leben besser, ...*

verb	real head N	possible head N	prep.	core of PP	PP function
verlief	*Sprung*	*Sprung*	*vom*	*Elfenbeinturm*	noun modifier

Special PPs

1. **Pronominal adverb and pronominal core:** Pronominal adverbs are placeholders for PPs. They are extracted if they occur in an ambiguous position. But they are marked so that they can be investigated separately from regular PPs. The core of the PP is left open. Pronominal adverbs are similar to PPs with a pronominal core. A personal pronoun core is not extracted since it does not provide information for the PP attachment task. However, the reflexive pronoun *sich* will be extracted since it can be used to identify special verb readings.

   (3.80) *Wie der Magistrat dieser Tage* ***dazu*** *mitteilte, ...*

   (3.81) *Als er zwei Jahre alt war, zogen seine Eltern* ***mit ihm*** *in die damalige DDR.*

   (3.82) *... die aber keine grundlegenden Änderungen* ***mit sich*** *bringen.*

verb	real head N	possible head N	prep.	core of PP	PP function
mitteilte	/	*Tage*	*da-zu*	/	verb modifier
zogen	/	*Eltern*	*mit*	/	verb modifier
bringen	/	*Änderungen*	*mit*	sich	verb modifier

2. **Adverbial or adjectival core:** If the core of the PP is an adverb or an adjective, then this core will be extracted and marked with its part-of-speech. Adverbs and adjectives may help to determine the semantic type of the PP (local, temporal etc.) and thus provide valuable information for the PP attachment.

   (3.83) *Rechenzentren werden noch heute nach den Standards* ***von gestern*** *gebaut.*

   (3.84) *... erst dann werden wir das Gesamtsystem* ***von hier*** *betreiben.*

verb	real head N	possible head N	prep.	core of PP	PP function
gebaut	*Standards*	*Standards*	*von*	gestern	noun modifier
betreiben	/	*Gesamtsystem*	*von*	hier	verb modifier

3. **Comparative phrase:** Comparative phrases with *als, wie* which are annotated as PPs are extracted in the same way as PPs, but they are marked so that they can be investigated separately.

   (3.85) *Theodor Bergmann bilanziert sehr knapp den Sozialismus* ***als offenen Prozeß****, ...*

Automatic comparison of the sixtuples is needed to check the consistency of the annotator judgement. We checked the attachment decisions on the level of quadruples V, N_1, P, N_2 and triples V, P, N_2 and N_1, P, N_2. We also checked full forms and lemmas. For the few contradictions we went back to the sentences to double-check the attachment decision and, if necessary, corrected it in the test set.

From the complete 10,000 sentences of the NEGRA treebank we obtain 6064 sixtuples[13], 2664 with verb attachments (44%) and 3400 with noun attachments (56%). We call this the NEGRA$_{forms}$ test set. Table 3.1 provides a detailed overview of the characteristics of this test set.

The test set contains 2489 verb form types, of which 298 are reflexive verb form types. The possible attachment nouns consist of 4062 types. In 2976 of the noun attachment cases the possible attachment noun is identical to the real attachment noun (87.5%).

The PPs can be distinguished according to the type of preposition. 4747 PPs start with a regular preposition (78%). 1056 PPs are introduced by a contracted preposition (17%), and 111 PPs consist of only a pronominal adverb (2%). Comparative particle phrases occur 145 times (3%). Circumpositions are very rare (only 5 cases). The test cases show 59 different prepositions, 20 contracted preposition types, and 24 pronominal adverb types. For 134 PPs no nominal head (i.e. no noun inside the PP) was found. These PPs contain an adverbal or adjectival head.

In addition to the NEGRA$_{forms}$ test set, we created a lemmatized version which we call the NEGRA$_{lemma}$ test set. Every word form in the sixtuples was matched to its lemma. Lemmatization works as described for the training corpus.

In addition we ran proper name recognition over the NEGRA test sentences. The figures for the proper name tokens in the NEGRA test set given in table 3.1 and table 3.2 on page 88 are based on the automatically recognized names.

[13]This is about double the size of the Penn test set established for English by [Ratnaparkhi et al. 1994]. That test set of 3097 sentences was used in many of the experiments reported in section 2.2.

	NEGRA_{forms}		CZ_{forms}	
number of sixtuples	6064		4562	
noun attachments	3400	56%	2801	61%
verb attachments	2664	44%	1761	39%
all verb form tokens	6064		4562	
reflexive verb form tokens	530	9%	340	7%
all verb form types	2489		1535	
reflexive verb form types	298	12%	163	11%
possible attachment noun form tokens	6064		4562	
including proper name tokens	301	5%	544	12%
possible attachment noun form types	4062		2832	
real attachment noun form tokens	3382		2801	
including proper name tokens	52	2%	123	4%
real attachment noun form types	2368		1720	
possible attachment = real attachment	2962	88%	2474	88%
possible attachment <> real attachment	416	12%	327	12%
preposition tokens	4747	78%	3830	84%
contracted preposition tokens	1056	17%	639	14%
circumposition tokens	5	0%	4	0%
pronominal adverb tokens	111	2%	41	1%
comparative particle tokens	145	3%	48	1%
preposition types	59		56	
contracted preposition types	20		13	
circumposition types	5		3	
pronominal adverb types	24		15	
comparative particle types (*als, wie*)	2		2	
PP core noun form tokens	5930		4520	
including proper name tokens	324	6%	630	14%
PPs without nominal head	134	2%	42	1%
PP core noun form types	3790		2680	

Table 3.1: Comparison of the two test sets

3.2.2 Compilation of a Computer Magazine Treebank

Since the NEGRA corpus domain does not correspond with our training corpus (computer magazine), we manually compiled and disambiguated our own treebank so that we can evaluate our method against test cases from the same domain. We semi-automatically disambiguated 3000 sentences and annotated them in the NEGRA format. In order to be compatible with the German test suite, we used the same annotation scheme as [Skut et al. 1997].

We selected our evaluation sentences from the 1996 volume of the Computer-Zeitung. Thus we ensured that the training corpus (Computer-Zeitung 1993-1995 + 1997) and the test set are distinct. The 1996 volume was prepared (cleaned and tagged) as described in section 3.1. From the tagged sentences we selected 3000 sentences that contained

1. at least one full verb and

2. at least one sequence of a noun followed by a preposition.

With these conditions we restricted the sentence set to those sentences that contained a prepositional phrase in an ambiguous position.

Manually assigning a complete syntax tree to a sentence is a labour-intensive task. This task can be facilitated if the most obvious phrases are automatically parsed. We used our chunk parser for NPs and PPs to speed up the manual annotation. We also used the NEGRA Annotate tool [Brants et al. 1997] to semi-automatically assign syntax trees to all (preparsed) sentences. This tool comes with a built-in parser that suggests categories over selected nodes. The sentence structures were judged by two linguists to minimize errors. Finally, completeness and consistency checks were applied to ensure that every word and every constituent was linked to the sentence structure.

In order to use the annotated sentences for evaluation, we extracted the relevant information from the sentences as described above. From the 3000 annotated sentences we obtain 4562 sixtuples, 1761 with verb attachments (39%) and 2801 with noun attachments (61%). Table 3.1 on the facing page gives the details. The ratio of reflexive verb tokens to all verb tokens and also the distribution of preposition types is surprisingly similar to the NEGRA corpus.

We call this corpus the CZ_{forms} test set. We also created a lemmatized version of this corpus which we call the CZ_{lemma} test set. All verb forms and all noun forms were lemmatized as described above.

We noticed that in the CZ treebank the ratio of proper names to regular nouns (25% proper names, 75% regular nouns) as given by the PoS tags is much higher than in the NEGRA treebank (20.5% proper names). This was to be expected from a market-oriented computer magazine vs. a regular daily newspaper. Therefore, we extracted all proper nouns from the CZ_{lemma} test set and manually classified them as either company name, geographical name, organization name, person name or product name. Table 3.2 gives an overview of the proper name occurrences in this test set.

The proper names of the NEGRA treebank were automatically classified into company name, geographical name and person name. The table thus gives only a rough comparison.

In our experiments we will use the proper name classes to compensate for the sparse data in the proper name tokens (cf. section 4.4.5).

	CZ test set		NEGRA test set	
name class	tokens	types	tokens	types
company names	517	264	15	12
geographical names	231	90	338	138
organization names	97	49		
person names	136	88	324	250
product names	316	171		
total	1297	662	677	400

Table 3.2: Proper names in the test sets

In this chapter we have shown how we processed our corpora and enriched them with linguistic information on different levels: word level information (PoS tags, lemmas), phrasal information (NPs and PPs), and semantic information (proper names, time and location for PPs). In the following chapter we will show how to exploit this information for computing cooccurrence values to disambiguate PP attachments.

Chapter 4

Experiments in Using Cooccurrence Values

4.1 Setting the Baseline with Linguistic Means

In order to appreciate the performance of the statistical disambiguation method, we need to define a baseline. In the simplest form this could mean that we decide on noun attachment for all test cases since noun attachment is more frequent than verb attachment in both test sets (61% to 39% in the CZ test set and 56% to 44% in the NEGRA test set). A more elaborate disambiguation uses linguistic resources. We have access to a list of 466 support verb units and to the verbal subcategorization (subcat) information from the CELEX database.

4.1.1 Prepositional Object Verbs

We use our list of support verb units to disambiguate based on the verb lemma, the preposition and the PP noun (N_2). This leads to 97 correct verb attachment cases for the CZ test set. In section 4.6 we will investigate support verb units in more detail.

In addition we use subcat information from the CELEX database [Baayen et al. 1995]. This database contains subcat information for 9173 verbs (10,931 verbs if reflexive and non-reflexive readings are counted separately). If a verb is classified as requiring a prepositional object, the preposition is supplied. Some examples:

verb	preposition	requirement for the verb
flehen	*um*	prepositional object
warten	*auf* + accusative	optional prepositional object
adressieren	*an* + dative	prepositional object + accusative object
trachten	*nach*	prepositional object + dative object
sich abfinden	*mit*	prepositional object and reflexive pronoun

The CELEX information thus contains the case requirement for a preposition if that preposition governs both accusative and dative NPs (this applies only to the prepositions *an, auf, in, über, unter, vor*). CELEX distinguishes between obligatory and optional subcat requirements and reflexivity requirements.

For a first evaluation we use all CELEX verbs that obligatorily subcategorize for a prepositional object, and we use the verb with the required preposition. If a verb has multiple

prepositional requirements, it will lead to multiple verb + preposition pairs (e.g. *haften für, haften an; votieren für, votieren gegen*). This selection includes verbs that have additional readings without a prepositional object. Reflexive and non-reflexive readings are taken to be different verbs. With these restrictions we extract 1381 pairs. We then use these pairs for an evaluation against the verb lemmas from the CZ test set with the following disambiguation algorithm: If the triple verb + preposition + PP noun is a support verb unit, or if the pair verb + preposition is listed in CELEX, then decide on verb attachment. In the remaining cases use noun attachment as default.

```
if (support_verb_unit(V,P,N2)) then
  verb attachment
elsif (celex_prep_object(V,P)) then
  verb attachment
else
  noun attachment
```

Table 4.1 summarizes the results. In this experiment we used the grammatical case requirement of the preposition for the test cases that contain contracted prepositions. Each contracted preposition is a combination of a preposition and a determiner and thus contains information on dative or accusative. For instance, the contracted form *am* stands for *an* plus the dative determiner *dem*, whereas *ans* contains the accusative determiner *das*. If the test case was *(anschließen, Kabel, ans, Internet)* and CELEX determines that *anschließen* requires a prepositional object with *an* plus accusative, the CELEX information will lead to the desired verb attachment. But if the test case was *(anschließen, Kabel, am, Fernseher)*, then the CELEX information will not trigger an attachment. Each test case with a contracted preposition was compared to the CELEX V+P pair with the appropriate grammatical case requirement of the preposition.

Still, the result is sobering. Only 570 verb attachments can be decided leading to an overall accuracy of 66.12% (percentage of correctly disambiguated test cases). The verb attachments include 97 test cases that were decided based on the support verb units with an accuracy of 100%. But for the other verb attachments the confusion between different verb readings and the disregard of the noun requirements leads to many incorrect attachments.

	correct	incorrect	accuracy
noun attachment	2581	1318	66.20%
verb attachment	374	196	65.61%
total	2955	1514	66.12%

Table 4.1: Attachment accuracy for the CZ_{lemma} test set with prepositional objects from CELEX

4.1.2 All Prepositional Requirement Verbs

In a second experiment we selected those verbs from the CELEX database that have any type of prepositional requirement (obligatory or optional; object or adverbial) but no reading

without a prepositional requirement. That is, we eliminate verbs with non-prepositional readings from the test. For example, the verb *übergehen* has three readings that require a prepositional object (with *auf, in, zu*) according to CELEX. But this verb also has readings without any prepositional requirements.[1] Such verbs are now excluded. On the other hand, a verb such as *warten* has only one reading according to CELEX, but its prepositional requirement is optional. Such verbs are now added. The selection results in 768 verb + preposition pairs. Using these pairs we run the evaluation against our CZ test set and observe the results in table 4.2.

	correct	incorrect	accuracy
noun attachment	2758	1543	64.12%
verb attachment	149	19	88.69%
total	2907	1562	65.05%

Table 4.2: Attachment accuracy for the CZ_{lemma} test set with all prepositional requirements from CELEX

Only a small number of verb attachments can be decided with these CELEX data. If we subtract the 97 cases that are decided by the support verb units, 71 test cases remain that were decided by applying the CELEX verb information. 52 out of these 71 cases were correctly attached (73%). This is not a satisfactory accuracy and covers only a minor fraction of our test cases.

In summary, we find that support verb units are a very reliable indicator of verb attachment but the CELEX data are not. Using linguistic information alone results in an attachment accuracy baseline of **65% to 66%**.

4.2 The Cooccurrence Value

We will now explore various possibilities to extract PP disambiguation information from the annotated corpora. We use the four annotated annual volumes of the Computer-Zeitung (CZ) to gather frequency data on the cooccurrence of nouns + prepositions and verbs + prepositions. We refer to this corpus as the training corpus. After each training we will apply the cooccurrence values for disambiguating the test cases in both the CZ test set and the NEGRA test set.

The cooccurrence value is the ratio of the bigram frequency count $freq(word, preposition)$ divided by the unigram frequency $freq(word)$. For our purposes *word* can be the verb or the reference noun N_1. The ratio describes the percentage of the cooccurrence of *word* + *preposition* against all occurrences of *word*. It is thus a straightforward association measure for a word pair. The cooccurrence value can be seen as the attachment probability of the preposition based on maximum likelihood estimates (cf. [Manning and Schütze 2000] p. 283). We write:

[1] The information whether a verbal prefix is separable is not available to the disambiguation procedure. Sometimes it could help to narrow the search for the correct verb reading: *Bei der letzten Beförderung wurde er übergangen. Bei der letzten Beförderung wurde übergegangen zu einem neuen Anreizsystem.*

$$cooc(W,P) = freq(W,P)/freq(W) \quad \text{with} \quad W \in \{V, N_1\}$$

The cooccurrence values for verb V and noun N_1 correspond to the probability estimates in [Ratnaparkhi 1998] except that Ratnaparkhi includes a back-off to the uniform distribution for the zero denominator case. We will add special precautions for this case in our disambiguation algorithm.

The cooccurrence values are also very similar to the probability estimates in [Hindle and Rooth 1993]. The differences are experimentally compared and discussed in section 7.1.1. They do not lead to improved attachment results.

The methodological difference lies not so much in the association measure nor in the kind of preprocessing. [Ratnaparkhi 1998] uses a PoS tagger and a chunker. [Hindle and Rooth 1993] use a shallow parser. They mostly differ in the extraction heuristics for cooccurring words. Ratnaparkhi uses only unambiguous attachments in the training, whereas Hindle and Rooth use both ambiguous and unambiguous cases. They give stronger weights to unambiguous attachments and evenly split the counts for ambiguous attachments. Our research, reported in this section, shows that raw cooccurrence counts, disregarding the difference between sure attachments and ambiguous attachments, gets us a long way towards the resolution of PP attachment ambiguities, but focussing on the unambiguous attachments will improve the results.

[Krenn and Evert 2001] have evaluated a number of association measures for extracting PP-verb collocations, concentrating on support verb units and figurative expressions. They evaluated Mutual information, Dice coefficient, χ^2 measure, a log-likelihood measure, t-score and a frequency measure. After comparing the results to two corpora, they conclude "that none of the AMs (association measures) is significantly better suited for the extraction of PP-verb collocations than mere cooccurrence frequency".

We start with computing cooccurrence values over word forms as they appear in the training corpus. Their application to the test sets leads to a first attachment accuracy[2] which is surprisingly good. But at the same time the attachment coverage (percentage of decidable cases) is low. A natural language corpus displays an uneven distribution. Few word forms occur very often but most word forms occur very rarely. That means that even in a large corpus many noun forms and verb forms occur with a low frequency and do not provide a sound basis for statistical investigation. Therefore we have to cluster the word forms into classes. We will use lemmatization, de-compounding and semantic classes for proper names as our main clustering methods. We will also explore the use of two semantic classes for PPs (temporal and local) and GermaNet synonym classes.

The goal is to increase the coverage as far as possible without losing attachment accuracy so that in the end only few cases remain for default attachment.

4.3 Experimenting with Word Forms

We will now describe in detail how we compute the cooccurrence values for nouns + prepositions and verbs + prepositions. We list the most frequent nouns, verbs and pairs in tables

[2]We use accuracy to denote the percentage of correctly disambiguated test cases. This corresponds to the notion of precision as used in contrast to recall in other evaluation schemes.

so that the reader gets an insight into the operations and results.

4.3.1 Computation of the N+P Cooccurrence Values

1. **Computation of the noun frequencies.** In order to compute the word form frequency $freq(N_{form})$ for all nouns in our corpus, we count every word that is tagged as regular noun (NN) or as proper name (NE). The tagger's distinction between proper names and regular nouns is not reliable. We therefore discard this distinction for the moment. On the other hand, we do use our corpus annotation of multiword proper names. We collect all elements of such multiword names into one unit (*Bill Gates, New York, Software AG*). We count each unit as one noun.[3] In the case of hyphenated compounds, only the last element is counted here and in all subsequent computations (*Microsoft-Werbefeldzug* → *Werbefeldzug; TK-Umsätze* → *Umsätze*). This reduction is applied only if the element following the hyphen starts with an upper case letter. This avoids reducing *Know-how* or *Joint-venture*.

   From our training corpus we computed the frequency for 188,928 noun form types. The following table contains the top-frequency nouns. These nouns are characteristic of the Computer-Zeitung which reports more on computer business than on technical details. It is surprising that a company name (*IBM*) is among these top frequent words and says something about the influence of this company on the industry. Furthermore, it is striking that the word *Jahr* is represented by two forms among the top ten.

noun N_{form}	$freq(N_{form})$
Prozent	13821
Unternehmen	12615
Mark	9320
Millionen	8710
Dollar	7961
Markt	7620
Software	7588
Jahr	6282
IBM	5573
System	5450
Jahren	4974
Anwendungen	4907

2. **Computation of the noun + preposition frequencies.** In order to compute the pair frequencies $freq(N_{form}, P)$, we search the training corpus for all token pairs in which a noun is immediately followed by a preposition. Noun selection has to be exactly the same as when counting the noun frequencies, i.e. we do not distinguish between proper name and regular noun tags, we do recognize multiword proper names, and for hyphenated compounds only the last word is counted.

[3] Variants of the same proper name (e.g. *Acer Inc.; Acer Group; Acer Computer GmbH*) are not recognized as referring to the same object.

All words tagged as prepositions (APPR) or contracted prepositions (APPRART) are regarded as prepositions. For the moment we disregard pronominal adverbs, circumpositions and comparative particles.

In our training corpus we find 120,666 different noun preposition pairs (types). The pairs with the highest frequencies are in the following table. This list is not very informative. We need to put every pair frequency in relation to the unigram noun frequency in order to see the binding strengths between nouns and prepositions.

noun N_{form}	P	$freq(N_{form}, P)$
Prozent	*auf*	1295
Zugriff	*auf*	986
Markt	*für*	899
Einsatz	*von*	661
Entwicklung	*von*	647
Anbieter	*von*	637
Reihe	*von*	635
Umsatz	*von*	569
Institut	*für*	567
Hersteller	*von*	539

3. **Computation of the noun + preposition cooccurrence values.** The cooccurrence strength of a noun form + preposition pair is called $cooc(N_{form}, P)$. It is computed by dividing the frequency of the pair $freq(N_{form}, P)$ by the frequency of the noun $freq(N_{form})$.

$$cooc(N_{form}, P) = freq(N_{form}, P)/freq(N_{form})$$

Only nouns with a frequency of more than 10 are used. We require $freq(N) > 10$ as an arbitrary threshold. One might suspect that a higher cut-off will lead to more reliable data. In any case it will increase the sparse data problem and lead to more undecidable test cases. We will explore higher cut-off values in section 4.14. For now, this is the top of the resulting cooccurrence value list:

noun N_{form}	P	$freq(N_{form}, P)$	$freq(N_{form})$	$cooc(N_{form}, P)$
Höchstmaß	*an*	13	13	1.00000
Dots	*per*	57	57	1.00000
Bundesinstitut	*für*	12	12	1.00000
Netzticker	*vom*	92	93	0.98925
Hinblick	*auf*	133	135	0.98519
Verweis	*auf*	21	22	0.95455
Umgang	*mit*	293	307	0.95440
Bundesministeriums	*für*	35	37	0.94595
Bundesanstalt	*für*	70	75	0.93333
Synonym	*für*	13	14	0.92857
Verzicht	*auf*	51	55	0.92727
Rückbesinnung	*auf*	12	13	0.92308

There are four noun forms with a perfect cooccurrence value of 1.0. For example *Höchstmaß* occurs 13 times in the training corpus and every time it is followed by the preposition *an*. The top ten list comprises three names of governmental organizations *Bundes** and one deverbal noun (*Rückbesinnung*). It also comprises one technical term from computer science (*Dots*) which occurs often in the phrase *Dots per Inch*.

4.3.2 Computation of the V+P Cooccurrence Values

The treatment of verb + preposition (V+P) cooccurrences is different from the treatment of N+P pairs since verb and preposition are seldom adjacent to each other in a German sentence. On the contrary, they can be far apart from each other, the only restriction being that they have to cooccur within the same clause. A clause is defined as a part of a sentence with one full verb and its complements and adjuncts. Only in the case of verb coordination a clause can contain more than one full verb. Clause boundary tags have been automatically added to our training corpus as described in section 3.1.7. Only clauses that contain exactly one full verb are used for the computation of the verb frequencies $freq(V_{form})$ and the pair frequencies $freq(V_{form}, P)$.

1. **Computation of the verb frequencies.** We count all word forms that have been tagged as full verbs (in whatever form). We are not interested in modal verbs and auxiliary verbs since prepositional phrases do not attach to them. Copula verbs are tagged as auxiliary verbs and are thus not counted. A separated verbal prefix is reattached to the verb during the computation.[4]

   Contrary to nouns, verbs often have more than one prepositional phrase attached to them. Therefore we count a verb as many times as there are prepositions in the same clause, and we count it once if it does not cooccur with any preposition. This procedure corresponds to the counting of nouns in which a noun is counted once if it cooccurs with a preposition and once if it occurs without one. Sentence 4.1 consists of two clauses. In the first clause the verb *bauen* is counted once since it cooccurs with the preposition *für*. In the second clause the verb *arbeiten* is counted twice since it cooccurs with both *an* and *mit*. Sentence 4.2 does not contain any PP, therefore the verb *ankündigen* is counted once.

   This manner of counting the verb frequencies assumes that a clause with two PPs ($V...PP_x...PP_y$) is the same as two clauses with one PP each ($V...PP_x$) and ($V...PP_y$). In other words, it assumes that the attachment of the two PPs to the verb is independent of each other. For verbal complements that is certainly not true. If a verb cooccurs with a certain PP complement, this choice delimits whether and which other complements it may accept. But for adjunct PPs the independence assumption is not a problem. A verb may take an open number of adjuncts. Since we do not distinguish between complements and adjuncts, we work with the independence assumption.

[4]The reattachment of the separated prefix to the verb is a possible source of errors. The PoS tagger has problems distinguishing between the right element in a circumposition (*Er erzählte das von sich* **aus**.) and a separated prefix (*Das Licht geht von allein* **aus**.). If such a circumposition element is mistagged as a separated prefix, it will get attached to the verb and lead to an ungrammatical verb (e.g. **auserzählte*). Fortunately, circumpositions are rare so that this tagging problem does not have a significant impact on our results.

(4.1) *So will Bull PCMCIA-Smartcard-Lesegeräte und Anwendungen* ***für NT-Netze*** *bauen, und Hewlett-Packard arbeitet* ***an Keyboards mit integriertem Lesegerät****.*

(4.2) *Einige kleinere Schulungsanbieter haben bereits ihre Schließung angekündigt.*

We collect a total of 18,726 verb form types from our corpus. The most frequent forms are listed in the following table. Note that the two verbs *stehen* and *kommen* are represented by two forms each in this top frequency list.

verb V_{form}	$freq(V_{form})$
gibt	5289
entwickelt	4044
stehen	3853
kommen	3764
steht	3669
bietet	3539
liegt	3270
machen	3065
kommt	3048
unterstützt	2789

2. **Computation of all verb + preposition pair frequencies.** We count all token pairs where a verb and a preposition cooccur in a clause. Example sentence 4.3 consists of two clauses with the verb forms *läuft* and *sparen*. Both clauses contain 3 prepositions. This will lead to the verb + preposition pairs *läuft in*, *läuft bis*, *läuft zum*, *sparen bei*, *sparen gegenüber*, and *sparen von*.

   (4.3) ***In Deutschland*** *läuft noch* ***bis zum 31. Januar*** *eine Sonderaktion,* $\langle CB \rangle$ ***bei welcher*** *der Anwender immerhin 900 Mark* ***gegenüber dem Listenpreis von 1847 Mark*** *sparen kann.*

   In this way we obtain 93,473 verb + preposition pairs.

3. **Computation of the verb + preposition cooccurrence values.** As for the N+P pairs, the cooccurrence strength of a verb + preposition pair is computed by dividing the frequency computed for the V+P pair with the frequency associated with the verb form.

$$cooc(V_{form}, P) = freq(V_{form}, P)/freq(V_{form})$$

   We apply the same cut-off criterion as with nouns. Only verb forms with a minimum frequency of more than 10 are used. We thus get cooccurrence values for 70,877 verb + preposition pairs (types). Here is the top of the resulting list:

verb V_{form}	P	$freq(V_{form}, P)$	$freq(V_{form})$	$cooc(V_{form}, P)$
logiert	*unter*	55	56	0.98214
paktiert	*mit*	13	14	0.92857
verlautet	*aus*	16	19	0.84211
gliedert	*in*	29	35	0.82857
getaktet	*mit*	79	101	0.78218
herumschlagen	*mit*	21	27	0.77778
besinnen	*auf*	17	22	0.77273
auszustatten	*mit*	38	50	0.76000
bangen	*um*	14	19	0.73684
heranzukommen	*an*	11	15	0.73333

The verb form *logiert* occurs 56 times and in 55 clauses it is accompanied by the preposition *unter* leading to the top cooccurrence value of 0.98. Note that this list contains one computer specific verb *takten* which has a high cooccurrence value with *mit*.

4.3.3 Disambiguation Results Based on Word Form Counts

With the N+P and V+P cooccurrence values for word forms we do a first evaluation over our test sets. From the sixtuples in the test sets we disregard the noun within the PP at the moment. We skip all test cases where the PP is not introduced by a preposition or by a contracted preposition (but by a circumposition, a comparative particle or a pronominal adverb). Furthermore, we skip all test cases where the possible attachment noun (that is the one giving rise to the ambiguity) is not identical to the real attachment noun. In these cases it is debatable whether to use the real attachment noun or the possible attachment noun for our experiments, and we will concentrate on the clear cases first.

For the CZ_{forms} test set these restrictions leave us with 4142 test cases. It turns out that for 2336 of these test cases we have obtained both cooccurrence values $cooc(N, P)$ and $cooc(V, P)$ in the training. The disambiguation algorithm in its simplest form is based on the comparison of the competing cooccurrence values for N+P and V+P. It does not include default attachment:

```
if ( cooc(N,P) && cooc(V,P) ) then
  if ( cooc(N,P) >= cooc(V,P) ) then
     noun attachment
  else
     verb attachment
```

The disambiguation results are summarized in table 4.3.

The **attachment accuracy** (percentage of correct attachments) of 71.40% is higher than the baseline but still rather disappointing. But we notice a striking imbalance between the noun attachment accuracy (almost 94%) and the verb attachment accuracy (55%). This means that our cooccurrence values favor verb attachment. The comparison of the verb cooccurrence value and the noun cooccurrence value too often leads to verb attachment, and only the clear cases of noun attachment (i.e. the cases with a very strong tendency of noun attachment over verb attachment) remain. We observe an inherent imbalance between the

	correct	incorrect	accuracy
noun attachment	925	60	93.91%
verb attachment	743	608	55.00%
total	1668	668	71.40%

Table 4.3: Attachment accuracy for the CZ_{forms} test set.

cooccurrence values for verbs and nouns.[5] We propose to flatten out this imbalance with a noun factor.

The noun factor

The noun factor is supposed to strengthen the N+P cooccurrence values and thus to attract more noun attachment decisions. The noun attachment accuracy will suffer from the influence of the noun factor but the verb attachment accuracy and the overall accuracy will profit.

What is the rationale behind the imbalance between noun cooccurrence value and verb cooccurrence value? One influence is certainly the well-known fact that verbs bind their complements stronger than nouns. The omission of an obligatory verbal complement makes a sentence ungrammatical, whereas there are hardly any noun complements that are obligatory with the same rigidity. If we compare the cooccurrence values of verbs and their derived nouns, this difference becomes evident:

word W	P	$freq(W,P)$	$freq(W)$	$cooc(W,P)$
arbeiten	*an*	778	5309	0.14654
Arbeit	*an*	142	3853	0.03685
reduzieren	*auf*	219	1285	0.17043
Reduktion	*auf*	1	94	0.01064
warnen	*vor*	196	637	0.30769
Warnung	*vor*	10	78	0.12821

The imbalance between noun cooccurrence values and verb cooccurrence values can be quantified by comparing the overall tendency of nouns to cooccur with a preposition to the overall tendency of verbs to cooccur with a preposition. We compute the overall tendency as the cooccurrence value of all nouns with all prepositions. It is thus computed as the frequency of all N+P pairs divided by the frequency of all nouns.

$$cooc(all\_N, all\_P) = \sum_{(N,P)} freq(N,P) \quad / \sum_{N} freq(N)$$

The computation for the overall verb cooccurrence tendency is analogous. For the noun forms and verb forms in the CZ training corpus we get the following results:

[5] [Hindle and Rooth 1993] also report on this imbalance for English: 92.1% correct noun argument attachments and 84.6% correct verb argument attachments; 74.7% correct noun adjunct attachments and 64.4% correct verb adjunct attachments.

- $cooc(all\_N_{forms}, all\_Ps) = \frac{314{,}028}{1{,}724{,}085} = 0.182$
- $cooc(all\_V_{forms}, all\_Ps) = \frac{462{,}185}{596{,}804} = 0.774$

In our training corpus we have found 314,028 N+P pairs (tokens) and 1.72 million noun tokens. This leads to an overall noun cooccurrence value of 0.182. The noun factor is then the ratio of the overall verb cooccurrence tendency divided by the overall noun cooccurrence tendency:

$$noun\_factor = \frac{cooc(all\_V, all\_P)}{cooc(all\_N, all\_P)}$$

This leads to a noun factor of 0.774/0.182 = 4.25. In the disambiguation algorithm we multiply the noun cooccurrence value with this noun factor before comparing it to the verb cooccurrence value. Our disambiguation algorithm now works as:

```
if ( cooc(N,P) && cooc(V,P) ) then
  if ( (cooc(N,P) * noun_factor) >= cooc(V,P) ) then
     noun attachment
  else
     verb attachment
```

	factor	correct	incorrect	accuracy
noun attachment	4.25	1377	280	83.10%
verb attachment		524	157	76.94%
total		1901	437	81.31%
decidable test cases		2338 (of 4142) coverage: 57%		

Table 4.4: Attachment accuracy for the CZ_{forms} test set using the noun factor.

Table 4.4 shows that attachments based on the cooccurrence values of raw word forms are correct in 1901 out of 2338 test cases (81.31%) when we employ the noun factor of 4.25. It clearly exceeds the level of attachment accuracy of our pilot study (76%) where we evaluated only against some hundred sentences (see [Mehl et al. 1998]). But it is striking that we can decide the attachment only for 57% of our test cases (2338 out of 4142).

The imbalance between noun and verb attachment accuracy is now smaller but persists at 6% difference. If we try to come to a (near) perfect balance, we need to increase the noun factor to 5.2 which will give us the results in table 4.5.

There are three main reasons that speak against this solution. First, the attachment accuracy is worse than with the empirically founded noun factor of 4.25. Second, the judgement of balance between noun and verb attachment accuracy is based on the test cases and thus a supervised aspect in the otherwise unsupervised approach. Third, we would expect that the ratio of the number of all noun attachments to the number of all verb attachments reflects the ratio of noun attachments to verb attachments in the test set. We find that among the 2338 solved test cases there are 66% manually determined noun attachments and 34% verb

	factor	correct	incorrect	accuracy
noun attachment	5.2	1419	342	80.58%
verb attachment		461	116	79.90%
total		1880	458	80.41%
decidable test cases		2338 (of 4142) coverage: 57%		

Table 4.5: Balanced attachment accuracies for the CZ_{forms} test set using the noun factor.

attachments. The noun factor of 4.25 leads to 71% noun attachments which is still 5% away from the expected value. But the noun factor of 5.2 leads to 75% noun attachments which is clearly worse. Therefore we stick to the noun factor as defined above and accept that it leads to an imbalance between noun and verb attachment accuracy.

Support for this noun factor computation and application also comes from the observation that a noun factor of 4.25 leads to the maximum overall attachment accuracy for the given data. We evaluated with noun factors from 1 to 10 in steps of 0.25 and found that 4.25 gives the best result. See figure 4.1 for a plot of the noun factor effects on the attachment accuracy.

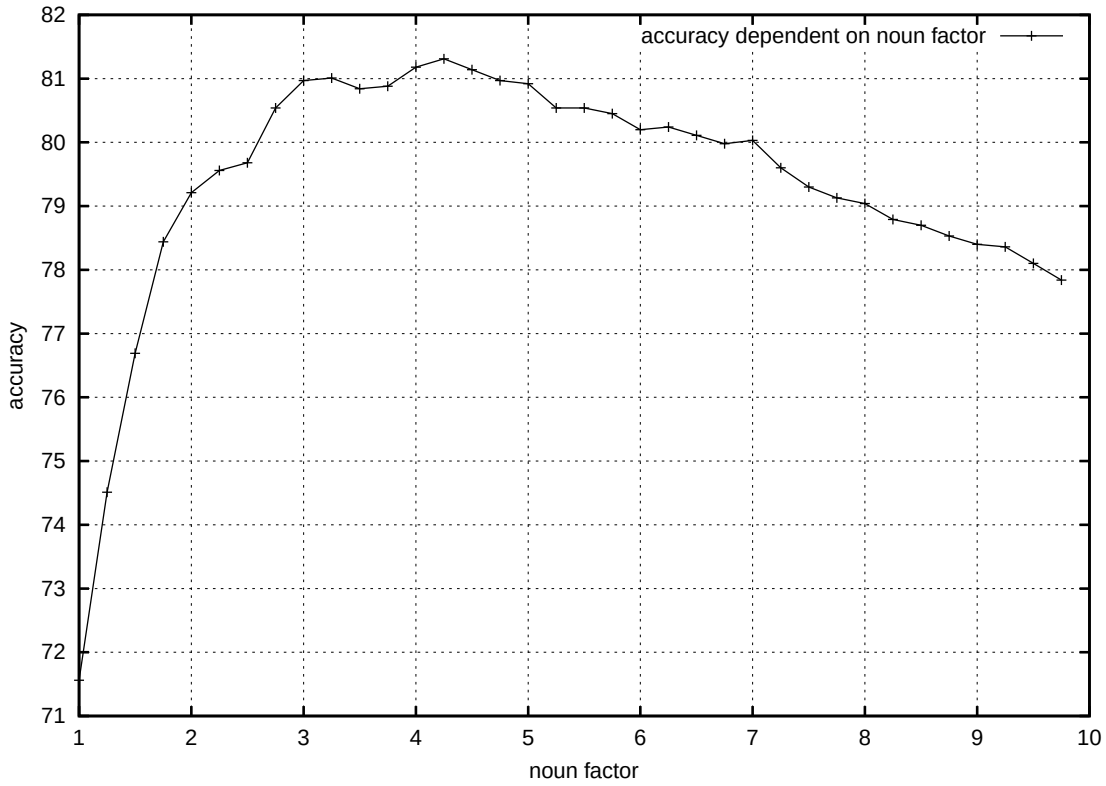


Figure 4.1: Accuracy as a function of the noun factor (for word form counts).

	factor	correct	incorrect	accuracy
noun attachment	4.25	917	264	77.64%
verb attachment		338	140	70.71%
total		1255	404	75.65%
decidable test cases		1659 (of 5387) coverage: 31%		

Table 4.6: Attachment accuracy for the NEGRA$_{forms}$ test set using the noun factor.

We also checked the influence of differing noun factors based on individual prepositions. The computation of a preposition-specific noun factor is analogous to the computation of the overall noun factor except that we sum separately for each preposition.

preposition P	$freq(all\_V, P)$	$freq(all\_N, P)$	$noun\_factor(P)$
entgegen	59	2	85.22146
laut	1761	230	22.11864
neben	2017	268	21.74193
vorbehaltlich	6	1	17.33318
abzüglich	5	1	14.44432
seit	3108	660	13.60392
angesichts	338	72	13.56161
...			...
samt	122	138	2.55392
mitsamt	14	16	2.52776
namens	265	336	2.27842
fürs	74	111	1.92591
beiderseits	2	3	1.92591
versus	9	27	0.96295
kontra	1	6	0.48148

This table shows that the preposition *entgegen* has the strongest tendency to cooccur with verbs in contrast to nouns. In our corpus it occured 59 times with a verb but only twice following a noun. These raw cooccurrence frequencies are divided by the frequency of all verbs (596,804) and all nouns (1,724,085) respectively before the resulting two ratios are divided to give the preposition-specific noun factor. The bottom end of the list shows prepositions that are more likely to cooccur with a noun than with a verb.

The use of these preposition-specific noun factors did not result in an improvement of the attachment accuracy (instead it resulted in a noticeable decrease to 79%). We therefore continue to work with the general noun factor.

Let us compare the results of the CZ evaluation to our second test set, the NEGRA$_{forms}$ set. We apply the same restrictions and are left with 5387 test cases (= 6064 - 416 possible attachment <> real attachment - 5 circumposition cases - 111 pronominal adverb cases - 145 comparative particle cases). The disambiguation results are summarized in table 4.6.

The attachment accuracy is 75.65% and thus significantly lower than for the CZ_{forms} corpus. Furthermore the attachment coverage of 31% (1659 out of 5387) is way below the value for the CZ_{forms} corpus. This indicates that our method is dependent on the training corpus both in terms of attachment accuracy and coverage. The computation of the cooccurrence values over the same text type as the test set leads to significantly better results.

In general, we must increase the attachment coverage without a decrease in the attachment accuracy. That means we have to investigate various methods to tackle the sparse data problem.

4.3.4 Possible Attachment Nouns vs. Real Attachment Nouns

But first, we need to look at the test cases that were left out due to the difference between the possible attachment noun and the real attachment noun. We illustrate the problem with an example. In sentence 4.4 the PP *zur Verwandlung* is in an ambiguous position since it follows immediately after the noun *Menschen*. There, the noun *Menschen* is considered the possible attachment site. But in this sentence the PP is attached neither to this possible attachment noun nor to the verb but to a noun earlier in the sentence, *Lust*. We call that noun the real attachment noun. In 88% of the noun attachment cases in the CZ test set the PP attaches to the immediately preceding noun. Only for 12% we have an intervening noun.

Then the PP has a choice between three attachments sites, and in order to resolve this ambiguity, we will have to compare the cooccurrence values for all three possible sites. For the moment we make the simplifying assumption that the possible attachment noun is not present and therefore the real attachment noun is triggering the ambiguity. This corresponds to turning sentence 4.4 into 4.5.

(4.4) *Andererseits beflügelt die Maske die Lust des Menschen* ***zur Verwandlung****.*

(4.5) *Andererseits beflügelt die Maske die Lust* ***zur Verwandlung****.*

By accepting the real attachment noun as the ambiguity trigger, we add all test cases with a difference between the real and the possible attachment noun to the test set. For the CZ_{forms} corpus we then have 4469 test cases.

	factor	correct	incorrect	accuracy
noun attachment	4.25	1507	280	84.33%
verb attachment		524	214	71.00%
total		2031	494	80.43%
decidable test cases		2525 (of 4469) coverage: 57%		

Table 4.7: Attachment accuracy for the extended CZ_{forms} test set using the noun factor.

The disambiguation algorithm based on word form counts decides 2525 out of 4469 test cases corresponding to an attachment coverage of 57%. This coverage rate is the same as before but we notice a loss of almost 1% in the attachment accuracy after the integration of the additional test cases. With this in mind, we will include these test cases in the subsequent tests.

4.4 Experimenting with Lemmas

The first step to reduce the sparse data problem and to increase the attachment coverage is to map all word forms to their lemmas (i.e. their base forms). Since the lemma information is already included in our corpora (cf. section 3.1.4), we will now use the lemmas for the computation of the cooccurrence values instead of the word forms. We expect a small decrease in the number of noun types but a substantial decrease in the number of verb types since German verbs have up to 15 different forms.[6]

4.4.1 Noun Lemmas

1. **Computation of the noun lemma frequencies.** In order to compute the lemma frequencies $freq(N_{lem})$ for all nouns in our corpus, we count the lemmas of all words tagged as regular noun (NN) or as proper name (NE). In a first approach the lemma of a compound noun is the base form of the complete compound (*Forschungsinstituts* → *Forschungsinstitut*) rather than the base form of its last element. In the case of hyphenated compounds only the lemma of the last element is counted. Again, we discard the distinction between proper names and regular nouns but we use multiword names. For all nouns and names without a lemma we use the word form itself.

   From our training corpus we compute the frequency for 161,236 noun lemma types (compared to 188,928 noun form types). The number of noun lemma types is only 15% lower than the number of noun form types. In other words, most nouns occur only in one form in our corpus. This is the top of the noun lemma frequency list.

noun N_{lem}	$freq(N_{lem})$
Jahr	16734
Unternehmen	14338
System	14334
Prozent	13823
Mark	9321
Million	9153
Markt	8958
Dollar	7998
Software	7594
Produkt	6722

2. **Computation of the noun lemma + preposition frequencies.** In order to compute the $freq(N_{lem}, P)$ we count all token pairs (noun lemma, preposition) where a noun is immediately followed by a preposition. Noun lemma selection is exactly the same as when counting the noun lemma frequencies.

   All words tagged as prepositions (APPR) or contracted prepositions (APPRART) are considered as prepositions. All contracted prepositions are mapped to their base form counterparts (e.g. *am* → *an*, *zur* → *zu*). We disregard pronominal adverbs, circumpositions and comparative particles.

[6] Consider the verb *fahren* with its forms: *ich fahre, du fährst, er fährt, wir fahren, ihr fahrt, ich fuhr, du fuhrst, wir fuhren, ihr fuhrt, ich führe, du führest, er führe, wir führen, ihr führet, gefahren.*

From our training corpus we compute the frequency for 100,040 noun lemma + preposition pairs (compared to 120,666 noun form + preposition pairs).

3. **Computation of the noun lemma + preposition cooccurrence values.** The cooccurrence values of a noun lemma + preposition pair is called $cooc(N_{lem}, P)$. It is computed in the same way as for the word forms, i.e. by dividing the frequency of the pair $freq(N_{lem}, P)$ by the frequency of the noun lemma $freq(N_{lem})$. Only noun lemmas with a minimum frequency of more than 10 are used. Here is the top and the bottom of the resulting list:

noun N_{lem}	P	$freq(N_{lem}, P)$	$freq(N_{lem})$	$cooc(N_{lem}, P)$
Höchstmaß	*an*	13	13	1.00000
Dots	*per*	57	57	1.00000
Bundesinstitut	*für*	16	16	1.00000
Hinblick	*auf*	133	135	0.98519
Abkehr	*von*	40	41	0.97561
Netzticker	*von*	92	95	0.96842
Umgang	*mit*	300	314	0.95541
...				...
Prozent	*trotz*	1	13823	0.00007
Prozent	*ohne*	1	13823	0.00007
Prozent	*jenseits*	1	13823	0.00007
Jahr	*zugunsten*	1	16734	0.00006
Jahr	*trotz*	1	16734	0.00006
Jahr	*statt*	1	16734	0.00006

4.4.2 Verb Lemmas

1. **Computation of the verb lemma frequencies.** In order to compute the verb lemma frequencies $freq(V_{lem})$ we count all lemmas for which the word form has been tagged as a full verb. A separated verbal prefix is reattached to the verb lemma during the computation. Like verb forms, verb lemmas are counted as many times as there are prepositions in the same clause. And we count the lemma once if it does not cooccur with any preposition.

   We collect a total of 8061 verb lemma types from our corpus (compared to 18,726 verb form types this is a 57% reduction). The most frequent lemmas are listed in the following table. Note that the frequencies are now much higher since they are combined from all verb forms. The verb form *kommen* used to have a frequency of 3764 but now its lemma has a frequency of 9082.

verb V_{lem}	$freq(V_{lem})$
kommen	9082
geben	8926
stehen	8650
machen	7026
entwickeln	6605
liegen	6600
anbieten	5755
bieten	5732
gehen	5441
arbeiten	5309

2. **Computation of all verb lemma + preposition pair frequencies.** In order to compute $freq(V_{lem}, P)$ we count all token pairs where the verb and a preposition cooccur in a clause. All contracted prepositions are reduced to their base form counterparts. Circumpositions, pronominal adverbs and comparative particles are disregarded.

   In this way we obtain 45,745 verb lemma + preposition pairs (compared to 93,473 verb form + preposition pairs).

3. **Computation of the verb lemma + preposition cooccurrence values.** The cooccurrence value of a verb lemma + preposition pair is computed as for the word forms. Only verb lemmas with a minimum frequency of more than 10 are used. We get cooccurrence values for 37,437 verb lemma + preposition pairs (compared to 70,877 verb form + preposition pairs). Here is the top of the resulting list:

verb V_{lem}	P	$freq(V_{lem}, P)$	$freq(V_{lem})$	$cooc(V_{lem}, P)$
logieren	*unter*	55	56	0.98214
heraushalten	*aus*	10	11	0.90909
abfassen	*in*	9	11	0.81818
herumschlagen	*mit*	29	36	0.80556
takten	*mit*	86	115	0.74783
paktieren	*mit*	14	19	0.73684
assoziieren	*mit*	8	11	0.72727
protzen	*mit*	13	18	0.72222
herangehen	*an*	13	18	0.72222
besinnen	*auf*	26	36	0.72222

   For some of the verbs the cooccurrence value has not changed much from the word form count (e.g. *logieren unter, herumschlagen mit, takten mit*). However, *paktieren mit* has decreased from 0.93 to 0.74. For such low frequency verbs the lemmatization will often provide (slightly) higher frequencies and thus more reliable cooccurrence values. It is also striking that three values in the top ten are based on the minimum frequency of 11 (*heraushalten aus, abfassen in, assoziieren mit*).

4.4.3 Disambiguation Results Based on Lemma Counts

With the N+P and V+P cooccurrence values for lemmas we perform a second round of evaluations over our test sets. We continue to skip all test cases in which the PP is not introduced by a preposition or by a contracted preposition.

For the CZ_{lemma} test set these restrictions leave us with 4469 test cases. For 3238 of these test cases we have both cooccurrence values $cooc(N_{lem}, P)$ and $cooc(V_{lem}, P)$. The result is summarized in table 4.8.

	factor	correct	incorrect	accuracy
noun attachment	4.25	1822	391	82.33%
verb attachment		711	314	69.37%
total		2533	705	78.23%
decidable test cases		3238 (of 4469) coverage: 72%		

Table 4.8: Attachment accuracy for the CZ_{lemma} test set.

We notice a 2% loss in **attachment accuracy** (from 80.43% to 78.23%) but a sharp rise in the **attachment coverage** from 57% to 72%. The latter is based on the fact that the combined frequencies of all forms of a lemma may place it above the minimum frequency threshold, whereas the frequencies for the forms were below the threshold and therefore the forms could not be used for the cooccurrence computations.

The loss in accuracy could either be based on using the lemmas or on higher difficulties in the additionally resolved test cases. We therefore reran the test only on those 2525 test cases that were previously resolved based on the word forms (with 80.43% accuracy). The lemma-based test resulted in 79.82% accuracy. This means that we lose about 0.5% accuracy due to the shift from word forms to lemmas, and the remaining 1.5% loss is due to the additional test cases. It is clear that lemmatization may lead to some loss in accuracy since some forms of different words are mapped to the same lemma. For example, both *Datum* and *Daten* are mapped to the lemma *Datum*. It would be desirable to avoid this and rather stick with the word form if the lemma is not unique.

Let us compare the CZ test results to the results for the $NEGRA_{lemma}$ test set. We apply the same restrictions and are left with 5803 test cases (= 6064 - 5 circumposition cases - 111 pronominal adverb cases - 145 comparative particle cases). Table 4.9 shows the results.

The disambiguation results for the $NEGRA_{lemma}$ test set are analogous to the results for the CZ_{lemma} test set. Again we notice a 2% loss in attachment accuracy and a 13% rise in the attachment coverage (to 44%) compared to the $NEGRA_{forms}$ experiment in table 4.6.

4.4.4 Using the Core of Compounds

Lemmatizing is a way of clustering word forms into lemma classes. The noun lemmas that we used above had only a small effect on reducing the number of noun types (15% reduction) compared to the verb lemmas (57% reduction). This is due to the large number of nominal compounds in German.

We proceed to use only the last element of a nominal compound for lemmatization (*Forschungsinstituts* → *Institut*). For this we exploit our lemmatizer's ability to segment

	factor	correct	incorrect	accuracy
noun attachment	4.25	1354	359	79.04%
verb attachment		533	331	61.69%
total		1887	690	73.22%
decidable test cases		2577 (of 5803) coverage: 44%		

Table 4.9: Attachment accuracy for the NEGRA$_{lemma}$ test set.

compounds and to mark compound boundaries.

We make the simplifying assumption that the behavior of a noun with respect to preposition cooccurrence is dependent on its last element, the core noun. We call the lemma of the core noun the **short lemma** of the compound in order to distinguish it from the lemma of the complete compound. For non-compounded nouns we use the regular lemma as before.

The table shows the results of the use of short lemmas with respect to the number of types in our corpus:

	$freq(N)$ types	$freq(N,P)$ types	$cooc(N,P)$ types
word forms	188,928	120,666	69,072
lemmas	161,236	100,040	56,876
short lemmas	80,533	60,958	44,151

Obviously the number of short lemmas is much smaller than the number of complete lemmas. The frequencies of many nouns and pairs will thus be higher and lead to a wider coverage of the cooccurrence values, i.e. a higher attachment coverage.

Using the same restrictions as in the above experiments, our test set CZ$_{shortlemma}$ consists of 4469 test cases. For 3687 of these test cases we now have both cooccurrence values $cooc(N_{slem}, P)$ and $cooc(V_{lem}, P)$. The result is summarized in table 4.10. There is no loss in attachment accuracy but a substantial rise in the attachment coverage from 72% to 83% (the number of decidable cases).

	factor	correct	incorrect	accuracy
noun attachment	4.25	1997	400	83.31%
verb attachment		885	403	68.71%
total		2882	803	78.21%
decidable test cases		3685 (of 4469) coverage: 83%		

Table 4.10: Attachment accuracy for the CZ$_{shortlemma}$ test set.

In our second evaluation with the NEGRA$_{shortlemma}$ test set we apply the same restrictions as above and are left with 5803 test cases. The result is shown in table 4.11.

The loss in attachment accuracy for the NEGRA$_{shortlemma}$ test set is more visible than for the CZ$_{shortlemma}$ test set. Here we notice a 1.5% loss in attachment accuracy but a 17% rise in the attachment coverage to 61% (3507 out of 5803 cases can now be decided).

	factor	correct	incorrect	accuracy
noun attachment	4.25	1736	460	79.05%
verb attachment		813	498	62.01%
total		2549	958	72.68%
decidable test cases		3507 (of 5803) coverage: 61%		

Table 4.11: Attachment accuracy for the NEGRA$_{shortlemma}$ test set.

Another possible simplification is the reduction of female forms to male forms (*Mitarbeiterin/MitarbeiterIn → Mitarbeiter*). This will help to avoid the usual low frequencies of the female forms. But even with the help of Gertwol's segment boundary information this mapping is not trivial since umlauts and elision are involved (*Philolog-in → Philolog-e*; *Studienrät-in → Studienrat*).

Furthermore we considered the reduction of diminuitive forms ending in *-chen* or *-lein*, but these occur very rarely in our corpus. The most frequent ones are *Teilchen* (38), *Brötchen* (17), and *Kästchen* (14 times). Some diminuitive forms do not have a regular base form (*Wehwehchen *Wehweh; Scherflein *Scherf*). Some have taken on a lexicalized meaning (*Brötchen, Hintertürchen, Fräulein*).

During the course of the project we found that we might also cluster different nominalizations of the same verb (*das Zusammenschalten, die Zusammenschaltung → das Zusammenschalten*). In addition all number words fall in the same class and could be clustered (*Hundert, Million, Milliarde*). The same is true of measurement units (*Megahertz, Gigahertz; Kilobyte, Megabyte*). Some nominal prefixes that lead to weak segmentation boundaries in Gertwol could still lead to reduced forms (*Vizepräsident → Präsident*). Clustering is also possible over abbreviations (*Megahertz, MHz*). These reduction methods have not been explored.

4.4.5 Using Proper Name Classes

When we checked the undecidable test cases from our previous experiments, we noticed that proper names are involved in many of these cases. In evaluating against the CZ$_{shortlemma}$ test set, we were left with 782 undecidable cases. These can be separated into cases in which the $cooc(N, P)$ or the $cooc(V, P)$ or both are missing.

only $cooc(N, P)$ missing	567	73%
only $cooc(V, P)$ missing	164	21%
both $cooc(N, P)$ and $cooc(V, P)$ missing	51	6%
total number of undecidable cases	782	100%

When we analyse the 567 test cases of missing $cooc(N, P)$ we find that in almost half of these (277 cases) the reference noun is a proper name.[7] The proper names are distributed as follows:

[7] In addition there are 10 cases involving proper names among the 51 cases where both $cooc(N, P)$ and $cooc(V, P)$ are missing.

name class	undecidable	all name cases
company names	103	217
geographical names	17	46
organization names	23	37
person names	59	66
product names	75	100
total	277	466

The $CZ_{shortlemma}$ test set contains a proper name as attachment noun in 466 test cases (out of 4469 test cases). Only 189 of these cases can be resolved using the lemma (substituted by the word form if no lemma is found).

We therefore change the computation of our cooccurrence values. We now compute the cooccurrence values for the semantic name classes rather than for the proper names individually. For example, we compute the cooccurrence values of the class of company names with all prepositions. All company names are subsumed into this class. We perform this computation for company names, geographical names and person names since these names were automatically annotated in our training corpus.

With this clustering we reduce the number of noun types and we get high token frequencies for the three semantic classes. Company names are by far the most frequent in the CZ training corpus. Person names and geographical names have about the same frequency.

class	$freq(class)$
company names	115,343
geographical names	41,100
person names	39,368

The number of noun types is substantially reduced from 80,500 to 56,000. These 24,500 types are now subsumed under the three proper name classes.

	$freq(N)$ types	$freq(N,P)$ types	$cooc(N,P)$ types
word forms	188,928	120,666	69,072
lemmas	161,236	100,040	56,876
short lemmas	80,533	60,958	44,151
short lemmas and name classes	55,968	50,356	38,374

Assuming that all names within a semantic name class behave similarly towards the prepositions, we expect to increase the attachment coverage without losing attachment accuracy. And this is exactly what we observe (see table 4.12). The attachment accuracy increases slightly to 78.36% (compared to 78.21% in table 4.10), but the attachment coverage increases from 83% to 86% (3850 out of 4469 cases are decidable). Note that in this experiment all company names, geographical names and person names were mapped to their semantic classes, including the ones that previously had cooccurrence values via their word form or lemma.

We also ran the same test against the NEGRA test set (see table 4.13). We observe an increase of 2% on the coverage and an improvement of 1% on the attachment accuracy. This result is a more realistic improvement since it is based on the automatically recognized proper names in the NEGRA test set, whereas the proper names in the CZ test set were manually annotated.

When we mention the test sets as $CZ_{shortlemma}$ or $NEGRA_{shortlemma}$ in the following sections, this will include the name class symbols as lemmas for the proper names.

	factor	correct	incorrect	accuracy
noun attachment	4.25	2034	412	83.16%
verb attachment		983	421	70.01%
total		3017	833	78.36%
decidable test cases		3850 (of 4469) coverage: 86%		

Table 4.12: Attachment accuracy for the $CZ_{shortlemma}$ test set with names.

	factor	correct	incorrect	accuracy
noun attachment	4.25	1756	490	78.18%
verb attachment		944	509	64.97%
total		2700	999	72.99%
decidable test cases		3699 (of 5803) coverage: 64%		

Table 4.13: Attachment accuracy for the $NEGRA_{shortlemma}$ test set with names.

4.4.6 Using the Cooccurrence Values against a Threshold

So far we have increased the attachment coverage by clustering the corpus tokens into classes. A second way of tackling the sparse data problem lies in using partial information. Instead of insisting on both $cooc(N, P)$ and $cooc(V, P)$ values, we can back off to either value for those cases with only one value available. Comparing this value against a given threshold we decide on the attachment. If, for instance, $cooc(N, P)$ is available (but no $cooc(V, P)$ value), and if this value is above a $threshold(N)$, then we decide on noun attachment. If $cooc(N, P)$ is below the threshold, we take no decision.

```
if ( cooc(N,P) && cooc(V,P) ) then
  if ( (cooc(N,P) * noun_factor) >= cooc(V,P) ) then
     noun attachment
  else
     verb attachment
elsif ( cooc(N,P) > threshold(N) ) then
  noun attachment
elsif ( cooc(V,P) > threshold(V) ) then
  verb attachment
```

Now the problem arises on how to set the thresholds. It is obvious that the attachment decision gets more reliable the higher we set the thresholds. At the same time the number of decidable cases decreases. We aim to set the threshold in such a way that using this partial information is not worse than using the $cooc(N, P)$ and $cooc(V, P)$ values. We derive the noun threshold from the average of all noun cooccurrence values.

$$threshold(N) = \frac{\sum_{(N,P)} cooc(N,P)}{|cooc(N,P)|}$$

From our data we derive a sum of 1246.75 from 38,374 noun cooccurrence values leading to an average of 0.032. We use this as our noun threshold. In order to consequently employ the noun factor, the verb threshold is the product of the noun threshold and the noun factor.

$$threshold(V) = threshold(N) * noun\_factor$$

This follows from our assumption that the noun factor balances out an inherent difference between the noun and verb cooccurrence values. We thus work with a verb threshold of 0.136.

We only use the threshold for test cases with a missing cooccurrence value. Noun threshold comparison leads to 68 additional noun attachments out of which 55 are correct (an accuracy of 80.88%). Verb threshold comparison handles 123 additional verb attachments (92 correct) with an accuracy of 74.80%. This leads to a total of 4041 attachment decisions (90% attachment coverage).

	factor	correct	incorrect	accuracy	threshold
noun attachment	4.25	2089	425	83.09%	0.032
verb attachment		1075	452	70.40%	0.136
total		3164	877	78.30%	
decidable test cases		4041 (of 4469) coverage: 90.4%			

Table 4.14: Attachment accuracy for the $CZ_{shortlemma}$ test set using thresholds.

428 test cases remain undecidable. For 43 of these neither $cooc(N,P)$ nor $cooc(V,P)$ is known. For 98 test cases the value $cooc(N,P)$ is known but it is below the noun threshold, and for 287 test cases the value $cooc(V,P)$ is below the verb threshold.

Some of the approaches described in the literature have also used thresholds. [Ratnaparkhi 1998] uses the constant $\frac{1}{|\mathcal{P}|}$ where $\mathcal{P}$ is the set of possible prepositions. Since we work with a set of 100 prepositions and 20 contracted prepositions, this will amount to $1/120 = 0.0083$. If we use this value as noun threshold in our disambiguation algorithm, we increase the coverage to 94% but lose about 2% of attachment accuracy (77.04%).

The coverage increase based on threshold comparison is higher if the prior coverage level is lower. This can be seen from the evaluation against the NEGRA test set. The threshold employment leads to a 9% increase in coverage (see table 4.15).

We are content with the 90% attachment coverage for the CZ test set and we now try to increase the attachment accuracy by varying the computation of the cooccurrence values and by investigating the use of linguistic knowledge.

4.5 Sure Attachment and Possible Attachment

Our method for computing the cooccurrence values so far does not distinguish between ambiguously and non-ambiguously positioned PPs. E.g. a PP immediately following a personal

	factor	correct	incorrect	accuracy	threshold
noun attachment	4.25	1872	525	78.10%	0.032
verb attachment		1210	636	65.55%	0.136
total		3082	1161	72.64%	
decidable test cases		4243 (of 5803) coverage: 73%			

Table 4.15: Attachment accuracy for the NEGRA$_{shortlemma}$ test set using thresholds.

pronoun cannot be attached to a noun. It is very likely that this PP needs to be attached to the verb. Therefore, such a PP should result in a higher influence on $cooc(V, P)$ than a PP that is in an ambiguous position. ([Hindle and Rooth 1993] demonstrated the positive impact of this distinction for English.)

In order to account for such cases of sure attachment we need to identify all PP positions for sure noun attachment and sure verb attachment.

Sure verb attachment

In German a PP can be attached to a noun if it is right-adjacent to this noun. This means that all PPs following any other type of word can be considered a sure verb attachment.[8] In particular any sentence-initial PP can be considered a sure verb attachment (as in 4.6). Other examples are a PP following an adverb (as in 4.7) or a PP following a relative pronoun (as in 4.8).

(4.6) ***An EU-externe Länder*** *dürfen Daten nur exportiert werden, ...*

(4.7) *Es muß noch* ***vom EU-Ministerrat und dem Parlament*** *verabschiedet werden.*

(4.8) *..., die* ***ohne Änderungen*** *auf Windows- und Apple-PCs laufen.*

Sure noun attachment

Determining a sure noun attachment is more difficult. If a clause does not contain a full verb, as is the case with any copula sentence, a PP must be attached to the noun (or to an adjective).

(4.9) *Hintergrund dieses Kurseinbruchs ist die gedämpfte Gewinnerwartung* ***für 1995****.*

Furthermore, we find sure noun attachments in the sentence-initial constituent of a German assertive matrix clause. It is generally assumed that such clauses contain exactly one constituent in front of the finite verb (cf. [Zifonun et al. 1997] section E4 "Die Linearstruktur des Satzes" p. 1495). Therefore such clauses are called verb-second clauses in contrast to verb-first clauses (e.g. yes-no questions) and to verb-last clauses (subordinate clauses).

[8] We continue to disregard the small number of PPs that are attached to adjectives.

If, for instance, a sentence starts with an NP followed by a PP and the finite verb, the PP must be an integral part of the NP (as in example 4.10). Even if two PPs are part of a coordinated sentence-initial constituent (as in 4.11), these PPs will have to attach to the preceding nouns. They are not accessible for verb attachment.

(4.10) *Die Abkehr* ***von den proprietären Produkten*** *erzeugt mehr Wettbewerb ...*

(4.11) *Auch durch die weltweite Entrüstung* ***über diese Haltung*** *und mahnende Worte* ***von Branchen- und Börsenanalysten*** *ließ sich Firmenchef Andy Grove zunächst nicht beirren.*

In order to automatically identify the verb-second clauses in our training corpus we collect the sequence of constituents at the beginning of a sentence. The sequence ends with the finite verb or with a clause boundary (e.g. marking the boundary to a relative clause or some subordinate clause). If the finite verb or the subordinate clause marker are in sentence-initial position, the sequence is empty and we cannot determine a sure noun attachment in the first constituent position. In this way we mark 64,939 PPs as sure noun attachments in our training corpus.

When computing the frequencies, we will now distinguish between a PP that is a sure noun attachment, a PP that is a sure verb attachment, and one that is ambiguous. When computing the verb frequencies, a sure verb attachment counts as 1 point, a sure noun attachment as 0 points and an ambiguous PP as half a point. When computing the noun frequencies, we will count a sure noun attachment as 1 point and an ambiguous PP as half a point. Sure verb attachment PPs have not been counted for the noun + preposition frequencies in any of our experiments.

The improved precision in counting and the subsequent recomputation of the cooccurrence values lead to a new noun factor and a new threshold. The noun factor is higher since mostly the N+P pair counts have lost value. This also results in a lower noun threshold. We observe an increase in attachment accuracy from 78.30% to 80.54% (see table 4.16).

	factor	correct	incorrect	accuracy	threshold
noun attachment	5.48	2132	401	84.17%	0.020
verb attachment		1092	378	74.28%	0.109
total		3224	779	80.54%	
decidable test cases		4003 (of 4469) coverage: 89.6%			

Table 4.16: Attachment accuracy for the $CZ_{shortlemma}$ test set using sure attachments.

Incrementally applying "almost" sure attachment

In addition to insisting on sure attachments based on linguistic rules we can employ the cooccurrence values that we computed so far to find "almost" sure attachments. This corresponds to the incremental step in the [Hindle and Rooth 1993] experiments.

We determine the thresholds that lead to 95% correct attachments both for verbs and nouns. For verbs we find that a cooccurrence value above 0.4 leads to this accuracy and for nouns the threshold is at 0.05.

With these thresholds we redo the computation of the cooccurrence values. When we encouter a PP in an ambiguous position and its old cooccurrence value $cooc(V, P) > 0.4$, then $freq(V, P)$ is incremented by 1 and no count is made for the noun attachment frequency. If, on the other hand, the old noun value $cooc(N, P) > 0.05$, then $freq(N, P)$ is incremented by 1 and no count is made for the verb attachment frequency. If both thresholds apply or none of them, we give 0.5 to the frequency counts of both the verb and the noun (as before).

It turns out that only 3364 of the old N+P cooccurrence values (8.84%) and 243 V+P cooccurrence values (0.68%) are above these thresholds. In computing the new cooccurrence values this leads to the following distribution of the PP tokens in the one-verb clauses of the training corpus.

sure noun attachments	38,645
sure verb attachments	241,673
almost sure noun attachment	41,191
almost sure verb attachment	4,367
split due to ambiguity	146,495

Since a higher number of almost sure noun attachments (than of almost sure verb attachments) could be recognized, the value for the noun factor shifts back to 4.58, and based on the average noun cooccurrence value, the noun threshold increases to 0.024.

	factor	correct	incorrect	accuracy	threshold
noun attachment	4.58	2107	353	85.65%	0.024
verb attachment		1133	388	74.49%	0.109
total		3240	741	81.39%	
decidable test cases		3981 (of 4469) coverage: 89.1%			

Table 4.17: Attachment accuracy for the $CZ_{shortlemma}$ test set using almost sure and sure attachments.

We observe an improvement in the attachment accuracy from 80.54% to 81.39% (with a slight loss in the coverage due to the higher thresholds). We will now explore the use of some linguistic resources.

4.6 Idiomatic Usage of PPs

PPs are often part of collocational or idiomatic expressions. We distinguish three types of idiomatic usage with PPs involved.

1. **Frozen PPs** are PPs that function as a preposition. Many frozen PPs subcategorize for a PP with a special preposition *mit Hilfe* ***von****, im Vergleich* ***mit***. This subcategorization requirement can be exploited for annotating additional sure noun attachments in our training corpus.

2. **Support verb units** are combinations of a PP (or NP) and a semantically weak verb like *ans Werk gehen, auf der Kippe stehen.* These PPs must be counted as sure verb attachments when computing the cooccurrence values. The support verb units can also be used in the disambiguation step taking into account the core noun within the PP.

3. **General idioms** are all other idiomatic expressions, be they a complex noun phrase like *ein Wink mit dem Zaunpfahl* or a verb phrase like *zwei Fliegen mit einer Klappe schlagen.*

4.6.1 Using Frozen PPs and Support Verb Units

We used a list of 82 frozen PPs that have a prepositional subcategorization requirement to mark sure noun attachments in our training corpus. The list was obtained from [Schröder 1990] and extended as described in section 1.2.2. In addition we employed a list of 466 support verb units with PP + verb combinations to mark sure verb attachments in the training corpus.[9] With the help of these resources we were able to mark 3309 PPs as sure noun attachments and 7194 PPs as sure verb attachments in our training corpus.

Evaluating the new cooccurrence values against the CZ test set, it turns out that this move does not change the overall attachment accuracy (81.4%) nor the attachment coverage (89%). This may be due to the fact that the number of new sure noun PPs is too small to have an impact and that many of the sure verb PPs were counted as sure verb attachments before since they did not appear in an ambiguous position.

In another experiment we checked if the support verb units that occur in our test set were correctly disambiguated. In order to recognize them we compared the list of support verb units to the triple "verb + preposition + PP noun" (V, P, N_2). It is important that the core noun is used in its textual form, i.e. without lemmatization and decompounding. Some support verb units contain a plural noun (e.g. *zu Lasten gehen*) and will not be found if the noun is reduced to the singular base form. Nouns in support verb units are usually not compounds. So if a compound occurs as N_2 in the test, it should not be considered as a support verb unit. For example, the test quadruple (*bringt Zuwachs in Grössenordnung*) should not be considered as an instance of the support verb unit *in Ordnung bringen.* The disambiguation algorithm now works as follows:

```
if ( support_verb_unit(V,P,N2) ) then
  verb attachment
elsif ( cooc(N,P) && cooc(V,P) ) then
  if ( (cooc(N,P) * noun_factor) >= cooc(V,P) ) then
    noun attachment
  else
    verb attachment
elsif ( cooc(N,P) > threshold(N) ) then
  noun attachment
elsif ( cooc(V,P) > threshold(V) ) then
  verb attachment
```

[9] Thanks to Brigitte Krenn for making the list of support verb units available to us.

The CZ test set comprises 97 test cases with support verb units from our list. Before using the support verb units 90 of these test cases were correctly disambiguated as verb attachments, 5 were incorrectly treated as noun attachments and 2 were not decided. By using the support verb units as a knowledge source for disambiguation, we correctly predict verb attachment in all test cases. We thus increase the attachment accuracy to 81.52% (see table 4.18). The noun factor and the thresholds are kept from the previous experiment.

	factor	correct	incorrect	accuracy	threshold
noun attachment	4.58	2107	348	85.82%	0.024
verb attachment		1140	388	74.61%	0.109
total		3247	736	81.52%	
decidable test cases		3983 (of 4469) coverage: 89.1%			

Table 4.18: Attachment accuracy for the $CZ_{shortlemma}$ test set using support verb units.

4.6.2 Using Other Idioms

In order to investigate the impact of idiomatic usage on our disambiguation results we have extracted all idioms containing the preposition *mit* from a large collection of German idioms. After manually checking and cleaning these idioms, we have obtained 261 idioms for *mit* (228 involving a verb and 33 involving simply an NP or a PP). The idioms are structured very differently but the unifying criterion is that each idiom establishes a meaning that cannot be derived from the literal meanings of its parts. We have listed some examples in table 4.19.

German idiom	type of idiom	corresponding English term
mit Ach und Krach durchkommen	special PP + verb	*to scrape through*
mit Kanonen auf Spatzen schiessen	two special PPs + verb	*to break a fly on the wheel*
gemeinsame Sache machen mit jmd.	special NP + verb sub-categorizing for *mit*-PP	*to make common cause with someone*
das Kind mit dem Bade ausschütten	special NP + special PP + verb	*to throw the baby out with the bathwater*
sich mit Ruhm bekleckern	special PP + reflexive verb	*to cover oneself with glory*
ein Wink mit dem Zaunpfahl	complex NP including a special PP	*a broad hint*
mit Haken und Ösen	special PP	*with dirty tricks*

Table 4.19: Examples of idioms containing the preposition *mit*

We searched our corpus (which contains 49,277 sentences with the preposition *mit*) for all occurrences of *mit*-idioms. We collected 469 idiom tokens, 68 types.

idiom	$freq(idiom)$
mit sich bringen	123
Schritt halten mit etwas/jmd.	54
Geschäfte machen mit jmd.	41
mit von der Partie sein	23
mit auf den Weg geben	14
gemeinsame Sache machen mit jmd.	13
zwei Fliegen mit einer Klappe schlagen	12
sich Mühe geben mit etwas/jmd.	12
Ernst machen mit etwas	11
Nägel mit Köpfen machen	10
ins Gespräch kommen mit jmd.	10

Only these 11 idioms occurred 10 times or more. The most frequent idiom was *mit sich bringen*, but it is debatable whether to count it as an idiom or rather a special type of support verb unit. This unit must be taken into account when computing the cooccurrence values. Since the verb *bringen* cooccurs a total of 494 times with the preposition *mit* in our corpus (absolute frequency being 4443), the idiomatic usage does have a considerable impact on its cooccurrence value. The other idioms occur so rarely in our training corpus that they will not really change the cooccurrence values.

4.7 Deverbal and Regular Nouns

Deverbal nouns inherit their valency requirements in a weakened form from the respective verbs.[10] This is most evident if the verb takes a prepositional complement. We investigated the four most productive derivational suffixes that are used to create German nouns out of verbs: *-ation, -en, -e, -ung*. By far the most frequent of these is *-ung*. We disregarded the suffix *-er* since it serves to denote the person undergoing the activity, and we assume that such a person form does not preserve as many properties of the underlying verb as the noun denoting the process.

$W_{lem}P$	$freq(W_{lem}, P)$	$freq(W_{lem})$	$cooc(W_{lem}, P)$
eindringen in	66.0	106	0.62264
Eindringen in	6.5	24	0.27083
fragen nach	65.5	699	0.09371
Frage nach	94.5	2183	0.04329
kooperieren mit	135.0	432	0.31250
Kooperation mit	233.0	1252	0.18610
warnen vor	168.5	608	0.27714
Warnung vor	8.0	78	0.10256

This table shows that the cooccurrence values of the nouns are usually lower than the ones of their verbal counterparts.[11] Nouns do not bind their complements as strongly as verbs.

[10] This is also known for English. [Bowen 2001] found that for 411 nouns with PP complements 196 were derived nouns (with an overt derivational suffix) and 118 were 'linked' nouns (verb noun homographs).

[11] The frequency count for pairs is no longer an integer because of the split in counting ambiguous PPs.

But for all these nouns the listed cooccurrence value is the highest among all cooccuring prepositions.

The preposition *von* is special with deverbal nouns since it often denotes the subject or object of the underlying verb. In example 4.12 the *von*-PP is the logical subject of *Eindringen*. Because of this special status the cooccurrence value of the preposition *von* with a deverbal noun is often higher as with the underlying verb as can be seen in the following table.

$W_{lem}P$	$freq(W_{lem}, P)$	$freq(W_{lem})$	$cooc(W_{lem}, P)$
eindringen von	4.5	106	0.04245
Eindringen von	4.5	24	0.18750
vermeiden von	14.5	351	0.04131
Vermeidung von	24.0	75	0.32000

(4.12) *Spezielle Werkstoffe verhindern andererseits das Eindringen* ***von elektromagnetischen Wellen****.*

(4.13) *Die Vermeidung* ***von Direkt- und Reflexblendung*** *durch Tages- und Kunstlicht sollte dabei im Vordergrund stehen.*

The preposition *durch* marks the subject if *von* marks the object (as in example 4.13), but is often omitted and thus does not have an impact on our computations.

We see the following options to apply these dependencies to the improvement of noun cooccurrence values and also to increase the attachment coverage.

- We may strengthen the cooccurrence values for deverbal nouns with low frequencies based on the cooccurrence values of the underlying verbs.
- We may generate the cooccurrence values for deverbal nouns that were unseen during training if there exist cooccurrence values for the underlying verbs.

4.7.1 Strengthening the Cooccurrence Values of Deverbal Nouns

In the CZ test set we find 1170 test cases with the reference noun ending in a deverbal suffix (*-ation* (128), *-e* (390), *-en* (89), *-ung* (563)).[12] For 1078 of these test cases we have computed $cooc(N, P)$ from our training data. For 400 of these we have also computed the corresponding $cooc(V, P)$ with V being the base verb of N. The number of verb cooccurrence values is so much lower since many of the nouns with suffix *-e* do not have corresponding verbs (e.g. *Höhe *höhen; Seite *seiten; Experte *experten*). This holds also for few of the other nouns with deverbal suffix (e.g. *Neuerung *neuern*).

We then checked how many of these test cases correspond to V+P pairs with P being part of a prepositional requirement for the verb. We thus searched all V+P pairs in the CELEX database. Only 89 test cases passed this test. This means that only for 89 test cases we may use $cooc(V, P)$ to support the corresponding value $cooc(N_V, P)$.[13] The following table shows two of these test cases. In the first case the pair *Umstellung + auf* could be supported by *umstellen + auf*, and in the second example *Beteiligung + an* could be supported by *beteiligen + an*.

[12] The reference nouns in the CZ test set comprise a total of 47 different nominal suffixes, only 15 of which are deverbal suffixes according to [Hoeppner 1980].

[13] We write $cooc(N_V, P)$ to denote the cooccurrence value of a deverbal noun.

verb	head noun	prep.	core of PP	PP function
erfordern	*Währungs#umstellung*	*auf*	*Euro*	noun modifier
veräußern	*Omnitel-Beteiligung*	*an*	*Mannesmann*	noun modifier

Before we tried to use the verbal support for the deverbal nouns, we checked how many of the 89 test cases (76 noun attachments and 13 verb attachments) were correctly resolved. It turned out that the deverbal nouns "can speak for themselves". They do not need the support of the underlying verbs. 83 of the test cases (93.3%) are correctly attached. Five of the 6 errors are incorrect noun attachments, meaning that we would have to reduce $cooc(N, P)$. So, $cooc(V, P)$ will be of no use. One of them (4.14) is a truly ambiguous example that can be resolved only with deep world knowledge.

(4.14) *Zudem planen die Italiener, 8 Prozent ihrer Omnitel-Beteiligung* ***an Mannesmann*** *zu veräußern.*

The overall picture with deverbal nouns is that they inherit enough complement requirements from their underlying verbs that they collect good enough cooccurrence values in the training. Transfering cooccurrence information from verbs to their deverbal nouns will not contribute to an improved disambiguation result.

4.7.2 Generating a Cooccurrence Value for Unseen Deverbal Nouns

As stated above, there are 92 test cases with deverbal nouns for which we do not have $cooc(N_V, P)$. This may be due to a low frequency ($<= 10$) of the deverbal noun or to the non-existence of the N+P pair in the training data. But, if we have the corresponding $cooc(V, P)$ and if the verb requires the preposition as a complement, we may carry over the cooccurrence value to the deverbal noun.

Five out of these 92 test cases fall into this class. Example 4.15 leads to a sixtuple with reference noun *Brüten* and preposition *über*. But the noun *Brüten* does not occur in our training data. The corresponding verb *brüten* occurs 17 times and scores 6 points in cooccurrence with *über* resulting in (*cooc(brüten,über) = 0.35294*). Furthermore, *brüten* is listed in CELEX as requiring a prepositional object with *über* plus dative. We therefore transfer the cooccurrence value to the noun *Brüten* and correctly predict noun attachment for the *über*-PP. This transfer works correctly for all five applicable test cases.

(4.15) *Ich merkte auch, daß mir die Zusammenarbeit mit Menschen mehr Spaß machte als ... das Brüten* ***über Programmcodes****.*

In addition, there are seven out of the 92 test cases with the preposition *von*. As we have seen, the cooccurrence of a deverbal noun with this preposition usually requires the attachment of the *von*-PP to the noun. And again this holds true for all seven test cases.

In example 4.16 the deverbal noun *Ablegen* with preposition *von* does not provide a cooccurrence value since it occurs only 9 times in our training data and thus falls short of the minimal frequency threshold. But the information that this is a deverbal noun and the special preposition *von* gives enough evidence for a noun attachment decision.

(4.16) *Die Maschine verfügt über 64 CPUs ... für das Ablegen* ***von Szenen*** *...*

4.8 Reflexive Verbs

So far, we have neglected the difference between reflexive and non-reflexive verb usage. But this distinction is useful since in German most reflexive verbs also have a non-reflexive reading, and these readings often differ in their subcategorization requirement. E.g. the verb *sorgen* has a non-reflexive reading with a strict requirement for the preposition *für*, and it has a reflexive reading which calls for *um*.

In shallow corpus analysis the distinction between a reflexive versus a non-reflexive verb reading can be based on the occurrence of a reflexive pronoun within the same clause. Unfortunately, German reflexive pronouns for the first and second person (*mich, dich, uns, euch*) are homographic with their non-reflexive counterparts. But the third person reflexive pronoun (*sich*), which is by far the most frequent in technical texts, can serve to unambiguously identify reflexive verb usage.

In order to account for this distinction we extend the computation in the training procedure. While counting verbs and verb + preposition pairs (cf. step 1 in section 4.3.2), we also search for the reflexive pronoun *sich* within the same clause. We ignore *sich* if it occurs immediately after a preposition (cf. *mit sich bringen; für/in sich haben*) since this does not constitute a reflexive reading of the verb.

Around 6% of all one-verb clauses in the training corpus contain the reflexive pronoun *sich*. For the computation of the cooccurrence values we store the reflexive pronoun with the verb. We thus get more verb lemma types (now 9178) and more verb preposition pairs (now 47,725) than before. In the training data we count 1493 verb types with a reflexive pronoun. The most frequent ones are:

verb V	$freq(V)$
sich handeln	1540
sich befinden	1151
sich entwickeln	976
sich konzentrieren	933
sich finden	856
sich ergeben	816
sich eignen	804
sich machen	733
sich entscheiden	722
sich zeigen	716

In addition we count 6772 reflexive verb + preposition pairs and we compute 4861 cooccurrence values. The highest cooccurrence values are:

verb V	P	$freq(V,P)$	$freq(V)$	$cooc(V,P)$
sich gliedern	*in*	31.0	37	0.83784
sich herumschlagen	*mit*	24.0	31	0.77419
sich einfügen	*in*	16.5	23	0.71739
sich ausruhen	*auf*	11.0	16	0.68750
sich widerspiegeln	*in*	43.0	63	0.68254
sich schmücken	*mit*	9.5	14	0.67857
sich vertragen	*mit*	10.0	15	0.66667
sich niederlassen	*in*	9.0	14	0.64286
sich integrieren	*in*	7.0	11	0.63636
sich beziehen	*auf*	130.0	206	0.63107

We can now distinguish between *sich sorgen* and *sorgen*. From our statistics we see the difference in the cooccurrence preference. *sorgen + für* and *sich sorgen + um* have high cooccurrence values while the values for *sorgen + um* and *sich sorgen + für* are orders of magnitude lower.

verb V_{lem}	P	$freq(V_{lem},P)$	$freq(V_{lem})$	$cooc(V_{lem},P)$
sorgen	*für*	1064.5	2648	0.40200
sorgen	*um*	6.5	2648	0.00245
sich sorgen	*für*	1.5	31	0.04839
sich sorgen	*um*	13.5	31	0.43548

But, surprisingly, the evaluation of the cooccurrence values with the reflexive pronoun distinction does not show any improvements in the attachment precision when applied against the CZ test set. It stays at 81.5%. At the same time, the number of attachments decreases slightly.

Only 381 out of the 4469 CZ test cases (8.5%) contain reflexive verbs. 335 of the reflexive test cases were decided prior to the distinction between reflexive and non-reflexive verb readings. Only 67 of these test cases (20%) were incorrectly decided.

If we take the reflexive reading distinction into account, the picture does not change much. 307 reflexive test cases were decided. Some could no longer be decided since the frequency of the reflexive verb was below our minimum frequency threshold of 10. Still, 63 of the reflexive test cases (20.5%) are incorrectly decided.

How can this surprising behavior be explained? If one verb reading (reflexive or non-reflexive) dominates the frequency count of this verb, its cooccurrence value will not change much by counting the readings separately. Consider the non-reflexive reading of *sorgen* or the reflexive reading of the verb *einigen* in the following table.

	verb V_{lem}	P	$freq(V_{lem},P)$	$freq(V_{lem})$	$cooc(V_{lem},P)$
prior count	*sorgen*	*für*	1066.0	2679	0.39791
separate count	*sorgen*	*für*	1064.5	2648	0.40200
prior count	*(sich) einigen*	*auf*	128.5	356	0.36096
separate count	*sich einigen*	*auf*	123.5	337	0.36647
prior count	*(sich) sorgen*	*um*	20.0	2679	0.00747
separate count	*sich sorgen*	*um*	13.5	31	0.43548

The separate counting of reflexives does have a strong impact only on the cooccurrence values of rare readings (such as the reflexive reading of *sorgen*). But these rare readings will only account for a minor fraction of the test cases since the test cases are randomly selected and thus reflect the frequency distribution of verb readings.

In addition, there are a number of verbs that have the same preposition requirements in both their reflexive and non-reflexive readings (*sich/jmd. beteiligen an, sich/jmd. interessieren für*). A separate counting will have no impact.

Moreover, in our evaluation we have not distinguished between true reflexive verbs (like *sich kümmern*) and the reflexive usage of otherwise non-reflexive verbs. We may extract this information from the CELEX database. Following [Wahrig 1978], CELEX distinguishes between

- obligatory and optional reflexivity (*sich solidarisieren* vs. *sich waschen*),
- accusative and dative object reflexivity (*sich solidarisieren* vs. *sich überlegen*),
- true reflexivity and reciprocal reflexivity (*sich solidarisieren* vs. *sich überschneiden*).

As stated in section 4.1.1 CELEX contains 1758 verbs annotated with at least one reflexivity class. The reflexivity distribution is shown in the following table.

	obligatory	optional
dative object with true reflexivity	191	301
accusative object with true reflexivity	1592	691
dative object with reciprocal reflexivity	8	112
accusative object with reciprocal reflexivity	40	262
total	1831	1366

This means, for instance, that 1592 verb readings are annotated as requiring a reflexive accusative object. The same verb can have different readings manifested as different subcat frames requiring some sort of reflexivity. As an example consider the subcat requirements for the verb *klemmen* listed in the following table.

subcat requirements	reflexivity	preposition
no object required Ex: *Die Tür klemmt.*		
accusative object and dative object Ex: *Ich habe mir den Finger geklemmt.*	dative object	
accusative object and prepositional object Ex: *Ich klemme mich hinter die Aufgabe.*	accusative object	*hinter* + acc.
accusative object and location Ex: *Ich habe das Blatt an die Tür geklemmt.* Ex: *Ich habe mir das Blatt an die Tür geklemmt.*	optional dative obj.	*an* + acc.

From the 340 reflexive-verb test cases in the CZ test set only 50 test cases are sanctioned by CELEX as both reflexive and requiring the preposition. Eleven of these test cases seem rather unusual reflexive cases of the verb *finden* with the prepositions *in* or *zu*. Obviously, all

verbs with multiple subcategorization frames may lead to an incorrect choice of the CELEX subcat frame.

Of these 50 test cases 48 can be decided and lead to 79% attachment accuracy. The remaining 2 do not have a verb cooccurrence value. They are typical cases of rare reflexive verbs or rare reflexive readings such as *sich ergötzen an, sich bringen aus (Schußlinie)*. These two cases could be resolved by applying the CELEX information. Of the 10 incorrectly resolved cases four involve the verb *finden*.

Interestingly, CELEX does not provide any reflexive information for 16 verbs which occur with a reflexive pronoun in the CZ test cases (in 29 instances). Six of these verbs are not listed in CELEX at all: *einloggen, einwählen, heraussuchen, herunterladen, vervierfachen, zusammenschließen*. Out of these *einloggen, einwählen* and *herunterladen* are specific terms in computer science, the other three verbs are serious omissions.

Because of these CELEX limitations it might be worthwhile to consider other collections of reflexive verbs. We are aware of two lists compiled by [Griesbach and Uhlig 1994] and [Mater 1969], but we did not have access to them in a machine readable format. In particular Mater's list is very comprehensive with 525 verbs that have to be reflexive, 4640 verbs that can be reflexive, and a complementary list of 9388 verbs that cannot be used reflexively.

This section shows that using reflexive pronouns in the computation of the cooccurrence values does not significantly improve the overall PP disambiguation accuracy although it does help in single cases. It seems that we will have to use a deeper analysis of the complements to differentiate more precisely between verb readings.

4.9 Local and Temporal PPs

In our training corpus we automatically identified local and temporal PPs (cf. section 3.1.6). We suspected that these PPs would most often attach to the verb as has been reported for English. But German constituent order results in a different tendency for adjunct PPs.

[Griesbach 1986] gives a detailed account of this order. He starts with the usual division into *Vorfeld, Mittelfeld* and *Nachfeld* in which the fields are separated by the verbal elements. The *Vorfeld* can be occupied by at most one constituent, the *Nachfeld* is often empty. The most important is the *Mittelfeld*. [Griesbach 1986] identifies 12 positions in the *Mittelfeld*.[14] Positions 1 through 7 constitute the *Kontaktbereich* and positions 8 through 12 constitute the *Informationsbereich*. The *Kontaktbereich* is filled with elements that are presumably known to the hearer. In contrast, the *Informationsbereich* takes elements with new information that the speaker wants the hearer to alert to. We do not want to repeat all of Griesbach's arguments for all 12 positions. We will briefly summarize the main ideas and focus on the positions for the PPs in this scheme.

Kontaktbereich					*Informationsbereich*		
1 | 2 | 3	4	5	6	7	8 | 9 | 10	11	12
pronouns	subj	adjunct PP	acc obj	dat obj	moveables	PP obj	pred compl

Since pronouns are typically pointing back to known objects in the discourse, they occupy positions 1 through 3. Position 4 is occupied by the subject if it is not in the *Vorfeld*. Position 5 can be occupied by free adjuncts and is thus the first possible position for PPs. This means

[14] [Griesbach 1986] uses the term *Satzfeld* instead of the more common *Mittelfeld*.

that local and temporal PPs functioning as modifiers are positioned in front of any accusative object (position 6) or dative object (position 7). A PP in position 5 will only be ambiguous if position 4 is indeed filled with the subject.

Within the *Informationsbereich* the positions 8, 9 and 10 are not specifically assigned. If elements from the *Kontaktbereich* are taken over to the *Informationsbereich*, they will occupy these slots in the same order as in the *Kontaktbereich*. Position 11 will be occupied by a prepositional object and position 12 by a predicative complement (*Prädikatsergänzung*).

Of course, hardly ever will all these positions be filled in one particular sentence. They should rather be taken as indicators for the relative order of the constituents. For the PP attachment task we gather the following tendencies. Free adjunct PPs will often be positioned in front of the objects, which results in a smaller number of ambiguously positioned PPs than in English. However, a prepositional complement of the verb will rather be positioned at the very end of the *Mittelfeld* (in a position that is prone to noun vs. verb attachment ambiguity). We will now look at local and temporal PPs in turn.

4.9.1 Local PPs

If a local PP modifies the verb, it often follows the finite verb immediately (as in 4.17). In this position the PP is not ambiguous. If a local PP occurs in an ambiguous position as in 4.18, and if it is followed by an object (here the accusative object *mehrere Prototypen*), then it mostly attaches to the preceding noun rather than the verb. Example 4.19 shows a local PP following a temporal adverb.

(4.17) *Drahtlose Kommunikation wird* ***in den USA*** *bald das Bild bestimmen.*

(4.18) *Bereits zur Halbzeit erwartet die japanische Regierung aus dem neuen Forschungszentrum* ***in der Wissenschaftsstadt Tsukuba*** *mehrere Prototypen mit 10,000 Prozessoren.*

(4.19) *Hans-Rudi Koch erklärte offensiv, er wolle zukünftig* ***in Deutschland*** *in die Sprachkommunikation einsteigen.*

We checked one annual volume of our corpus for the positions of the automatically recognized local and temporal PPs. We had recognized 7052 local PPs and 8525 temporal PPs. We then checked for the tokens immediately preceding the PPs (cf. table 4.20).

We observe that 399 PPs that were automatically annotated as local PPs were clause-initial. This means they were either positioned at the beginning of a sentence or adjacent to a clause boundary marker or to a clause-initiating conjunction. In 472 cases the local PP was immediately preceded by a finite verb (auxiliary, modal or full verb). These are the cases with the local PPs as free adjuncts in position 5 according to the Griesbach scheme. 166 local PPs are preceded by a personal or reflexive pronoun which will also make them typical adjuncts in position 5. 838 local PPs follow some sort of particle such as adverb, negation particle, indefinite or demonstrative pronoun. These PPs cannot attach to a noun and will thus also account for verb attachments. Still, this leaves the surprising number of 4230 local PPs (60%) in the ambiguous position following a noun.

Our manually annotated test corpora did not contain information on the semantic classification of PPs. We therefore ran our program for the recognition of local and temporal

PPs over these sentences. We mapped the local and temporal tags to our extracted test cases (i.e. to the sixtuples). This allowed us to check the attachment decision for all local PPs in both the CZ and the NEGRA test sets (see table 4.21). The results from both test sets are surprisingly consistent: 71% of the local PPs are noun attachments and 29% are verb attachments.

positions of local and temporal PPs	freq(local PP)		freq(temporal PP)	
clause-initial PP	399	5.7%	1027	12.0%
finite verb precedes PP	472	6.7%	1314	15.4%
personal or reflexive pronoun precedes PP	166	2.4%	350	4.1%
particle (adverb, pronoun) precedes PP	838	11.9%	1697	19.9%
noun precedes PP (ambiguous position)	4230	60.0%	2547	29.9%
determiner precedes PP (adjective attachment)	134	1.9%	228	2.7%
miscellaneous	813	11.5%	1362	16.0%
total	7052	100%	8525	100%

temporal PPs preceding local PP	106 (1.2%)
local PPs preceding temporal PP	39 (0.5%)

Table 4.20: Positions of local and temporal PPs in the 1993 volume of the CZ corpus

4.9.2 Temporal PPs

In principle, temporal PPs will occur in the same positions as local PPs. Example 4.20 shows a subordinate (verb final) sentence with the temporal PP following the subject. The attachment of the PP is debatable; it is one of the cases in which both verb attachment and noun attachment result in the same overall meaning of the sentence.

Examples 4.21 and 4.22 demonstrate a precedence for temporal over local PPs. This corresponds to the order "temporal < causal < modal < local" given by [Griesbach 1986]. and also to [Helbig and Buscha 1998] who state that the order of two free adjuncts is weakly constrained by "(temporal, causal) < (modal, local)". Corpus statistics confirm this tendency. In the 1993 CZ corpus we find 106 temporal PPs immediately preceding a local PP but only 39 cases in the reverse order. In relation to the number of all annotated local and temporal PPs in the corpus, temporal precedence is about twice as frequent.

(4.20) *... während der Börsenkurs* ***zu diesem Zeitpunkt*** *bei nur 1349 Lire lag.*

(4.21) *... die sich* ***unmittelbar nach der Wende in Ostdeutschland*** *engagiert haben.*

(4.22) *... und wird diese gemeinsam mit dem neuen Partner Interop* ***im nächsten Jahr im Juni in Berlin*** *organisieren.*

(4.23) *Falls die positiven Gewinnprognosen* ***für die Jahre 1994 und 1995*** *zutreffen ...*

We analysed the automatically recognized temporal PPs in the same manner as the local PPs (cf. table 4.20). It is striking that temporal PPs occur less frequently in the ambiguous position behind a noun. This is probably due to the fact that temporal PPs describe the duration or point in time of an activity and will thus rather attach to the verb.

As for the local PPs, we also checked the attachment decisions in our test sets for the temporal PPs. Again the results are consistent across corpora (see table 4.21). 45-46% of the temporal PPs are noun attachments and 54-55% are verb attachments.

	local PPs		temporal PPs	
corpus	N attach	V attach	N attach	V attach
CZ test set	158 (71%)	66 (29%)	83 (45%)	103 (55%)
NEGRA test set	263 (71%)	106 (29%)	152 (46%)	181 (54%)

Table 4.21: Number of local and temporal PPs in the test sets

4.9.3 Using Attachment Tendencies in the Training

We will therefore employ these general attachment tendencies in the computation of the cooccurrence values. When computing the "frequencies" we will distribute the values accordingly. This is a supervised aspect in our unsupervised method. The attachment tendencies for local and temporal PPs were determined from the manually disambiguated material.

1. A sure noun-attached PP is counted as 1 for $freq(N, P)$ and 0 for $freq(V, P)$.
2. A sure verb-attached PP is counted as 0 for $freq(N, P)$ and 1 for $freq(V, P)$.
3. An ambiguously positioned **local** PP is counted as 0.7 for $freq(N, P)$ and 0.3 for $freq(V, P)$.
4. An ambiguously positioned **temporal** PP is counted as 0.45 for $freq(N, P)$ and 0.55 for $freq(V, P)$.
5. All other ambiguously positioned PPs are counted as 0.5 for $freq(N, P)$ and 0.5 for $freq(V, P)$.

The results are summarized in table 4.22. The noun factor is now at 5.4 since the local PPs shift weight to the N+P frequencies. Unfortunately, this weighting of the frequencies for temporal and local PPs results in a decrease in attachment accuracy.

	factor	correct	incorrect	accuracy	threshold
noun attachment	5.4	2151	408	84.06%	0.020
verb attachment		1091	370	74.67%	0.108
total		3242	778	80.65%	
decidable test cases		4020 (of 4469) coverage: 90.0%			

Table 4.22: Attachment accuracy for the $CZ_{shortlemma}$ test set using local and temporal weights.

4.9.4 Using Attachment Tendencies in the Disambiguation Algorithm

In addition to using the attachment tendencies of local and temporal PPs in training, we may also include them in our disambiguation algorithm. Dependent on these tendencies and on the fact that temporal and local PPs are often adjuncts rather than complements, we found that the attachment accuracy for local and temporal PPs is clearly below the average attachment accuracy (the coverage is at 88% for both).

	accuracy of local PPs	accuracy of temporal PPs
noun attachment	81.69%	59.80%
verb attachment	60.00%	79.03%
total	75.63%	67.07%

Table 4.23: Attachment accuracy for local and temporal PPs

In particular the attachment accuracy of temporal PPs is very low and strongly biased towards verb attachment. Such a bias can be leveled out via our noun factor. We modify the noun factor in the disambiguation algorithm in the following manner: We eliminate the general attachment bias from the noun factor and replace it with the specific attachment bias of local and temporal PPs.

The general attachment bias for the CZ test set is 61 / 39 based on the initial count that there are 61% noun attached test cases and 39% verb attached cases. The specific attachment bias for temporal and local PPs is derived from the figures in table 4.21. That means that the noun factor is adapted for the local and temporal PP test cases according to the following formulae:

$$noun\_factor(local) = \frac{noun\_factor}{\frac{61}{39}} * \frac{71}{29}$$

$$noun\_factor(temporal) = \frac{noun\_factor}{\frac{61}{39}} * \frac{45}{55}$$

Keeping the general noun factor of 5.4 will set the noun factor for local PPs to 8.45 and the noun factor for temporal PPs to 2.82. The verb threshold is dynamically adapted in accordance with the respective noun factor.

Adapting the values in this way leads to more evenly distributed values between noun and verb attachment accuracies and to an improvement of 2% in the accuracy of the temporal test cases (now at 69%). Since local and temporal weights in the training did not lead to an improvement, we leave them and continue with the prior training data. The accuracy for the local PP test cases stays the same (75.41%). Overall we observe a slight improvement in the accuracy (see table 4.24).

4.10 Pronominal Adverbs

Pronominal adverbs (*daran, dabei, ..., dazu*) are abbreviations for PPs and function as cataphoric or anaphoric pointers (mostly) to PP complements. In section 1.1.2 we introduced

	factor	correct	incorrect	accuracy	threshold
noun attachment	4.58	2117	347	85.92%	0.024
verb attachment		1147	388	74.72%	0.109
total		3264	735	81.62%	
decidable test cases		3999 (of 4469) coverage: 89.5%			

Table 4.24: Attachment accuracy for the $CZ_{shortlemma}$ test set using an adaptive noun factor for local and temporal PPs.

them in detail. Here, we will only provide two example sentences that exemplify the pronominal adverb *dafür* in ambiguous positions, with a noun attachment in 4.24 and with a verb attachment in 4.25.

(4.24) *Die folgenden beiden Programme mögen als sinnvolle Beispiele* ***dafür*** *gelten.*

(4.25) *Wer den Schritt ... nicht schon vollzogen hat, muß sich spätestens in diesem Kontext* ***dafür*** *entscheiden.*

As we mentioned in table 3.1 the NEGRA test set contains 111 test cases with pronominal adverbs and the CZ test set contains 41 such test cases. Since we had noted in the introduction (section 1.1.2) that the frequency distributions of prepositions and their pronominal adverb counterparts are different, these test cases have been ignored in our evaluations so far.

We checked the pronominal adverb test cases from both test sets using the cooccurrence values computed from the prepositions. The results are summarized in table 4.25.

	factor	CZ set accuracy	NEGRA set accuracy	threshold
noun attachment	4.58	30.00%	77.27%	0.024
verb attachment		90.00%	97.87%	0.109
total		70.00%	91.30%	
decidable test cases		coverage: 73%	coverage: 63%	

Table 4.25: Attachment accuracy for pronominal adverbs.

Although the test samples are very small, there is a clear tendency that the noun factor is not necessary for the pronominal adverbs. Most of them attach to the verb (83% in the CZ test set and 80% in the NEGRA test set) and therefore the accuracy of verb attachment is very high. We reran the evaluation without the noun factor and achieve accuracy values (81.25% for the CZ test cases and 83.75% for the NEGRA test cases) that are not much better than a default attachment to the verb. Since the coverage is also rather low (78% for the CZ test cases and 73% for the NEGRA test cases), we might as well assign verb attachment to all pronominal adverbs without considering the cooccurrence values.

In a final experiment we changed our training procedure. Instead of counting prepositions for the computation of the cooccurrence values, we only counted pronominal adverbs. All

pronominal adverbs were clustered via their respective prepositions. For example, *dar-aus, hier-aus* and *wor-aus* were all counted as the same pronominal adverb. The following table shows the nouns with the highest cooccurrence values with pronominal adverbs. It is striking that idiomatic usages account for some of these pairs: *(k)einen Hehl daraus machen; ein Schelm, wer Böses dabei denkt; ein Lied davon singen.*

noun N_1	$PronAdv$	$freq(N_1, PronAdv)$	$freq(N_1)$	$cooc(N_1, PronAdv)$
Hehl	*dar-aus*	6.0	18	0.33333
Gewähr	*da-für*	2.5	16	0.15625
Indiz	*da-für*	10.0	88	0.11364
Exempel	*da-für*	1.0	12	0.08333
Böse	*da-bei*	1.0	12	0.08333
Anzeichen	*da-für*	5.5	70	0.07857
Schuld	*dar-an*	10.5	155	0.06774
Lied	*da-von*	2.0	30	0.06667
Garant	*da-für*	1.5	23	0.06522
Aufschluß	*dar-über*	2.5	39	0.06410

The following table shows the highest cooccurrence values of verbs and pronominal adverbs. The examples with *da-mit: anspielen da-mit; abfinden da-mit* are not intuitive cases. This may be due to the fact that *damit* often is not used as pronominal adverb but rather functions as conjunction to introduce purpose clauses (*Finalsätze*).

verb V	$PronAdv$	$freq(V, PronAdv)$	$freq(V)$	$cooc(V, PronAdv)$
hinwegtäuschen	*dar-über*	23.5	37	0.63514
anspielen	*da-mit*	8.0	15	0.53333
ausgehen	*da-von*	396.0	777	0.50965
hindeuten	*dar-auf*	32.0	69	0.46377
hinweisen	*dar-auf*	102.5	227	0.45154
gesellen	*da-zu*	9.0	23	0.39130
zweifeln	*dar-an*	10.0	26	0.38462
handele	*da-bei*	10.0	26	0.38462
abfinden	*da-mit*	8.0	21	0.38095
verführen	*da-zu*	4.0	11	0.36364

For the evaluation of the pronominal adverb cases we lowered the noun threshold to the average noun cooccurrence value which is now at 0.004. The results are summarized in table 4.26.

	factor	CZ set accuracy	NEGRA set accuracy	threshold
noun attachment	1	50.00%	87.50%	0.004
verb attachment		88.89%	87.87%	0.004
total		83.87%	87.84%	
decidable test cases		coverage: 76%	coverage: 68%	

Table 4.26: Attachment accuracy for pronominal adverbs trained on pronominal adverbs.

Surprisingly, training on pronominal adverbs shows a clear improvement for the attachment of pronominal adverbs compared to training on prepositions. Although the number of training instances is much smaller than for the prepositions, the accuracy is higher (at about the same coverage level). This is clear evidence for the separate treatment of pronominal adverb attachment and prepositional attachment. It should be noted, however, that the bulk of the attachments is based on comparisons against the verb threshold. For the CZ test cases 10 out of 30 attachments are based on the verb threshold and for the NEGRA test cases 45 out of 74 attachments.

4.11 Comparative Phrases

In section 1.2.1 we introduced comparative phrases as borderline cases of PPs. Although we extracted those cases from the treebanks and added them to our test sets, we left them aside in the above evaluations. We recall that the CZ test set contains 48 test cases with comparative phrases and the NEGRA test set contains 145 such test cases (cf. table 3.1 on page 86).

Similar to the pronominal adverbs the comparative phrases have a much stronger tendency to attach to the verb than to the noun. In the CZ test set 66% of the comparative phrases are marked as verb attachments and in the NEGRA set 75% are verb attachments. Using the cooccurrence values obtained from training over the prepositions or over the pronominal adverbs will not help for the comparative phrase attachment since the comparative particles *als, wie* are not tagged as prepositions and therefore are not included in the cooccurrence sets.

The two comparative particles in German are homographic with conjunctions and interrogative pronouns. The following table shows the distribution of the PoS tags for these words in our corpus.

PoS tag	function	particle *als*	particle *wie*
KOKOM	comparative particle	24,511	11,600
KON	coordinating conjunction	526	107
KOUS	subordinating conjunction	1,307	5,085
PWAV	interrogative pronoun	0	750
total		26,344	17,542

Even if we consider that there is a certain error rate in these tags, the table shows that the overwhelming majority of usage for both words is as comparative particle.

We computed the cooccurrence values for all verbs and nouns with respect to the two comparative particles (when they were tagged as such). For the verbs we got clear cooccurrence values. The following table shows the top ten cooccurrence values for *als* and the top 2 for *wie*. This confirms the observation that a number of verbs take comparative phrases as complements. CELEX lists 36 verbs as requiring an "equivalence" phrase with *als*. Among the CELEX verbs are *fungieren, empfinden, auffassen, bezeichnen, werten* and *ansehen*.

verb V	$Particle$	$freq(V, Particle)$	$freq(V)$	$cooc(V, Particle)$
fungieren	*als*	202.0	234	0.86325
entpuppen	*als*	35.0	45	0.77778
abtun	*als*	14.0	18	0.77778
erweisen	*als*	232.0	315	0.73651
empfinden	*als*	38.5	56	0.68750
auffassen	*als*	7.5	11	0.68182
bezeichnen	*als*	339.0	505	0.67129
werten	*als*	80.5	127	0.63386
einstufen	*als*	61.0	101	0.60396
ansehen	*als*	151.5	259	0.58494
...				...
anmuten	*wie*	6.0	18	0.33333
dastehen	*wie*	3.0	11	0.27273

For the nouns the cooccurrence list is rather blurred even on the top. It starts off with cooccurrence values that are an order of magnitude lower than the top verb values. Second ranked is the noun *Einstufung* which is a deverbal noun based on *einstufen* which is in the top ten of the verb table. It seems that the cooccurrence of a noun with a comparative phrase is rather coincidental and therefore no clear cooccurrence values emerge.

noun N	$Particle$	$freq(N, Particle)$	$freq(N)$	$cooc(N, Particle)$
Gehör	*wie*	1.5	18	0.08333
Einstufung	*als*	1.0	12	0.08333
Reputation	*als*	1.0	14	0.07143
Philosoph	*wie*	1.0	14	0.07143
Vermarkter	*wie*	2.0	29	0.06897

We thus expect that the attachment accuracy for verbs is good but the accuracy for noun attachment rather bad. We derive a noun factor of 13.5 and a threshold of 0.0085. The evaluation results are summarized in table 4.27.

	factor	CZ set accuracy	NEGRA set accuracy	threshold
noun attachment	13.5	75.00%	54.17%	0.0085
verb attachment		92.31%	93.33%	0.1147
total		86.84%	82.14%	
decidable test cases		coverage: 98%	coverage: 58%	

Table 4.27: Attachment accuracy for comparative phrases

The result for the CZ test set must be cautiously interpreted. The test base is very small (48 instances). The results for the NEGRA test set give a more realistic picture. The accuracy score is about 4% above the default attachment base line since 75% of the NEGRA test cases are verb attachments. A good heuristic for the attachment of comparative phrases is the attachment to the verb unless there is strong evidence for noun attachment.

4.12 Using Pair and Triple Frequencies

So far we have used bigram frequencies over word pairs, (V, P) and (N, P), to compute the cooccurrence values. Some of the previous research (e.g. [Collins and Brooks 1995] and [Pantel and Lin 2000]) has shown that it is advantageous to include the noun from within the PP in the calculation. But moving from pair frequencies to triple frequencies will increase the sparse data problem. Therefore we will compute the pair frequencies and triple frequencies in parallel and use a cascaded disambiguation algorithm to exploit the triple cooccurrence values and the pair cooccurrence values in sequence.

But first we have to tackle the task of finding the noun within the PP in the training procedure. In analogy to chapter 2, we will call this noun N_2 and label the reference noun as N_1. Starting from a preposition, the training algorithm searches the PP which was annotated by our NP/PP chunker (cf. section 3.1.5). It accepts the lemma of the first noun within the PP as N_2. Compound nouns are reduced to their last element. Nouns that are semantically classified are represented by their semantic tag (⟨company⟩, ⟨person⟩, ⟨location⟩, ⟨time⟩). We list some extraction examples in the following table.

PP in training corpus	extracted P	extracted N_2
gegenüber ihrem Vorläufer	*gegenüber*	*Vorläufer*
von Vorträgen oder Vorführungen	*von*	*Vortrag*
in der PC- und Workstation-Technologie	*in*	*Technologie*
hinter einem traditionellen Zeitungslayout	*hinter*	*Layout*
von Ploenzke-Maintenance-Spezialist Thomas Engel	*von*	*Spezialist*
bis zehn Jahre	*bis*	⟨time⟩
von De Benedetti	*von*	⟨person⟩

If the PP chunker could not recognize a PP (because of its internal complexity), or if the PP does not contain a noun (but rather an adverb or pronoun), then no triple frequency is computed.

In analogy to the pair cooccurrence value, the triple cooccurrence value is computed as:

$$cooc(W, P, N_2) = freq(W)/freq(W, P, N_2) \quad with \quad W \in \{V, N_1\}$$

The following table shows a selection of the 20 highest cooccurrence triples (N_1, P, N_2) and some examples of triples with a semantic tag as N_1. The table includes

- a person name with the middle element *vom*. This name was missed by the proper name recognizer since usually the middle element is *von*. Such preposition-like name elements are annotated as proper name parts and are thus not tagged as prepositions.
- a city name like *Eching bei München* that was missed by the geographical name recognizer.
- parts of idioms: *die Spreu vom Weizen trennen; die Klinke in die Hand geben; wie das Pfeifen im Walde*
- a technical collocation: *Umdrehungen pro Minute*

- part of an organization name: *Forum InformatikerInnen für Frieden und gesellschaftliche Verantwortung*
- a mistagged noun: *Made in Germany.*[15]

The others show interesting generalizations. The noun *Sitz* is frequently followed by a locative *in*-PP[16] and *Nachfolge von* is typically followed by a person. A company is often mentioned with its location (varying with the prepositions *in* and *aus*) and a person with his or her company affiliation.

noun N_1	P	noun N_2	$freq(N_1, P, N_2)$	$freq(V)$	$cooc(N_1, P, N_2)$
Gerd	*von*	*Hövel*	11.0	18	0.61111
Pilz	*aus*	⟨location⟩	8.4	19	0.44211
Spreu	*von*	*Weizen*	7.0	16	0.43750
InformatikerIn	*für*	*Frieden*	6.0	14	0.42857
Klinke	*in*	*Hand*	4.5	11	0.40909
Sitz	*in*	⟨location⟩	134.9	418	0.32273
Nachfolge	*von*	⟨person⟩	27.5	86	0.31977
Made	*in*	*Germany*	7.5	24	0.31250
Pfeifen	*in*	*Wald*	3.0	11	0.27273
Quartier	*in*	⟨location⟩	15.2	58	0.26207
Umdrehung	*pro*	*Minute*	7.0	28	0.25000
Zusammenschalten	*von*	*Netz*	2.5	11	0.22727
Eching	*bei*	⟨location⟩	2.8	13	0.21538
...					...
⟨company⟩	*in*	⟨location⟩	783.00	126,733	0.00618
⟨person⟩	*von*	⟨company⟩	244.45	46,261	0.00528
⟨company⟩	*aus*	⟨location⟩	256.20	126,733	0.00202
⟨person⟩	*von*	*Institut*	48.00	46,261	0.00104

In the same manner we computed the triple frequencies for (V, P, N_2). The following table shows the highest ranked cooccurrence values for such triples. Again metaphorical and idiomatic usage accounts for most of these examples: *auf Lorbeeren ausruhen; sich mit Ruhm bekleckern; auf einen Zug aufspringen; aus der Taufe heben.* But there is also a technical collocation (*mit Megahertz takten*).

[15] Although *Made* is a German noun meaning *mite*.
[16] One could argue that *mit Sitz in* is a frozen PP.

verb V	P	noun N_2	$freq(V, P, N_2)$	$freq(V)$	$cooc(V, P, N_2)$
paktieren	*mit*	⟨company⟩	13.0	19	0.68421
ausruhen	*auf*	*Lorbeer*	11.0	18	0.61111
bekleckern	*mit*	*Ruhm*	6.0	11	0.54545
aufspringen	*auf*	*Zug*	39.0	74	0.52703
takten	*mit*	*Megahertz*	54.0	112	0.48214
rufen	*in*	*Leben*	130.0	282	0.46099
abfassen	*in*	*Sprache*	5.0	11	0.45455
heben	*aus*	*Taufe*	62.5	151	0.41391
krönen	*von*	*Erfolg*	5.5	14	0.39286
hüllen	*in*	*Schweigen*	9.0	24	0.37500
datieren	*aus*	⟨time⟩	6.0	17	0.35294
umtaufen	*in*	⟨company⟩	4.0	12	0.33333
hineinkommen	*in*	*Markt*	4.0	12	0.33333
beheimaten	*in*	⟨location⟩	7.3	22	0.33182
terminieren	*auf*	⟨time⟩	9.6	29	0.32931

If a triple (V, P, N_2) has a high cooccurrence value and there are other triples (V, P, N_2') with the same verb and preposition but differing N_2 and low cooccurrence values, then this is a good indicator for a synonymy of N_2 and N_2'. The two examples in the following table support this observation. *Lorbeer* is synonymously used for *Erfolg*, and *Frequenz* is the hyperonym of *Megahertz*, while *Megaherz* contains a spelling mistake, and *MHz* is the corresponding abbreviation.

verb V	P	noun N_2	$freq(V, P, N_2)$	$freq(V)$	$cooc(V, P, N_2)$
ausruhen	*auf*	*Erfolg*	1.0	18	0.05556
ausruhen	*auf*	*Lorbeer*	11.0	18	0.61111
takten	*mit*	*Frequenz*	4.0	112	0.03571
takten	*mit*	*Megahertz*	54.0	112	0.48214
takten	*mit*	*Megaherz*	1.0	112	0.00893
takten	*mit*	*MHz*	2.0	112	0.01786

With this kind of triple frequency computation we collected 150,379 (N_1, P, N_2) noun cooccurrence values (compared to 38,103 (N_1, P) pair values) and 233,170 (V, P, N_2) verb cooccurrence values (compared to 35,836 (V, P) pair values). We integrated these triple cooccurrence values into the disambiguation algorithm. If both $cooc(N_1, P, N_2)$ and $cooc(V, P, N_2)$ exist for a given test case, then the higher value decides the attachment.

```
if ( support_verb_unit(V,P,N2) ) then
  verb attachment
elsif ( cooc(N1,P,N2) && cooc(V,P,N2) ) then
  if ( (cooc(N1,P,N2) * noun_factor) >= cooc(V,P,N2) ) then
    noun attachment
  else
    verb attachment
elsif ( cooc(N1,P) && cooc(V,P) ) then
  if ( (cooc(N1,P) * noun_factor) >= cooc(V,P) ) then
```

```
     noun attachment
  else
     verb attachment
elsif ( cooc(N1,P) > threshold(N) ) then
  noun attachment
elsif ( cooc(V,P) > threshold(V) ) then
   verb attachment
```

The noun factors for triple comparison and pair comparison are computed separately. The noun factor for pairs is 5.47 and for triples 5.97.

	factor	correct	incorrect	accuracy	threshold
noun attachment	5.47; 5.97	2213	424	83.92%	0.020
verb attachment		1077	314	77.43%	0.109
total		3290	738	81.67%	
decidable test cases		4028 (of 4469) coverage: 90.13%			

Table 4.28: Attachment accuracy for the $CZ_{shortlemma}$ test set using triple comparisons.

The attachment accuracy is improved to 81.67% by the integration of the triple cooccurrence values. A split on the decision levels reveals that triple comparison is 4.41% better than pair comparison.

decision level	number of cases	accuracy
support verb units	97	100.00%
triple comparison	953	84.36%
pair comparison	2813	79.95%
$cooc(N_1, P) >$ threshold	74	85.13%
$cooc(V, P) >$ threshold	91	84.61%
total	4028	81.67%

Overall the attachment accuracies of noun attachment and verb attachment are almost balanced. This balance also holds on the triple comparison level. With a noun factor of 5.97 it results in 85.42% correct noun attachments and 80.97% correct verb attachments. On the pair level we observe 83.22% correct noun attachments and 73.72% correct verb attachments. The 84.36% for triple comparison demonstrates what we can expect if we enlarge our corpus and consequently increase the percentage of test cases that can be disambiguated based on triple cooccurrence values.

This finding is confirmed by evaluating the training data against the NEGRA test set. Only 205 of the NEGRA test cases are disambiguated within the triple comparison. We then observe an attachment accuracy of 78% for the triple comparison level which is about 4% higher than the accuracy for the pair comparison.

4.13 Using GermaNet

In section 4.4.4 we clustered the training tokens by using lemmas instead of inflected words. We extended the clustering by mapping all recognized proper names to one of the keyword tags ⟨company⟩, ⟨person⟩, or ⟨location⟩. The intention was to combine the frequency counts for all members of a class and thus to increase the attachment coverage.

This idea can be extended by using a thesaurus to cluster synonyms. For instance, we may combine the frequency counts of *Gespräch* and *Interview* or of *Konferenz, Kongress* and *Tagung*. Some of the research described in section 2.2 used WordNet to cluster English nouns and verbs (e.g. [Stetina and Nagao 1997, Li and Abe 1998]).

WordNet[17] is an on-line thesaurus for English that is structured to resemble the human lexical memory. Due to its broad lexical coverage and its free availability it has become one of the best-known and most-used thesauri. It organizes English nouns, verbs, adjectives and adverbs into hierarchical synonym sets (synsets), each synset standing for one lexical concept. WordNet (version 1.6) uses 66,025 noun synsets, 12,127 verb synsets, 17,915 adjective synsets and 3575 adverb synsets; the total number of senses is around 170,000. The roots of the synset hierarchy are a small number of generic concepts, whereas each concept is a unique beginner of a separate hierarchy. The individual synsets are linked by different relations. WordNet relations for nouns are antonymy (e.g. *top* vs. *bottom*), hyponymy (*maple* vs. *tree*), hypernymy (*plant* - *tree*), meronymy (*arm* - *body*) and holonymy (*body* - *arm*), for verbs we find relations such as antonymy (*rise* - *ascend*), hypernymy (*walk* - *limp*), entailment (*snore* - *sleep*) and troponymy (*limp* - *walk*). Synsets for verbs additionally contain verb frames to describe their subcategorization requirements.

No such large-scale thesaurus is available for German. But recently, a smaller thesaurus called GermaNet has been compiled at the University of Tübingen. It was built following the ideas and the format of WordNet. We will use the GermaNet synsets to cluster the nouns in our training corpus.

GermaNet[18] is a thesaurus for German with a structure similar to WordNet. It is based on a corpus with words taken - among others - from the CELEX lexical database and from several lists of lemmatized words gathered from newspaper texts (e.g. Frankfurter Rundschau). Our version of GermaNet includes 20,260 noun synsets, 7,214 verb synsets and 1,999 adjective synsets; it totally covers around 80,000 German words. The basic division of the database into the four word classes noun, adjective, verb and adverb is the same as in WordNet, although the analysis of adverbs is currently not implemented. GermaNet works with the same lexical relations as defined in WordNet with few exceptions such as changes in the frequency of their individual use. The main difference to WordNet is that GermaNet works with lemmas (as a consequence morphological processing is needed) and allows cross-classification of the relations between synsets. Cross-classification allows a more world-knowledge based hierarchy but needs restrictions to avoid incorrect inheritage.

If each word belonged to exactly one synonym class in GermaNet, the clustering task would be easy. One could simply substitute every word of this class by a class identifier. In fact, 5347 GermaNet tokens belong to exactly one synonym class.[19] This may seem like a substantial number. A closer look reveals that 942 of these tokens are feminine forms that

[17] For WordNet see `www.cogsci.princeton.edu/~wn/`.

[18] For GermaNet see `www.sfs.nphil.uni-tuebingen.de/lsd/`.

[19] We only consider synsets with more than one member, since we are only interested in synonyms.

are in a synonym relation with their masculine counterparts.

- *Ecuadorianerin, Ecuadorianer*
- *Bäckerin, Bäcker*
- *Angestellte, Angestellter*

Another 133 tokens are homonyms, they belong to two or more synonym classes. For example the word *Zug* belongs to the following three synonym classes in GermaNet.[20]

- *Zug: Eisenbahnzug*
- *Zug: Charaktereigenschaft, Charakterzug*
- *Zug: Zugkraft*

A precise mapping of such ambiguous words to a specific synonym class requires word sense disambiguation based on the context of the word or on the topic of the document. This is a complex task and outside the realm of this research.

Therefore we work with the simplifying assumption that every word occurs evenly frequent in all its synonym classes. During the training phase a word's frequency count will be distributed over all its synonym classes. If the word *Zug* occurs in the training corpus, the frequency count of all three of its synonym classes will be incremented.

In the evaluation phase we map every reference noun N_1 to its synonym classes. In case of multiple classes we have to decide which synonym class to use for the disambiguation algorithm. We select the synonym class with the highest cooccurrence value. The following table shows a nice example in which this heuristic leads to the correct disambiguation. The noun *Kunde* is a member of three GermaNet synonym classes corresponding to the meanings *knowledge, message*, and *customer*. In our previous experiments these meanings were conflated although grammatical gender could have been used to distinguish between *die Kunde* (sense 1 or 2) and *der Kunde* (sense 3). Previously, the cooccurrence value for *(Kunde, über)* was 0.00374. But since the two other members of the sense 1 class *(Wissen, Kenntnis)* contribute higher cooccurrence values for the preposition *über*, sense 1 results in the highest cooccurrence value. This corresponds to our linguistic intuitions.

noun N_1	P	$freq(N_1, P, N_2)$	$freq(N_1)$	$cooc(N_1, P, N_2)$
Kunde	*über*	23.20	6203	0.00374
Wissen	*über*	30.85	753	0.04097
Kenntnis	*über*	14.00	530	0.02642
nouns N_1 in class	P	$freq(class, P, N_2)$	$freq(class)$	$cooc(class, P, N_2)$
Kunde, Wissen, Kenntnis	*über*	68.05	7486	0.00909
Kunde, Botschaft	*über*	24.70	6366	0.00388
Kunde, Kundin	*über*	23.20	6210	0.00374

[20] In Switzerland *Zug* is also the name of a city and a canton.

At the same time this approach leads to considerably lower cooccurrence values for *(Wissen, über)* and *(Kenntnis, über)*, which could mean that they are now too low for correct attachment decisions. It would have been cleaner to distinguish between *die Kunde* and *der Kunde* from the beginning so that only the attachment tendency of the feminine noun would impact the synonym class.

Suprisingly, using GermaNet in this way has no positive impact on attachment accuracy and coverage. Evaluating over the CZ test set results in 81.59% accuracy and 90.5% coverage, although 1125 nouns in the test set were substituted by their synonym class.

Why is this so? What results can we expect from using GermaNet? A side effect like the disambiguation of *Kunde* in the above example is rare. Rather, we had expected an increase in the attachment accuracy based on higher frequencies of word classes compared to single words. But this is not necessarily true. Consider the cooccurrence values for *Tagung, Konferenz* and *Kongreß* in the following table.

noun N_1	P	$freq(N_1, P, N_2)$	$freq(N_1)$	$cooc(N_1, P, N_2)$
Kongreß	*in*	30.80	459	0.06710
Tagung	*in*	29.20	311	0.09389
Konferenz	*in*	81.65	921	0.08865
Kongreß	*zu*	8.50	459	0.01852
Tagung	*zu*	3.50	311	0.01125
Konferenz	*zu*	16.50	921	0.01792
Kongreß	*für*	11.50	459	0.02505
Tagung	*für*	2.00	311	0.00643
Konferenz	*für*	12.45	921	0.01352
nouns N_1 in class	P	$freq(class, P, N_2)$	$freq(class)$	$cooc(class, P, N_2)$
Konferenz, Tagung, Kongreß	*in*	141.65	1691	0.08377
Konferenz, Tagung, Kongreß	*zu*	28.50	1691	0.01685
Konferenz, Tagung, Kongreß	*für*	25.95	1691	0.01535

The cooccurrence values with respect to the prepositions *in* and *zu* are very similar for the three words. Such a similar behavior is evidence for mapping the three words to the same class. But then the cooccurrence value of the class is the same as that of any of the words and will not entail any difference in the disambiguation process.

If any of the members of a synonym class shows idiosyncratic behavior with respect to a given preposition (like *Tagung* does with *für*), this speciality is averaged out and may cause incorrect attachments for the idiosyncratic N+P combination. Consequently, we cannot expect to see an improvement in the attachment accuracy.

On the other hand we had at least expected an increase in coverage. If a noun N_1 has a corpus frequency below the minimal frequency threshold (set to 10), no cooccurrence value will be computed for this noun. But if this noun is a member of a synonym class, it may get its value from the class cooccurrence value if the combined frequency of all its members is above the threshold. One high frequency member suffices to provide a cooccurrence value for all members of the class. The following table shows the changes of the disambiguation results that are due to GermaNet.

	without GermaNet		with GermaNet	
decision level	number of cases	accuracy	number of cases	accuracy
support verb unit	97	100.00%	97	100.00%
triple comparison	953	84.36%	960	84.48%
pair comparison	2813	79.95%	2821	79.79%
$cooc(N_1, P) >$ threshold	74	85.13%	73	84.93%
$cooc(V, P) >$ threshold	91	84.61%	85	84.71%
total	4028	81.67%	4036	81.59%

Although 7 additional test cases are now handled at the triple comparison level and 8 additional at the pair comparison level, the overall accuracy is slightly lower than before. But since the differences are very small, we cannot draw a definite conclusion.

There are two main reasons for the additionally decided test cases. First, a noun occured frequently but still did not cooccur with the preposition in the training corpus. For example, *(Termin, nach)* did not cooccur in that corpus, although *Termin* occured 405 times. In GermaNet this noun is synonym with *Frist* and that noun cooccured once with the preposition *nach*. Therefore the combined scores of *Termin* and *Frist* lead to a (low) cooccurrence value.

The second reason is that a noun occured 10 times or less and the combined score lifts it over this threshold. The noun *Herrscher* occured exactly 10 times in the CZ training corpus and was thus eliminated by the minimal frequency threshold. But the feminine form *Herrscherin* occured once and this leads to the combined frequency of 11 and the computation of the cooccurrence value for *(Herrscher, über)* as 0.09091. This is a borderline case which shows that working with thresholds easily eliminates useful information and that therefore clustering the words is important.

4.14 Conclusions from the Cooccurrence Experiments

We have shown that cooccurrence statistics can be used to resolve PP attachment ambiguities. We started off with 71.4% attachment accuracy and 57% resolved cases when counting the word forms in our training corpus. We then introduced a noun factor to work against the bias of verb attachment. The distinction between sure and possible attachments, the use of a list of support verb units, and the use of the core noun within the PP (cooccurrence values over triples) lead to the largest improvement in accuracy. In parallel, we have shown that various clustering techniques can be used to increase the coverage. As a best result we have reported on 81.67% attachment accuracy and 90.1% coverage. Table 4.29 summarizes the results of the experiments with the CZ test set.

We have focused on the percentage of correct attachments for the decidable cases. But if we want to employ our method in a natural language processing system, we need to decide all cases. If we solve the remaining test cases with a default for noun attachment, we get the overall results as shown in table 4.30. Default noun attachment is only 56% accurate for the remaining 429 cases and reduces the overall attachment accuracy to 79.14%.

It is therefore imperative to increase the coverage before default attachment in order to keep the number of default decisions low. So what were the remaining unresolved cases that had to be subjected to default attachment? From the 429 unresolved cases there were 91 with a $cooc(N, P)$-value below the threshold (i.e. there is no $cooc(V, P)$-value), 293 with a $cooc(V, P)$-value below the threshold (and no $cooc(N, P)$-value), and 45 test cases with neither

	noun factor	accuracy	coverage	threshold(N)
word forms	/	71.40%	57%	/
word forms	4.25	81.31%	57%	/
incl. real att. nouns	4.25	80.43%	57%	/
(long) lemmas	4.25	78.23%	72%	/
short lemmas	4.25	78.21%	83%	/
proper names	4.25	78.36%	86%	/
threshold	4.25	78.30%	90.4%	0.032
sure/possible att.	5.48	80.54%	89.6%	0.020
almost sure att.	4.58	81.39%	89.1%	0.024
support verb units	4.58	81.52%	89.1%	0.024
local/temporal PPs	5.40	80.65%	90.0%	0.020
triple cooc. values	5.47/5.97	81.67%	90.1%	0.020
GermaNet synonyms	5.47/5.97	81.59%	90.3%	0.020
pronominal adverbs	1.0	83.87%	76%	0.004
comparative phrases	13.5	86.84%	98%	0.0085

Table 4.29: Overview of the experiments with cooccurrence values and the CZ test set

	factor	correct	incorrect	accuracy	threshold
noun attachment	5.47; 5.97	2464	619	79.92%	0.020
verb attachment		1073	313	77.42%	0.109
total		3537	932	79.14%	
decidable test cases		4469 (of 4469) coverage: 100%			

Table 4.30: Attachment accuracy for the $CZ_{shortlemma}$ test set with names, noun factor, thresholds, triple comparisons, GermaNet and defaults.

a verb nor a noun cooccurrence value. So, the missing noun cooccurrence values account for the majority of the undecided cases.

Analysis of the undecided cases

Noun cooccurrence values are missing for many reasons. The remaining CZ test cases contain

1. a number of complex names that were not recognized as belonging to any of the proper name classes in the corpus preparation step and thus were not counted as names in the training phase. These include
   - organization names (*Rensselaer Polytechnic Institute, Roybal Center, University of Alabama*), and
   - product names (*Baan IV, Internet Explorer, Universal Server*).

2. "unusual" nouns such as *Menetekel, Radikalinskis*.
3. misspelled nouns such as *Meagbit, Verbinung*.
4. foreign words (e.g. *Componentware, Firewall, Multithreating* (sic!)).
5. (few) lemmatization problems. For example, the noun *Namen* can be lemmatized as both *Namen* and *Name*. Through our lemma filter it was mapped to the former. But the form *Name* was lemmatized as the latter and thus the lemma counts were split.
6. compounds that could not be segmented due to unknown or foreign first elements (*Gigaoperationen, Migrationspfad*).
7. rare prepositions (*außerhalb, hinsichtlich, samt*). In addition it turned out that the preposition *via* was systematically mistagged as an adjective.

In comparison, few verb cooccurrence values are missing. Some are missing because of rare prepositions and the others because of rare verbs or special verbal compounds (*gutmachen, herauspressen, koproduzieren*) and one English verb that has been Germanized (*clustern*).

Analysis of the incorrectly attached cases

First, we checked whether there are prepositions that are especially bad in attachment in relation to their occurrence frequency. We checked this for all prepositions that occured more than 50 times in the CZ test set. The following table lists the number of occurrences and the percentage for all test cases and for the incorrectly attached cases. For example, the preposition *von* occured in 793 test cases which corresponds to 17.74% of all test cases. It is incorrectly attached in 69 cases which corresponds to 9.29% of the incorrectly attached test cases.

	overall		incorrectly attached	
preposition P	$freq(P)$	percentage	$freq(P)$	percentage
in	895	20.0269	249	33.5128
von	793	17.7445	69	9.2867
für	539	12.0609	86	11.5747
mit	387	8.6597	67	9.0175
zu	369	8.2569	46	6.1911
auf	357	7.9884	61	8.2100
bei	153	3.4236	37	4.9798
an	151	3.3788	24	3.2301
über	135	3.0208	14	1.8843
aus	106	2.3719	12	1.6151
um	84	1.8796	14	1.8843
unter	64	1.4321	15	2.0188
nach	64	1.4321	11	1.4805
zwischen	52	1.1636	4	0.5384

It is most obvious that *in* is a difficult preposition for the PP attachment task. It is far overrepresented in the incorrectly attached cases in comparison to its share of the test

cases. In contrast, *von* is an easy preposition. The most frequent prepositions that are always correctly attached are *per* (35 times), *gegen* (17), and *seit* (14).

Second, we checked whether the incorrect attachments correlate with low frequencies of the involved nouns and verbs. The cooccurrence value totally disregards the absolute frequencies. The only restriction is that the nouns and verbs occur more than 10 times. But there is no difference in confidence dependent on whether the noun occured 11 times or 11,000 times. We therefore tested whether the attachment accuracy improves if we increase the threshold.

frequency threshold	accuracy	coverage
$freq(W) > 10$	81.59%	90.3%
$freq(W) > 50$	81.77%	87.1%
$freq(W) > 100$	81.77%	83.1%
$freq(W) > 200$	81.57%	77.0%
$freq(W) > 400$	80.73%	66.5%

Surprisingly, a higher unigram frequency of verbs and nouns does not provide for a higher attachment accuracy. It naturally lowers the coverage since there are less cooccurrence values available, but the accuracy is almost constant.

So, incorrect attachments are not due to low frequencies but to contradictory evidence or small distances between cooccurrence values. We computed the distances between comparison values for all incorrectly attached test cases.

- For triples we computed the distance between $(cooc(N_1, P, N_2) * triple\_noun\_factor)$ and $cooc(V, P, N_2)$.
- For pairs we computed the distance between $(cooc(N_1, P) * noun\_factor)$ and $cooc(V, P)$.
- For threshold comparison we computed the distance between $cooc(W, P)$ and the respective threshold.

The following table shows the number of incorrect attachments and the average distances for the various decision levels. It is striking that the average distances for the incorrect noun attachment cases are bigger than for the verb attachment errors both for triple and pair comparisons. This is due to the influence of the noun factors.

decision level	type	number of cases	accuracy	average difference
triple comparison	wrong noun att.	107	85.42%	0.02369
	wrong verb att.	42	81.41%	0.00459
pair comparison	wrong noun att.	312	83.07%	0.11242
	wrong verb att.	258	73.62%	0.05938
$cooc(N_1, P) >$ threshold	wrong noun att.	11	84.93%	0.01307
$cooc(V, P) >$ threshold	wrong verb att.	13	84.71%	0.07804

After sorting the wrong attached cases with decreasing distances, it turned out that the three topmost test cases (those with highest distances) were based on clear errors in the manual attachment decision. The fourth was an incorrect noun attachment for example 4.26. The attachment decision for this example was based on $cooc(Zugang, zu) = 1.770$ and

cooc(offerieren,zu) = 0.044 which led to a clear prediction of noun attachment. It is a nice example for both noun and verb binding the same preposition. Indeed, the noun *Zugang* has a high attachment tendency with *zu*. But the noun within the PP overrides this tendency and makes it a verb attachment.

(4.26) *..., die den Internet-Zugang* ***zu einem Festpreis*** *offerieren.*

(4.27) *..., die aufmerksamen Zeitgeistern beim Surfen* ***ins Auge*** *springen.*

(4.28) *..., mit allen nationalen und internationalen Carriern* ***mit Aktivitäten*** *in München zusammenzuarbeiten.*

(4.29) *Zu ihr gehört ein schneller Compiler* ***zur Übersetzung*** *des Java-Programmcodes.*

(4.30) *Alle Module sind mit dem VN-Bus* ***mit einer Kapazität*** *von 400 Megabit pro Sekunde bestückt.*

Example 4.27 shows another incorrect noun attachment. It exemplifies that it is important to recognize idiomatic prepositional objects (*ins Auge springen*) so that they can be attached to the verb without using the cooccurrence values.

In example 4.28 the PP was incorrectly attached to the verb since *zusammenarbeiten* has a strong tendency to bind a *mit*-PP ($cooc(V, P) = 0.35819$). This test case was resolved by comparing the verb cooccurrence value against the verb threshold. There was no cooccurrence value for *(Carriern,mit)* since the noun could not be lemmatized and this particular noun form did not cooccur with that preposition in the training corpus.

The *zu*-PP in example 4.29 is incorrectly attached to the verb since *gehören* has a strong tendency to bind such a PP. But this requirement is satisfied by the other *zu*-PP in sentence-initial position. This shows the limitation of basing the attachment decision only on the quadruple (V, N_1, P, N_2). If the wider sentence context were used, this type of error could be avoided.

Finally, example 4.30 is also incorrectly verb-attached based on pair comparison. It stands for those cases that can only be correctly resolved with a detailed knowledge of the subject domain.

At the other end of the spectrum there are incorrectly attached cases with a very narrow distance between the cooccurrence values. Naturally, we find a number of examples in this range that show no clear attachment preference even for the human. The *für*-PP in example 4.31 was attached to the noun by the human annotator but attached to the verb by the system. The system based its decision on triple comparison of a very narrow margin ($cooc(N_1, P, N_2) = 0.00107$ including the noun factor; and $cooc(V, P, N_2) = 0.00196$). In fact, this is an example of an indeterminate PP which does not alter the sentence meaning no matter how it is attached.

(4.31) *Sie entwickeln jetzt schwerpunktmäßig Produkte* ***für Businesskunden****.*

We have shown how we can exploit the information from the annotated CZ training corpus to compute pair and triple cooccurrence values. We used different clustering techniques to increase the coverage of the test cases. Evaluating against the CZ test set and the NEGRA test set we have noticed a 5% better accuracy for the former. We will now move on to another training corpus in order to determine the influence of the training texts on the results.

Chapter 5

Evaluation across Corpora

In order to check the influence of the training corpus on the cooccurrence values, we performed a second set of experiments using a different newspaper corpus. In chapter 4 we had used four annual volumes of the Computer-Zeitung (CZ) to obtain frequency counts for nouns, verbs, their bigrams with prepositions, and the triples that included the PP noun.

For comparison we will now use the Neue Zürcher Zeitung (NZZ). It is a daily newspaper aiming at educated readers. It is very text oriented with few photos. We have access to four monthly volumes of the NZZ from 1994. In contrast to the Computer-Zeitung, the NZZ texts are annotated with XML tags for document structure (meta-information on date, author and page; titles on different levels and text blocks).

⟨DOC⟩ ⟨DOCID⟩ ak10.004 ⟨/DOCID⟩
⟨KURZTEXT⟩Basken/Taktik⟨/KURZTEXT⟩
⟨DATUM⟩10.01.94 ⟨/DATUM⟩
⟨AUTOR⟩BA⟨/AUTOR⟩
⟨PAGE⟩ 3 ⟨/PAGE⟩ ⟨AUSGABE_NR⟩ 7 ⟨/AUSGABE_NR⟩
⟨MAIN_TITLE⟩ Neue Parteiengespräche im Baskenland ⟨/MAIN_TITLE⟩
⟨MAIN_TITLE⟩ Geänderte Taktik Madrids gegenüber ETA? ⟨/MAIN_TITLE⟩
⟨DATE_INFO⟩ B. A. Madrid, 9. Januar ⟨/DATE_INFO⟩
⟨TEXT⟩ Unter den Politikern des spanischen Baskenlandes ist eine Polemik ausgebrochen, die sich um die möglichen Folgen dreht, die eine änderung der Taktik haben könnte, welche die Zentralregierung in Madrid in ihrem Kampf gegen die Terrororganisation ETA anwendet. Laut den Berichten verschiedener Zeitungen am Wochenende hat die spanische Regierung der Partei Herri Batasuna (HB) indirekt Gespräche angeboten, falls diese legale Organisation von ETA die Terroristen zu einer - vorerst befristeten - Einstellung der Gewaltakte zu bringen vermöchte. Sollten diese Meldungen zutreffen, liessen sie eine Wende in der Haltung der Madrider Regierung erkennen. Bisher zielte deren Politik darauf, HB zu isolieren und zu umgehen; die Absicht bestand darin, zu gesprächswilligen ETA-Mitgliedern direkt einen Kanal offenzuhalten, falls diese je bereit sein sollten, eine Abkehr der Organisation von gewalttätigen Methoden zu erreichen. ⟨/TEXT⟩
⟨SECTION_TITLE⟩ Ein neuer Innenminister ⟨/SECTION_TITLE⟩
⟨TEXT⟩ Die sich in Umrissen abzeichnende ... ⟨/TEXT⟩
⟨/DOC⟩

We delete the administrative information on date, author and pages but we keep the doc tags, text tags and title tags. This results in the following token counts in column 2 (before document removal).

month	number of tokens before doc. removal	number of tokens after doc. removal
January 94	1,795,133	1,682,297
April 94	1,855,840	1,744,048
May 94	1,810,544	1,695,846
June 94	2,144,699	1,695,846
total	7,606,216	7,152,873

This means that a monthly volume of the NZZ contains about 10% more tokens than an annual volume of the Computer-Zeitung. However, the NZZ count includes the remaining XML tags and also "noisy" tokens such as sports results (3:1 in football; 2:09,81 min in downhill ski racing; 214,2 m in ski jumping etc.) and chess moves (2. Sg1_f3 d7_d6). The newspaper also contains the Radio and TV programme (including titles in French and Italian) as well as listings of events such as church services and rock concerts.

We therefore checked for typical headers of such articles (*Fussball, Schach, Wetterbericht* etc.) and removed these articles. We took care to remove only those articles that contain tables and listings rather than running text. The removal procedure eliminated between 350 and 460 articles per month (resulting in the reduced token counts in column 3 above).

Articles in the NZZ are on average 25.7 sentences long (including document headers; standard deviation 37.3) while the average sentence length is 17.1 words (including punctuation symbols but excluding XML tags; standard deviation 14.7). The CZ corpus, for comparison, has an average article length of 20.1 sentences and an average sentence length of 15.7 words.

In addition, we had to account for the fact that our PoS tagger had been trained over Standard-German newspaper texts but the NZZ texts are in Swiss-German. Written Swiss-German differs most notably from Standard-German in that it does not use *'ß'* but rather *'ss'*.[1] Due to this difference the tagger systematically mistagged the conjunction *dass* which was spelled *daß* in Standard-German. We made sure that the Swiss variant was annotated with the correct PoS tag before the text was processed by the tagger.

Swiss-German differs not only in spelling rules but also in the vocabulary (cf. [Meyer 1989]). Among the differences are the Swiss-German prepositions *innert* and *ennet*. The former roughly corresponds to the Standard-German preposition *innerhalb* but it is semantically more restricted. *innert* is used almost exclusively for temporal PPs (see examples 5.1 and 5.2) whereas *innerhalb* can also introduce local PPs. And while *innerhalb* can be followed either by a genitive NP or a *von*-PP, *innert* governs mostly genitive NPs (and rarely dative NPs). Since *innert* is a frequent preposition in Swiss-German (417 occurrences in our NZZ corpus), we made sure that it is annotated with the PoS tag for prepositions.

(5.1) *... dass durch Änderung des Verkehrsplans die Voraussetzungen für die nötigen Bewilligungen* ***innert kurzer Zeit*** *geschaffen werden könnten.*

(5.2) *Damit soll die Arbeitslosenquote* ***innert sechs Jahren*** *auf 5% gedrückt werden.*

The Swiss-German preposition *ennet* is less frequently used. It translates into Standard-German as *jenseits* or *ausserhalb*.

[1] This difference has become less severe after the German spelling reform of the late 90s. The tagger training material and all our German corpora date prior to the reform and adhere to the old spelling rules.

(5.3) *Noch kurz zuvor hatten der ehemalige DDR-Skispringer Hans-Georg Aschenbach und die Schwimmerin Christiane Knacke* ***ennet des Stacheldrahts*** *mit vorgeblichen Enthüllungen "Pferd und Reiter" genannt.*

5.1 Cooccurrence Values for Lemmas

Except for the above modifications we processed the NZZ corpus in the same way as the Computer-Zeitung corpus. That means we used our modules for proper name recognition (person, location and company names), for PoS tagging, for lemmatization, NP/PP chunking and clause boundary detection. A glance at the results showed that person name and geographical name recognition were successful but company names were error-prone. Obviously, the keyword-based learning of company names needs to be adapted to the specifics of the new corpus. With the old learner it happened, that the acronyms for political parties (*CVP, SPD* etc.) were mistaken for company names.

We then extracted cooccurrence statistics over the annotated files. We used the same algorithm as in section 4.4.6. This includes using the core of compounds ("short" lemmas), and symbols for the three proper name classes. A look at the list of the N+P pairs with the highest cooccurrence values gives an impression of the difference in vocabulary between the NZZ and the CZ.

noun N_{lem}	P	$freq(N_{lem}, P)$	$freq(N_{lem})$	$cooc(N_{lem}, P)$
Zünglein	*an*	11	11	1.00000
Liborsatz	*für*	22	22	1.00000
Extraordinarius	*für*	12	12	1.00000
Dubio	*pro*	11	11	1.00000
Domenica	*in*	16	16	1.00000
Bezugnahme	*auf*	13	13	1.00000
Bezug	*auf*	338	339	0.99705
Hinblick	*auf*	350	355	0.98592
Partnership	*for*	23	24	0.95833
Diskothek	*in*	26	28	0.92857
Nachgang	*zu*	10	11	0.90909
Anlehnung	*an*	62	70	0.88571
Draht	*an*	248	292	0.84932
Horses	*in*	11	13	0.84615
Einblick	*in*	151	182	0.82967
Abkehr	*von*	58	70	0.82857

Based on the cooccurrence counts we computed the noun factor and the noun threshold according to the formulae introduced in sections 4.3.3 and 4.4.6. They are almost the same as for the CZ training.

When we had trained over the CZ corpus, we achieved an attachment accuracy of 78.30% and a coverage of 90.4%. With the NZZ training corpus, the accuracy is lower at 75.49% and the coverage at 80.9%.[2] The coverage reduction comes as no surprise. There are test

[2]In this chapter we use the CZ and NEGRA test sets based on verb lemmas, short noun lemmas and proper name classes. These test sets have been labeled $CZ_{shortlemma}$ and $NEGRA_{shortlemma}$ in chapter 4. The index will be omitted here.

	factor	correct	incorrect	accuracy	threshold
noun attachment	4.68	1819	411	81.57%	0.033
verb attachment		910	475	65.70%	0.154
total		2729	886	75.49%	
decidable test cases		3615 (of 4469) coverage: 80.9%			

Table 5.1: Attachment accuracy for the CZ test set.

cases specific to computer science, the domain of the CZ, with words that are not (frequently) found in a general newspaper.

verb	head noun	prep.	core of PP	PP function
portieren	*Linux*	*auf*	*Microkernel*	verb modifier
einloggen	*Browser*	*in*	*Datenbank*	verb modifier
drucken	*Dots*	*per*	*Inch*	noun modifier

But the decrease in the accuracy is more disturbing. We will apply sure attachment and triple cooccurrence values to work against this decrease.

In addition we used the NZZ training to evaluate against the NEGRA test set, which was compiled from another general newspaper, the Frankfurter Rundschau. We would thus not expect much difference in the attachment accuracy between the CZ training and the NZZ training.

	factor	correct	incorrect	accuracy	threshold
noun attachment	4.68	2176	616	77.94%	0.033
verb attachment		1287	608	67.91%	0.154
total		3463	1224	73.88%	
decidable test cases		4687 (of 5803) coverage: 80.8%			

Table 5.2: Attachment accuracy for the NEGRA test set.

The results are summarized in table 5.2 and need to be compared to the CZ training results in table 4.15 on page 112. Unlike the accuracy loss with the CZ test set, we observe a 1% accuracy gain for the NEGRA test set (from 72.64% to 73.88%). In addition, we notice a 7.8% gain in coverage (from 73% to 80.8%) based on the new training corpus. The accuracy difference between the CZ and the NEGRA test sets has shrunk from 5.66% (for the CZ training) to 1.61% for the NZZ training. This is clear evidence for the domain dependence of the disambiguation method. Training and testing over the same subject domain brings advantages both in terms of accuracy and coverage.

We now proceed to check whether the distinction between sure attachment PPs and ambiguous attachment PPs during the NZZ training leads to the same improvements as when we trained over the CZ corpus.

5.2 Sure Attachment and Possible Attachment

In the second training over the NZZ corpus we used the information about sure noun attachments and sure verb attachments. As in section 4.5 we counted PPs in sentence-initial constituents of matrix clauses as sure noun attachments. In addition, PPs following a frozen PP were counted as sure noun attachments. Sure noun PPs counted as one point for $freq(N, P)$. PPs in support verb units and PPs not following a noun were counted as sure verb attachments and scored one point for $freq(V, P)$. The count for all other PPs in ambiguous positions was split between $freq(N, P)$ and $freq(V, P)$.

The consideration of sure attachment PPs leads to a higher noun factor (5.96) and a lower noun threshold (0.02) which is the same tendency as observed in the CZ training. We use the same disambiguation algorithm as in section 4.6.1 which includes the direct access to support verb units. The results of table 5.3 based on the NZZ training are then comparable to table 4.18 on page 116 (based on the CZ training).

	factor	correct	incorrect	accuracy	threshold
noun attachment	5.96	1833	364	83.43%	0.020
verb attachment		977	454	68.27%	0.119
total		2810	818	77.45%	
decidable test cases		3628 (of 4469) coverage: 81.2%			

Table 5.3: Attachment accuracy for the CZ test set based on sure attachments.

The new disambiguation results confirm the findings from the CZ training. The consideration of sure attachment PPs in the training leads to improved disambiguation accuracy. In the CZ training the improvement was 3.22% including the application of almost sure attachment PPs, which we skipped in the NZZ training. Still, the accuracy improvement due to sure attachment PPs is close to 2% for the CZ test set in the NZZ training (from 75.49% to 77.45%).

The same type of improvement can also be observed for the NEGRA test set. Training with regard to sure attachment PPs improves the accuracy from 73.88% to 75.25% (cf. table 5.4).

	factor	correct	incorrect	accuracy	threshold
noun attachment	5.96	2180	565	79.42%	0.020
verb attachment		1351	596	69.39%	0.119
total		3531	1161	75.25%	
decidable test cases		4692 (of 5803) coverage: 80.8%			

Table 5.4: Attachment accuracy for the NEGRA test set based on sure attachments.

5.3 Using Pair and Triple Frequencies

In a final training over the NZZ corpus we extracted both pair frequencies and triple frequencies ((N_1, P, N_2) and (V, P, N_2)) in the manner described in section 4.12. We list some of the triples with the highest cooccurrence values in the following table. It includes idioms (*Zünglein an der Waage*), foreign language collocations (*in Dubio pro Reo, Work in Progress*), a city name (*Uetikon am See*), a special term from the stock exchange (*Liborsatz für Anlage*)[3], radio, TV and theatre programmes (*Ariadne auf Naxos; Auf Draht am Morgen/Mittag/Abend*), and governmental organizations (*Kommisariat für Flüchtlinge, Departementes für Angelegenheiten*). Programme titles that occur frequently in the newspaper may easily influence the cooccurrence values and should therefore be eliminated from the training.

noun N_1	P	noun N_2	$freq(N_1, P, N_2)$	$freq(V)$	$cooc(N_1, P, N_2)$
Draht	*an*	⟨time⟩	239.0	292	0.81849
Liborsatz	*für*	*Anlage*	16.5	22	0.75000
Zünglein	*an*	*Waage*	8.0	11	0.72727
Brise	*aus*	*West*	21.0	30	0.70000
Dubio	*pro*	*Reo*	4.5	11	0.40909
Generalkonsul	*in*	⟨location⟩	4.3	11	0.39091
Uetikon	*an*	*See*	6.5	17	0.38235
Work	*in*	*Progress*	4.0	11	0.36364
Dorn	*in*	*Auge*	11.5	32	0.35938
Ariadne	*auf*	*Naxos*	8.5	24	0.35417
Tulpe	*aus*	⟨location⟩	4.0	12	0.33333
Kommissariat	*für*	*Flüchtling*	19.5	60	0.32500
Widerhandlung	*gegen*	*Gesetz*	5.5	19	0.28947
Grand Prix	*von*	⟨location⟩	18.5	67	0.27612
Departementes	*für*	*Angelegenheit*	3.0	12	0.25000

We computed the noun factor separately for the pair cooccurrence values and the triple cooccurrence values. The triple cooccurrence value (6.66) is higher than the pair cooccurrence value (5.96). This corresponds roughly to the difference of the noun factors computed after the CZ training: 5.97 for the triples and 5.47 for the pairs.

In the evaluation of the triple cooccurrence values we use the same disambiguation algorithm as in section 4.12. This includes firstly the application of support verb units, then the cascaded application of triple and pair cooccurrence values and finally the comparison of the cooccurrence values against the thresholds. Adding triple comparison leads to an accuracy improvement of close to 1% for the CZ test set (cf. table 5.5).

The attachment accuracy for the NEGRA test set stays at the same level (formerly 75.25% now 75.28%) as documented in table 5.6.

A look at the decision levels reveals that only a minor fraction of the test cases (less than 10%) can be disambiguated on the basis of triple value comparisons. When we trained over the CZ corpus, more than 20% of the CZ test cases were handled by triple comparison. Therefore the impact of the triple value comparison is limited. It can also be seen that the verb threshold is too low and leads to accuracies far below the other decision levels.

[3]The NZZ corpus contains the word *Liborsatz* spelled with a hyphen, too: *Libor-Satz*.

	factor	correct	incorrect	accuracy	threshold
noun attachment	5.96; 6.66	1870	367	83.59%	0.020
verb attachment		976	420	69.91%	0.119
total		2846	787	78.34%	
decidable test cases		3633 (of 4469) coverage: 81.3%			

Table 5.5: Attachment accuracy for the CZ test set based on sure attachments and triple comparisons.

	factor	correct	incorrect	accuracy	threshold
noun attachment	5.96; 6.66	2212	594	78.83%	0.020
verb attachment		1323	567	70.00%	0.119
total		3535	1161	75.28%	
decidable test cases		4696 (of 5803) coverage: 80.9%			

Table 5.6: Attachment accuracy for the NEGRA test set based on sure attachments and triple comparisons.

	CZ test set		NEGRA test set	
decision level	number of cases	accuracy	number of cases	accuracy
support verb unit	97	100.00%	96	98.96%
triple comparison	283	79.15%	302	78.81%
pair comparison	3019	77.97%	3941	74.50%
$cooc(N_1, P) >$ threshold	82	82.93%	130	80.77%
$cooc(V, P) >$ threshold	152	67.67%	227	70.93%
total	3633	78.34%	4696	75.28%

In conclusion of this chapter we maintain that using a general newspaper training corpus will worsen the attachment accuracy and the coverage for the computer science newspaper test set (the CZ test set), but it will improve the accuracy and increase the coverage for the general newspaper test set (the NEGRA test set). The values for noun factors and thresholds are very much in line with training over the CZ corpus. Also the improvements for the consideration of sure attachments are parallel to our experiments in chapter 4. In the next chapter we will explore yet another corpus and its special access restrictions, the World Wide Web.

Chapter 6

Using the WWW as Training Corpus

In the previous chapters, our cooccurrence values were derived from locally accessible text corpora, the Computer-Zeitung and the Neue Zürcher Zeitung. Coverage was limited to 90% for test sets from the same domain as the training corpus and even lower if the test set and the training corpora were from different domains (80%).

In this chapter, we investigate a corpus that is many orders of magnitude larger than our local corpora; we compute the cooccurrence values from frequencies in the world wide web (WWW). Some WWW search engines such as AltaVista (`www.altavista.com`) provide a frequency ('number of pages found') for every query. We will use these frequencies to compute the cooccurrence values. When using the AltaVista frequencies, we cannot restrict the cooccurrence of N+P and V+P as precisely as when using a local corpus. Our hypothesis is that the size of the WWW will compensate the rough queries.

We owe the idea of querying the WWW for ambiguity resolution to [Grefenstette 1999]. He has shown that WWW frequencies can be used to find the correct translation of German compounds if the possible translations of their parts are known.

6.1 Using Pair Frequencies

When we worked with the local training corpora, the determination of unigram and bigram frequencies was corpus-driven. We worked through the corpora and computed the frequencies for all nouns, verbs, and all N+P and V+P pairs. This is not feasible for the WWW. Therefore the frequencies are determined test set-driven. We compiled lists from the CZ test set with all nouns, verbs and all N+P and V+P pairs. For every entry in the lists we automatically queried AltaVista.

AltaVista distinguishes between regular search and advanced search. Regular search allows for single word queries, multiple word queries (interpreted as connected by Boolean AND), and also queries with the NEAR operator. The NEAR operator in AltaVista restricts the search to documents in which the two words cooccur within 10 words.

Querying a WWW search engine for thousands of words is very time-consuming if every query finds only one frequency. We therefore used multiple word queries and extracted the frequency information from the list "The number of documents that contain your search

terms". In this way we got dozens of frequencies with one query. Unfortunately, this is restricted to regular search, and it does not work if the NEAR operator is used.

For all queries we used AltaVista restricted to German documents. In a first experiment[1] we assumed that all forms of a noun (and of a verb) behave in the same way towards prepositions and we therefore queried only for the lemmas. If a lemma could not be determined (e.g. if a word form was unknown to Gertwol as is often the case for proper names), the word form was used instead of the lemma.

- For nouns we used the nominative singular form in the queries. Compounds are reduced to their last element. For verbs we used the infinitive form in the queries. The prepositions were used as they appear in the test set (i.e. no reduction of contracted prepositions to their base forms).
- For cooccurrence frequencies we queried for `N NEAR P` and `V NEAR P`.

As an example, we will contrast cooccurrence values computed from Computer-Zeitung frequencies against values computed from WWW frequencies. We compare the highest cooccurrence values from the CZ based on word form counts. AltaVista provided the frequencies in columns 6 and 7 which led to the cooccurrence values in column 8.

		CZ training corpus			WWW training corpus		
noun N_{form}	P	$f(N,P)$	$f(N)$	$cooc(N,P)$	$f(N,P)$	$f(N)$	$cooc(N,P)$
Höchstmaß	*an*	13	13	1.00000	15,469	17,102	0.90451
Dots	*per*	57	57	1.00000	351	2155	0.16288
Bundesinstitut	*für*	12	12	1.00000	11,936	12,477	0.95664
Netzticker	*vom*	92	93	0.98925	4	59	0.06780
Hinblick	*auf*	133	135	0.98519	48,376	48,686	0.99363
Verweis	*auf*	21	22	0.95455	31,436	47,547	0.66116
Umgang	*mit*	293	307	0.95440	63,355	76,835	0.82456
Bundesministeriums	*für*	35	37	0.94595	33,714	36,773	0.91681
Bundesanstalt	*für*	70	75	0.93333	45,171	49,460	0.91328
Synonym	*für*	13	14	0.92857	14,574	20,841	0.69929
Verzicht	*auf*	51	55	0.92727	37,535	48,076	0.78074
Rückbesinnung	*auf*	12	13	0.92308	5,042	6,031	0.83601

In general the WWW cooccurrence values are lower than the CZ values (with the exception of *Hinblick, auf*). The differences are largest for domain-specific nouns such as *Dots* and *Netzticker*. Both *Verweis, auf* and *Verzicht, auf* seem to be influenced by low frequencies or by newspaper-specific usage in the CZ corpus. They score much lower in the WWW. The cooccurrence values for the governmental institutions are very similar including their relative ranking. With these constraints in mind, we computed the frequencies for all nouns, verbs, N+P pairs and V+P pairs occuring in the CZ test set.

6.1.1 Evaluation Results for Lemmas

The cooccurrence values will be applied as in the initial disambiguation algorithm in chapter 4: If both $cooc(N,P)$ and $cooc(V,P)$ are available, the higher value decides the attachment. Table 6.1 shows the results. The coverage is very high (98%). Only 92 test cases could not be

[1] The general ideas detailed in this section were published as [Volk 2000].

decided. The accuracy is low but we notice a bias towards verb attachment which results in a high accuracy for noun attachments (83.78%) and a very low accuracy for verb attachment (48.60%). We need to resort to the noun factor to work against this bias.

	correct	incorrect	accuracy
noun attachment	1250	242	83.78%
verb attachment	1402	1483	48.60%
total	2652	1725	60.59%
decidable test cases	4377 (of 4469) coverage: 98%		

Table 6.1: Results for the CZ_{lemmas} test set.

In principle, the noun factor is computed as described in section 4.3.3. We had computed it as the general attachment tendency of all prepositions to verbs against the tendency of all prepositions to nouns. The computation worked over all prepositions, nouns, and verbs from the training corpus. Now, we have to restrict ourselves to the cooccurrence values that we have, i.e. all values based on the test set. We determine a noun factor of 6.73. The noun factor is used to strengthen the noun cooccurrence values before comparing them to the verb cooccurrence values. The results are shown in table 6.2.

	factor	correct	incorrect	accuracy
noun attachment	6.73	2274	1003	69.39%
verb attachment		641	459	58.27%
total		2915	1462	66.60%
decidable test cases		4377 (of 4469) coverage: 98%		

Table 6.2: Results for the CZ test set with a noun factor.

The overall accuracy has increased from 60.59% to 66.60%. Still, this is a disappointing result. It is only 3% better than default attachment to nouns. Obviously, the imprecise queries to the WWW search engine lead to too much noise into the frequency data.

Cooccurrence value above threshold

Therefore we try to find a subset of the test cases for which the attachment quality is at least equal to that of our local corpora experiments. We observe that high cooccurrence values are strong indicators of a specific attachment. If, for instance, we require either $cooc(N, P)$ or $cooc(V, P)$ to be above a certain cooccurrence threshold, we may increase the accuracy. That means, we now use the following disambiguation algorithm:

```
if ( cooc(N,P) > threshold(N) ) && ( cooc(V,P) > threshold(V) ) then
  if ( (cooc(N,P) * noun_factor) >= cooc(V,P) ) then  noun attachment
  else  verb attachment
elsif ( cooc(N,P) > threshold(N) ) then  noun attachment
elsif ( cooc(V,P) > threshold(V) ) then  verb attachment
```

Unlike in our previous experiments, the thresholds are now also used to restrict the cooccurrence value comparison. We first set the noun threshold to the average noun cooccurrence value (0.216). This results in 1780 decided test cases with an accuracy of 80.51%. Second, we let the verb threshold to be the noun threshold times the noun factor, as we did in chapter 4. With the noun factor of 6.73 this results in a verb threshold of 1.45. None of the cooccurrence values will be above this threshold. Such a threshold can be discarded.

Then we tried to use the average verb cooccurrence value (0.31) as verb threshold. But this turned out to be too low. It would lead to a verb attachment accuracy of 60.37% (for 916 test cases). Manual fine-tuning showed that a verb threshold of 0.6 leads to a balanced result (see table 6.3).

	factor	correct	incorrect	accuracy	threshold
noun attachment	6.73	1425	331	81.15%	0.216
verb attachment		236	56	80.82%	0.600
total		1661	387	81.10%	
decidable test cases		2048 (of 4469) coverage: 46%			

Table 6.3: Results for the CZ test set based on threshold comparisons.

These results indicate that we can resolve 46% of the test cases with an accuracy of 81.10% by restricting the cooccurrence values to be above thresholds. But we have to concede that there is a "supervised" aspect in this approach. The manual setting of the verb threshold was based on observing the attachment results.

Minimal distance between cooccurrence values

As an alternative to a minimal cooccurrence threshold we investigated a minimal distance between $cooc(N, P)$ and $cooc(V, P)$. It is obvious that an attachment decision is better founded the larger this distance. Our disambiguation algorithm now is:

```
if ( cooc(N,P) ) && ( cooc(V,P) ) &&
   ( |( cooc(N,P) * noun_factor ) - cooc(V,P)| > distance ) then
  if ( (cooc(N,P) * noun_factor) >= cooc(V,P) ) then
    noun attachment
  else
    verb attachment
```

With a distance value of 0.95, we again reached 80.88% correct attachments and a coverage of 45%. So, there is not much difference to the minimum thresholds. But we observed an imbalance between noun attachment accuracy (80.57%) and verb attachment accuracy (91.38%). Obviously, the noun factor is too strong. If we adjust the noun factor to 4.5 and accordingly the minimal distance to 0.5, then we reach an accuracy of 80.80% with 50% coverage (see table 6.4). Alternatively, we may stick to the coverage of 46% (as for the threshold comparisons) and then reach 82.03% accuracy with a noun factor of 4.0 and a minimal distance of 0.5.

	factor	correct	incorrect	accuracy
noun attachment	4.5	1625	385	80.85%
verb attachment		172	42	80.37%
total		1797	427	80.80%
decidable test cases		2224 (of 4469) coverage: 50%		

Table 6.4: Results for the CZ test set with a minimal distance (0.5).

So, the minimal distance is superior to threshold comparisons in that it allows to resolve half of the test cases with an attachment coverage comparable to detailed corpus analysis. But again it requires manual adjustment of the noun factor and the minimal distance value.

6.1.2 Evaluation Results for Word Forms

In the first experiment with WWW-based cooccurrence values we had lemmatized all noun and verb forms. The intention was to reduce the number of values to be computed by mapping every word form to its lemma.

Obviously, the lemmatization introduces a number of potential errors. First, some word forms are ambiguous towards their lemma (e.g. *rasten* can be a form of either *rasen - to race* or *rasten - to rest*). When filtering for the correct lemma, we may pick the wrong one.[2]

Second, different word forms of a lemma may behave differently with respect to a given preposition. For instance, the plural noun *Verhandlungen* has a high rate of cooccurrence with the preposition *mit* since it is often used in the sense of "negotiations with". The singular form *Verhandlung* can be used in the same sense but is more often used in the sense of "hearing" or "trial" without the preposition. This is reflected in the different cooccurrence values:

noun N	prep P	$freq(N,P)$	$freq(N)$	$cooc(N,P)$
Verhandlung	*mit*	$10,444$	$41,656$	0.2507
Verhandlungen	*mit*	$43,854$	$55,645$	0.7881

In addition, the goal of reducing the sparse data problem by using lemmas rather than word forms cannot be achieved with AltaVista searches since AltaVista does not use a lemmatized index but full forms. And it is not self-evident that the lemma is the most frequently used form. The following table shows the AltaVista frequencies for the most important forms of the verbs *denken* and *zeigen*.

[2]Note, however, that some word forms might have homonyms that spoil the frequency value, whereas their lemma is unambiguous. As an example, think of the English verb form *saw* with its noun homonym, whereas searching the lemma *see* does not suffer from such interference.

person, number, tense	V	$freq(V)$	V	$freq(V)$
1st sg. present / imperative sg.	*denke*	107,348	*zeige*	42,224
2nd sg. present	*denkst*	17,496	*zeigst*	2,315
3rd sg. and 2nd pl. present / imperative pl.	*denkt*	101,486	*zeigt*	**446,642**
1st and 3rd pl. present / infinitive	*denken*	**228,928**	*zeigen*	366,287
past participle	*gedacht*	150,153	*gezeigt*	192,543

The frequency for *denken* is highest for the infinitive form, but for *zeigen* the frequency of the 3rd singular form (which also functions as 2nd plural and imperative plural form) is higher than of the infinitive form.

Therefore, we ran a second evaluation querying AltaVista with the full forms as they appear in the CZ corpus. Two small modifications were kept from our first set of experiments. In the case of hyphenated compounds we use only the last component (*Berlin-Umzug* → *Umzug*). And, as in all our experiments, a separated verbal prefix is attached (*deutete ... an* → *andeutete*) since the prefixed verb is different from its non-prefixed mother. The results are shown in table 6.5.

	factor	correct	incorrect	accuracy	threshold
noun attachment	6.73	2333	1014	69.70%	0.001
verb attachment		523	275	65.54%	0.001
total		2856	1289	68.90%	
decidable test cases		4145 (of 4469) coverage: 93%			

Table 6.5: Results for the CZ_{forms} test set with noun factor.

Compared to the lemma results (table 6.2), the coverage decreases from 98% to 93%, but the accuracy increases from 66.60% to 68.90%. This increase is in line with the higher accuracy we obtained for word forms over lemmas in chapter 4. The overall accuracy is still way below the 80% mark which we have come to expect from our local corpora experiments. Of course, restrictions with thresholds and minimal distance could be applied in the same manner as for the lemmas.

These experiments have shown that frequency values easily obtainable from WWW search engines can be used to resolve PP attachment ambiguities. But in order to obtain a sufficient level of accuracy, we had to sacrifice 50% test case coverage. In principle, the sparse data problem almost disappears when using the WWW as training corpus for cooccurrence frequencies. But the rough corpus queries with the NEAR operator include too much noise in the frequency counts. We will now extend the method to include the PP noun and query for triple frequencies.

6.2 Using Triple Frequencies

In the more successful experiments for PP attachment the cooccurrence statistics included the noun within the PP. The purpose of this move becomes immediately clear if we compare the PPs in the example sentences 6.1 and 6.2. Since both PPs start with the same preposition, only the noun within the PP helps to find the correct attachment.

(6.1) *Peter saw the thief* ***with his own eyes****.*

(6.2) *Peter saw the thief* ***with the red coat****.*

In a new round of experiments[3] we have included the head noun of the PP in the queries. Let us look at two example sentences from our corpus and the frequencies found in the WWW:

(6.3) *Die Liste gibt einen Überblick* ***über die 50 erfolgreichsten Firmen****.*

(6.4) *Unisource hat die Voraussetzungen* ***für die Gründung*** *eines Betriebsrates geschaffen.*

noun or verb W	P	noun N_2	$freq(W, P, N_2)$	$freq(W)$	$cooc(W, P, N_2)$
Überblick	*über*	*Firmen*	397	270,746	0.001466
Voraussetzungen	*für*	*Gründung*	274	255,010	0.001074
gibt	*über*	*Firmen*	513	1,212,843	0.000422
geschaffen	*für*	*Gründung*	139	172,499	0.000805

The cooccurrence values $cooc(N_1, P, N_2)$ are higher than $cooc(V, P, N_2)$, and thus the model correctly predicts noun attachment in both cases. Our test set consists of 4383 test cases from the CZ test set, out of which 63% are noun attachments and 37% verb attachments.[4]

We queried AltaVista in order to obtain the frequency data for our cooccurrence values. For all queries, we used AltaVista advanced search restricted to German documents. For cooccurrence frequencies we use the NEAR operator.

- For nouns and verbs, we queried for the word form by itself since word forms are more reliable than lemmas.
- For cooccurrence frequencies, we queried for `verb NEAR preposition NEAR N2` and `N1 NEAR preposition NEAR N2` again using the verb forms and noun forms as they appear in the corpus.

We then computed the cooccurrence values for all cases in which both the word form frequency and the cooccurrence frequency are above zero.

6.2.1 Evaluation Results for Word Forms

We evaluated these cooccurrence values against the CZ test set, using the most basic disambiguation algorithm including default attachments. If both cooccurrence values $cooc(N_1, P, N_2)$ and $cooc(V, P, N_2)$ exist, the attachment decision is based on the higher value. If one or both cooccurrence values are missing, we decide in favour of noun attachment since 63% of our test cases are noun attachment cases. The disambiguation results are summarized in table 6.6.

The attachment accuracy is improved by 6.5% compared to pure guessing, and it is better than using pair frequencies from the WWW. But it is far below the accuracy that we computed in the local corpora experiments. Even in the WWW, many of our test triples do not occur. Only 2422 (55%) of the 4383 test cases can be decided by comparing noun and verb cooccurrence values. The attachment accuracy for these test cases is 74.32% and thus about 5% higher than when forcing a decision on all cases (cf. table 6.7)

[3]This section has been published as [Volk 2001].

[4]The number of 4383 test cases dates from an earlier stage of the project.

	correct	incorrect	accuracy
noun attachment	2553	1129	69.34%
verb attachment	495	206	70.61%
total	3048	1335	69.54%

Table 6.6: Results for the complete CZ test set.

	correct	incorrect	accuracy
noun attachment	1305	416	75.83%
verb attachment	495	206	70.61%
total	1800	622	74.32%
decidable test cases	2422 (of 4383) coverage: 55%		

Table 6.7: Results for the CZ test set when requiring both $cooc(N_1, P, N_2)$ and $cooc(V, P, N_2)$.

6.2.2 Evaluation with Threshold Comparisons

A way of tackling the sparse data problem lies in using partial information. Instead of insisting on both $cooc(N_1, P, N_2)$ and $cooc(V, P, N_2)$ values, we will back off to either value for those cases with only one available cooccurrence value. Comparing this value against a given threshold, we decide on the attachment. Thus we extend the disambiguation algorithm as follows (which is comparable to the algorithm in section 4.4.6):

```
if (cooc(N1,P,N2) && cooc(V,P,N2)) then
  if (cooc(N1,P,N2) >= cooc(V,P,N2)) then  noun attachment
  else  verb attachment
elsif (cooc(N1,P,N2) > threshold) then
  noun attachment
elsif (cooc(V,P,N2) > threshold) then
  verb attachment
```

If we compute the threshold as the average cooccurrence value (like in chapter 4), we get 0.0061 for the noun threshold and 0.0033 for the verb threshold. With these thresholds we obtain an accuracy of 75.13% and a coverage of 59%. But the threshold comparisons by themselves result in much higher accuracy levels (94% for noun threshold comparison and 84% for verb threshold comparison). So, if we focus on coverage increase, we may further lower the threshold. That means, we set the thresholds so that we keep the overall attachment accuracy at around 75%.

We thus set the thresholds to 0.001 and obtain the result in table 6.8. The attachment coverage has risen from 55% to 63%; 2768 out of 4383 cases can be decided based on either both cooccurrence values or on the comparison of one cooccurrence value against the threshold.

	correct	incorrect	accuracy	threshold
noun attachment	1448	446	76.45%	0.001
verb attachment	629	245	71.97%	0.001
total	2077	691	75.04%	
decidable test cases	2768 (of 4383) coverage: 63%			

Table 6.8: Results for the CZ test set when requiring either $cooc(N_1, P, N_2)$ or $cooc(V, P, N_2)$.

6.2.3 Evaluation with a Combination of Word Forms and Lemmas

The above frequencies were based on word form counts. But German is a highly inflecting language for verbs, nouns and adjectives. If a rare verb form (e.g. a conjunctive verb form) or a rare noun form (e.g. a new compound form) appears in the test set, it often results in a zero frequency for the triple in the WWW. But we may safely assume that the cooccurrence tendency is constant for the different verb forms. We may therefore combine the rare verb form with a more frequent form of this verb. We decided to query with the given verb form and with the corresponding verb lemma (the infinitive form).

For nouns we also query for the lemma. We reduce compound nouns to the last compound element and we do the same for hyphenated compounds. We also reduce company names ending in *GmbH* or *Systemhaus* to these keywords and use them in lieu of the lemma (e.g. *CSD Software GmbH* → *GmbH*). We cannot reduce them to semantic class symbols as we did with our local corpora since we cannot query the WWW for such symbols. The cooccurrence value is now computed as:

$$cooc(W, P, N_2) = \frac{freq(W_{form}, P, N_2) + freq(W_{lemma}, P, N_2)}{freq(W_{form}) + freq(W_{lemma})}$$

The disambiguation algorithm is the same as above, and we use the same threshold of 0.001. As table 6.9 shows, the attachment accuracy stays at around 75%, but the attachment coverage increases from 63% to 71%.

	correct	incorrect	accuracy	threshold
noun attachment	1615	459	77.87%	0.001
verb attachment	735	300	71.01%	0.001
total	2350	759	75.59%	
decidable test cases	3109 (of 4383) coverage: 71%			

Table 6.9: Results for the CZ test set combining word form and lemma counts.

In order to complete the picture, we evaluate without using the threshold. We get an attachment accuracy of 74.72% at an attachment coverage of 65%. This is a 10% coverage

increase over the word forms result (cf. table 6.7 on page 160). If, in addition, we use any single cooccurrence value (i.e. we set the threshold to 0), the attachment accuracy slightly decreases to 74.23% at an attachment coverage of 85%. This means that for 85% of our test cases, we have at least one triple cooccurrence value from the WWW frequencies. If we default the remaining cases to noun attachment, we end up with an accuracy of 73.08%, which is significantly higher than our initial result for triple frequencies of 69.54% (reported in table 6.6 on page 160).

The most important lesson from these experiments is that triples (W, P, N_2) are much more reliable than tuples (W, P) for deciding the PP attachment site. Using a large corpus, such as the WWW, helps to obtain frequency values for many triples and thus provides cooccurrence values for most cases.

Furthermore, we have shown that querying for word forms and lemmas substantially increases the number of decidable cases without any loss in the attachment accuracy. We could further enhance the cooccurrence frequencies by querying for all word forms, as long as the WWW search engines index every word form separately, or by determining the most frequent word form beforehand.

If we are interested only in highly reliable disambiguation cases (80% accuracy or more), we may lower the number of decidable cases by increasing the threshold or by requiring a minimal distance between $cooc(V, P, N_2)$ and $cooc(N_1, P, N_2)$.

When using frequencies from the WWW, the number of decidable cases should be higher for English since the number of English documents in the WWW by far exceeds the number of German documents. Still the problem remains that querying for cooccurrence frequencies with WWW search engines using the NEAR operator allows only for very rough queries. For instance, the query `P NEAR N2` does not guarantee that the preposition and the noun cooccur within the same PP. It matches even if the noun N_2 precedes the preposition. We will now explore improved queries.

6.3 Variations in Query Formulation

WWW search engines are not prepared for linguistic queries, but for general knowledge queries. For instance, it is not possible to query for documents that contain the English word *can* as a noun. For the PP disambiguation task, we need cooccurrence frequencies for full verbs + PPs as well as for nouns + PPs. From a linguistic point of view we will have to use the following queries.

- For noun attachment, we would have to query for a noun N_1 occurring in the same phrase as a PP that is headed by the preposition P and contains the noun N_2 as head noun of the internal NP. The immediate sequence of N_1 and P is the typical case for a PP attached to a noun, but there are numerous variations with intervening genitive attributes or other PPs.

- For verb attachment, we would have to query for a verb V occurring in the same clause as a PP that is headed by the preposition P and contains the noun N_2. Unlike in English, the German verb may occur in front of the PP or behind the PP, depending on the type of clause.

Since we cannot query standard WWW search engines with linguistic operators ('in the same phrase', 'in the same clause'), we have to approximate these cooccurrence constraints with the available operators. In the previous section we used the NEAR operator (`V NEAR P NEAR N2`). In this section we investigate using more precise queries.

1. For verb attachment, we will query for `V NEAR "P DET N2"` with an appropriate determiner `DET`. This means that we will query for the sequence `P DET N2` NEAR the verb and thus ensure that P and N_2 cooccur in a standard PP. For contracted prepositions `PREPDET` (formed by a combination of a preposition and a determiner, like *am, ins, zur*), we do not need an explicit determiner and we will query for `V NEAR "PREPDET N2"`.

2. For noun attachment, we will query for `"N1 P DET N2"` with an appropriate determiner `DET`. This will search for the noun N_1 and the PP immediately following each other as it is most often the case, if the PP is attached to N_1.

3. For nouns and verbs, we query for the word form and the lemma by themselves.

Our test set again consists of the 4383 test cases from the CZ test set. We extract all tuples (P, N_2) from the test set and turn these tuples into complete PPs. We use the PP as found in the treebank (e.g. *mit elektronischen Medien*) and convert it into a "standard form" with the definite determiner (*mit den Medien*). If the PP in the treebank contains a number (e.g. *auf 5,50 Dollar*), it will be substituted by a "typical" number (*auf 100 Dollar*). If the preposition governs both dative and accusative case, two PPs are formed (e.g. *an dem/das Management*). We then combine the PPs with the verb V and the reference noun N_1 from the test set and query AltaVista for the frequency. For the triple (*Angebot für Unternehmen*), the following queries will be generated.

```
"Angebot für das Unternehmen"
"Angebot für die Unternehmen"
"Angebot für ein Unternehmen"
"Angebot für ihr Unternehmen"
"Angebot für Unternehmen"
```

The frequencies for all variations of the same triple will be added for the combined frequency of the triple. The five variations in our example lead to the WWW frequencies $5 + 4 + 0 + 44 + 100 = 153 =$ *freq(Angebot, für, Unternehmen)*.

For both verb and noun, we use the inflected form as found in the test set, and in a separate query we use the lemma. The lemma of a compound noun is computed as the base form of its last element. For example, we will thus query for:

```
lagen   NEAR "über den Erwartungen"
liegen  NEAR "über den Erwartungen"
"Aktivitäten im Internet"
"Aktivität im Internet"
"Ansprechpartnern bei den Behörden"
"Partner bei den Behörden"
```

Based on the WWW frequencies, we will compute the cooccurrence values by summing up the lemma triple frequencies and the word form triple frequencies and divide this sum by the sum of the lemma and word form unigram frequencies (as in section 6.2.3).

6.3.1 Evaluation with Word Forms and Lemmas

We first evaluate the cooccurrence values against the CZ test set using our standard disambiguation algorithm (without noun factor and threshold comparison). The results are summarized in table 6.10.

	correct	incorrect	accuracy
noun attachment	591	67	89.82%
verb attachment	392	344	53.26%
total	983	411	70.52%
decidable test cases	1394 (of 4383) coverage: 32%		

Table 6.10: Results for the CZ test set based on verb/noun+PP frequencies.

Out of 4383 test cases we can only decide 1394 test cases (32%) on the basis of comparing the cooccurrence values of both the verb and the noun. For 68% of the test cases, either $cooc(N_1, P, N_2)$ or $cooc(V, P, N_2)$ or both are unavailable due to sparse data in the part of the WWW indexed by the search engine. This result is way below the results in the previous section when we queried more vaguely for `W NEAR P NEAR N2`. With these triples we had observed an attachment accuracy of 74.72% and an attachment coverage of 65%. This attachment coverage was based on 77.05% correct noun attachments and 69.95% correct verb attachments.

In the new evaluation the difference between the noun attachment accuracy (89.82%) and the verb attachment accuracy (53.26%) is much larger. This is due to the asymmetry in the queries: for $cooc(V, P, N_2)$ we are using the NEAR operator, but for $cooc(N_1, P, N_2)$ we require a sequence of the words. We will counterbalance this asymmetry in the disambiguation algorithm again with the introduction of a noun factor. The noun factor is derived as described in section 4.3.3. The attachment accuracy is now much better (cf. table 6.11). It has increased from 70.52% to 79.05%.

	factor	correct	incorrect	accuracy
noun attachment	6.27	856	213	80.07%
verb attachment		246	79	75.69%
total		1102	292	79.05%
decidable test cases		1394 (of 4383) coverage: 32%		

Table 6.11: Results for the CZ test set based on verb/noun+PP frequencies and a noun factor.

6.3.2 Evaluation with Threshold Comparisons

Since the coverage is low, we try to increase it by adding threshold comparison to the disambiguation algorithm (as in section 4.4.6). In a first attempt we set the threshold to 0. This

means, we decide on an attachment if the respective cooccurrence value is available at all. The results are shown in table 6.12.

	factor	correct	incorrect	accuracy	threshold
noun attachment	6.27	1319	269	83.06%	0
verb attachment		728	402	64.42%	0
total		2047	671	75.31%	
decidable test cases		2718 (of 4383) coverage: 62%			

Table 6.12: Results for the CZ test set based on verb/noun+PP frequencies and thresholds.

The coverage has risen from 32% to 62%, but the attachment accuracy has dropped from 79.05% to 75.31%. In particular, the verb attachment accuracy has dropped from 75.69% to 64.42%. In fact, the attachment accuracy for the verb threshold comparison is 59.88% while the noun attachment accuracy for these comparisons is 89%. Obviously there are verb cooccurrence values $cooc(V, P, N_2)$ that are not reliable. We cut them off by setting the verb threshold to 0.001 (and maintain the noun threshold at 0).

	factor	correct	incorrect	accuracy	threshold
noun attachment	6.27	1319	269	83.06%	0
verb attachment		584	200	74.49%	0.001
total		1903	469	80.23%	
decidable test cases		2372 (of 4383) coverage: 54%			

Table 6.13: Results for the CZ test set based on verb/noun+PP frequencies and thresholds.

The attachment coverage is now at 54% with 2372 decidable cases. This means we can decide somewhat more than half of our test cases with an accuracy of 80% (cf. table 6.13).

6.4 Conclusions from the WWW Experiments

We have shown that frequencies obtainable from a standard WWW search engine can be used for the resolution of PP attachment ambiguities. We see this as one step towards "harvesting" the WWW for linguistic purposes.

This research supports earlier findings that using the frequencies of triples (W, P, N_2) is more reliable for the PP attachment task than using the frequencies of tuples (W, P), and the WWW provides useful frequency information for many triples (83% of our test cases). Many of the remaining test cases were not solved since they involve proper names (person names, company names, product names) as either N_1 or N_2. These names are likely to result in zero frequencies for WWW queries. One way of avoiding this bottleneck is proper name classification and querying for well-known (i.e. frequently used) representatives of the classes. As an example, we might turn *Computer von Robertson Stephens & Co.* into *Computer*

von IBM. Of course, it would be even better if we could query the WWW search engine for *Computer von* ⟨company⟩ which matched any company name.

When querying for standard PPs consisting of the sequence "P+DET+N2" with a specific determiner DET, we are severely limiting the search. The NP may occur with other determiners (indefinite or pronominal determiners) or with intervening adjectives or complex adjective phrases. Therefore it would be better if we could use a parametrizable NEXT operator (e.g. `P NEXT 3 N2`). This query will match if the noun N_2 follows the preposition as one of the next three words. This would make the query more flexible than a sequence but still restrict the search to the necessary order (P before N_2) and the typical range between preposition and noun. The NEXT operator is sometimes available in information retrieval systems but not in the WWW search engines that we are aware of.

Another possibility for improved queries is a SAME_SENTENCE operator that will restrict its arguments to cooccur within the same sentence. We could use it to query for verb attachments: `V SAME_SENTENCE (P NEXT 3 N2)` will query for the verb V cooccurring within the same sentence as the PP. From a linguistic point of view, this is the minimum requirement for the PP being attached to the verb. To be linguistically precise, we must require the verb to cooccur within the same clause as the PP. But none of these operators is available in current WWW search engines.

One option to escape this dilemma is the implementation of a linguistic search engine that would index the WWW in the same manner as AltaVista or Google but offer linguistic operators for query formulation. Obviously, any constraint to increase the query precision will reduce the frequency counts and may thus lead to sparse data. The linguistic search engine will therefore have to allow for semantic word classes to counterbalance this problem.

Another option is to automatically process (a number of) the web pages that are retrieved by querying a standard WWW search engine. For the purpose of PP attachment, one could think of the following procedure.

1. One queries the search engine for all German documents that contain the noun N_1 (or the verb V), possibly restricted to a subject domain.

2. A fixed number of the retrieved pages are automatically loaded. Let us assume the thousand top-ranked pages are loaded via the URLs provided by the search engine.

3. From these documents all sentences that contain the search word are extracted (which requires sentence boundary recognition).

4. The extracted sentences are compiled and subjected to corpus processing (with proper name recognition, PoS tagging, lemmatization etc.) leading to an annotated corpus similar to the one described in section 3.1.

5. The annotated corpus can then be used for the computation of unigram, bigram and triple frequencies.

The disambiguation results reported in this section are below the achievements of using local corpora and shallow parsing but they are surprisingly good given the ease of access to the frequency values and the rough queries. We assume that in the future natural language processing systems will query the WWW for ever more information when they need to resolve ambiguities.

Chapter 7

Comparison with Other Methods

In chapter 2 we introduced a number of statistical approaches for the resolution of PP attachment ambiguities. We will now describe the evaluation of three of these approaches against the cooccurrence value approach. We first look at an unsupervised approach, the Lexical Association score, and reformulate it in terms of cooccurrence values. We will then move on to the two most influential supervised approaches, the Back-off method and the Transformation-based method. Due to the lack of a large German treebank, we will alternately use one of our test sets as training corpus and the other one as test corpus. Finally, we will show that it is possible to intertwine unsupervised and supervised decision levels to get the best of both worlds into a combined disambiguation algorithm with complete coverage and high accuracy.

7.1 Comparison with Other Unsupervised Methods

7.1.1 The Lexical Association Score

In our experiments we have based the PP attachment decisions on comparisons of cooccurrence values. A competing association measure is the Lexical Association (LA) score introduced by [Hindle and Rooth 1993]. In section 2.2 we briefly mentioned this score and we will now provide more details and evaluate it by using our training and test data.

The Lexical Association score in its simplest form is defined as:

$$LA(V, N_1, P) = log_2 \frac{prob(verb\_attach\ P|V, N_1)}{prob(noun\_attach\ P|V, N_1)}$$

The decision procedure is then:

```
if ( lexical_association_score(V,N1,P) > 0 ) then
  verb attachment
elsif ( lexical_association_score(V,N1,P) < 0 ) then
  noun attachment
```

An LA score of exactly 0 means that there is no tendency for a specific attachment, and one has to leave the attachment either undecided or one has to resort to a default attachment.

As with the cooccurrence values, the probabilities are estimated from cooccurrence counts. But unlike in our approach, Hindle and Rooth include a NULL preposition for computing the probability of verb attachments.

$$prob(verb\_attach\ P|V, N_1) = \frac{freq(V,P)}{freq(V)} * \frac{freq(N_1, NULL)}{freq(N_1)}$$

$$prob(noun\_attach\ P|V, N_1) = \frac{freq(N_1,P)}{freq(N_1)}$$

[Hindle and Rooth 1993] argue for using the NULL preposition with verb attachments but not for noun attachments (p. 109):

> We use the notation NULL to emphasize that in order for a preposition licensed by the verb to be in the immediately postnominal position, the noun must have no following complements (or adjuncts). For the case of noun attachment, the verb may or may not have additional prepositional complements following the prepositional phrase associated with the noun.

In order to get a picture of the type of nouns with high and low NULL preposition values, we computed the NULL ratio and sorted them accordingly. The following table shows a selection of the nouns from the top and the bottom of this list.

noun N_1	$freq(N_1, NULL)$	$freq(N_1)$	$cooc(N_1, NULL)$
Language	171.50	172	0.99709
Verfügung	2239.05	2246	0.99691
Transaction	119.50	120	0.99583
Vordergrund	306.50	308	0.99513
Taufe	82.55	83	0.99458
Visier	177.00	178	0.99438
Tatsache	256.50	258	0.99419
Document	76.50	77	0.99351
Mitte	911.55	918	0.99297
...			...
Festhalten	5.05	15	0.33667
Made	7.40	24	0.30833
Stühlerücken	3.50	12	0.29167
Rückbesinnung	3.00	13	0.23077
Gegensatz	102.50	620	0.16532
Hinblick	3.50	135	0.02593

There is a suprisingly high number of nouns that have a strong tendency not to take any prepositional complements or adjucts. These include:

- English nouns that are part of a name (e.g. *Language* as part of *Programming Language One (PL/1), Structured Query Language, National Language Support (NLS)* etc.),
- nouns that form support verb units or idiomatic units and are thus positioned at the right end of the clause adjacent to the clause final punctuation mark or the verb group (this conforms to the order in the German *Mittelfeld* described in section 4.9). Such units are *zur Verfügung stehen/stellen, in den Vordergrund stellen/rücken, im Vordergrund stehen, aus der Taufe heben, im Visier haben, ins Visier nehmen.*

- nouns that tend to be followed by a *dass*-clause or occur in copula clauses (e.g. *die Tatsache, dass ...*),
- nouns that are used for measurement information and are thus followed by another noun or a genitive NP (e.g. *Mitte April, zur Mitte des Jahres*).

The bottom of the list is characterized by nouns that show strong prepositional requirements and hardly occur without a preposition. We have seen some of these nouns in the top cooccurrence lists in chapter 4.

Back to the Lexical Association score, we notice that in our terms the formula could be rewritten as:

$$LA(V, N_1, P) = log_2 \frac{cooc(V, P) * cooc(N_1, NULL)}{cooc(N_1, P)}$$

Since the logarithmic function is only a means of normalizing the decision procedure, the difference between the Lexical Association score and our cooccurrence value comparison boils down to the factor $cooc(N_1, NULL)$. The value of $cooc(N_1, NULL)$ approximates 1 if the noun N_1 often occurs without being followed by a PP. In other words, if N_1 seldom takes a prepositional complement or adjunct. In these cases the impact of this factor will be small. If, on the other hand, N_1 is often followed by a preposition, the factor weakens the verb attachment side. One could say that $cooc(N_1, NULL)$ describes the general tendency of N_1 to attach to any preposition.

We will now compare the Lexical Association score with the cooccurrence values using the same training and test corpora. Similar to Hindle and Rooth we use a partially parsed corpus as training material. We base our comparison on verb lemmas, short noun lemmas, and symbols for proper names as described in chapter 4. We use the weighted frequency counts as in section 4.9.3 and briefly repeated here:

1. A sure noun attached PP is counted as 1 for $freq(N_1, P)$.
2. A sure verb attached PP is counted as 1 for $freq(V, P)$.
3. The counts for ambiguously positioned PPs are split:
   - A local PP is split as 0.7 for $freq(N_1, P)$ and 0.3 for $freq(V, P)$.
   - A temporal PP is split as 0.45 for $freq(N_1, P)$ and 0.55 for $freq(V, P)$.
   - Other PPs are evenly split as 0.5 for $freq(N_1, P)$ and 0.5 for $freq(V, P)$.

These frequency counts include the "almost sure attachments" from section 4.5 which correspond to the incremental step in the Hindle and Rooth counting. In that step, LA scores greater than 2.0 or less than -2.0 (which presumably are sure attachments) are used to assign the preposition to the verb or to the noun respectively. The special split values for the local and temporal PPs were not used by Hindle and Rooth but are used here so that we get a clean comparison between the Lexical Association score and the cooccurrence values. Finally, we computed the $(N_1, NULL)$ frequencies as the difference between the unigram frequency of the noun and the bigram frequency of this noun with any preposition. For example, the noun *Laie* occurs 67 times in the CZ training corpus. It scores 1 point with

the preposition *an* and 0.5 points each with the prepositions *aus, bei* and *von*. That means, $freq(Laie, NULL) = 67 - 1 - (3 * 0.5) = 64.5$.

$$freq(N_1, NULL) = freq(N_1) - \sum_P freq(N_1, P)$$

Using the LA score in this way results in the disambiguation performance summarized in table 7.1.[1]

	correct	incorrect	accuracy
noun attachment	1307	73	94.71%
verb attachment	1319	1126	53.95%
total	2626	1199	68.65%
decidable test cases	3825 (of 4469) coverage: 85.6%		

Table 7.1: Results for the CZ test set based on the Lexical Association score.

Obviously, we have the same problem with the imbalance between noun attachment and verb attachment as we had in our experiments with the cooccurrence value. We therefore suggest to use the noun factor in the computation of the Lexical Association score.

$$LA(V, N_1, P) = log_2 \frac{cooc(V, P) * cooc(N_1, NULL)}{cooc(N_1, P) * noun\_factor}$$

This leads to the desired improvement in the attachment accuracy (81.44%) as table 7.2 shows.

Lexical Association score					cooc. values
	factor	correct	incorrect	accuracy	accuracy
noun attachment	4.58	2118	395	84.28%	85.51%
verb attachment		997	315	75.99%	73.37%
total		3115	710	81.44%	81.00%
decidable test cases		3825 (of 4469) coverage: 85.6%			85.6%

Table 7.2: Results for the CZ test set based on the Lexical Association score with noun factor.

In order to guarantee a fair comparison between these LA score results and the cooccurrence value results, we conducted a cooccurrence value experiment with the same noun factor and only with pair comparisons, i.e. no triple comparison and no threshold comparison.[2] This has to lead to the same coverage (85.6%). But it results in a slightly lower attachment accuracy (81.00%) (cf. the rightmost column in table 7.2). This means that there is a small positive influence of the $cooc(N_1, NULL)$ factor.

[1] In this chapter we use the CZ and NEGRA test sets based on verb lemmas, short noun lemmas and proper name classes. These test sets have been labeled $CZ_{shortlemma}$ and $NEGRA_{shortlemma}$ in chapter 4. The index will be omitted in this chapter.

[2] This is the same test as reported in table 4.17 but without the use of threshold comparisons.

Lexical Association with interpolation

The Lexical Association score depends on the existence of the values $cooc(V, P)$ and $cooc(N_1, P)$ in the same manner as the cooccurrence value comparison. If one of these values is 0, i.e. the pair has not been seen in the trainig corpus, then both scores are not defined and no disambiguation decision can be reached. We had therefore added the comparisons against thresholds which covered another 3% of the test cases.

[Hindle and Rooth 1993] suggest a different approach. They introduce a method for interpolation that devalues low frequency events but leads to an attachment decision in (almost) all cases. The idea is to redefine the probabilities with recourse to the general attachment tendency of the preposition as:

$$prob(noun\_attach\ P|V, N_1) = \frac{freq(N_1, P) + \frac{freq(all\_N,P)}{freq(all\_N)}}{freq(N_1) + 1}$$

with

$$freq(all\_N, P) = \sum_{N_1} freq(N_1, P) \quad \text{and} \quad freq(all\_N) = \sum_{N_1} freq(N_1)$$

When $freq(N_1)$ is zero, the estimate for $prob(noun\_attach\ P|V, N_1)$ is determined by $\frac{freq(all\_N,P)}{freq(all\_N)}$ which is the average attachment tendency for the preposition P across all nouns. If the training corpus contained one case of a noun and this occured with the preposition P (that is $freq(N_1) = 1$ and $freq(N_1, P) = 1$), then the estimate is nearly cut in half. When $freq(N_1, P)$ is large, the interpolation does not make much difference since it amounts to adding less than one to the counter and one to the denominator. The verb probability is redefined analogously. Accordingly, the Lexical Association is now computed as:

$$LA(V, N_1, P) = log_2 \frac{\frac{freq(V,P)+\frac{freq(all\_V,P)}{freq(all\_V)}}{freq(V)+1} * \frac{freq(N_1,NULL)+\frac{freq(all\_N,NULL)}{freq(all\_N)}}{freq(N_1)+1}}{\frac{freq(N_1,P)+\frac{freq(all\_N,P)}{freq(all\_N)}}{freq(N_1)+1} * noun\_factor}$$

Using the redefined Lexical Association score leads to almost complete attachment coverage for the CZ test cases and naturally to a decrease in the attachment accuracy since many test cases were disambiguated on the basis of rather weak evidence (cf. table 7.3).

Lexical Association score					cooc. values
	factor	correct	incorrect	accuracy	accuracy
noun attachment	4.58	2370	554	81.05%	78.73%
verb attachment		1134	403	73.78%	73.37%
total		3504	957	78.55%	77.02%
decidable test cases		4461 (of 4469) coverage: 99.82%			100%

Table 7.3: Results for the CZ test set based on the Lexical Association score with interpolation and noun factor.

But why is the coverage not complete? The interpolation relies on the fact that every preposition in the test set has been observed in the training set. If a preposition has not been

seen, then $freq(V, P) = 0$ and $freq(all\_V, P) = 0$ lead to $log_2(0)$ which is not defined. As we had mentioned in section 4.14, the preposition *via* is systematically mistagged as an adjective in our training corpus and as a consequence the few test cases with this preposition cannot be solved.

The attachment accuracy of 78.55% for the Lexical Association score with interpolation compares favorably with the attachment accuracy of 77.02% for the cooccurrence values plus default attachment (default is noun attachment). But this advantage disappears if the cooccurrence-based disambiguation algorithm steps from pair comparison to threshold comparison (with a noun threshold of 0.024 and an according verb threshold of 0.11) and then to default attachment. This step-down strategy leads to an attachment accuracy of 78.72% for the cooccurrence values (at complete coverage). But it remains to be explored if the LA score interpolation could be used to substitute default attachment. We will look at this option in section 7.3.

7.2 Comparison with Supervised Methods

In contrast to the unsupervised approaches that rely solely on corpus counts, the supervised approaches are based on manually disambiguated training material. In section 2.2.1 we have shown that supervised approaches achieved the best PP attachment results for English. We will explore two of these methods although we have only small training sets available.

7.2.1 The Back-off Model

In section 2.2 we presented the Back-off model as introduced by [Collins and Brooks 1995]. This model is based on the idea of using the best information available and backing off to the next best level whenever an information level is missing. For the PP attachment task this means using the attachment tendency for the complete quadruple (V, N_1, P, N_2) if the quadruple has been seen in the training data. If not, the algorithm backs off to the attachment tendency of triples. All triples that contain the preposition are considered. The triple information is used if any of the triples has been seen in the training data. Else, the algorithm backs off to pairs, then to the preposition alone, and finally to default attachment.

The attachment tendency on each level is computed as the fraction of the relative frequency to the absolute frequency. The complete algorithm is given in section 2.2. We reimplemented this algorithm in Perl. In a first experiment we used the NEGRA test set as training material and evaluated against the CZ test set. Both test sets were subjected to the following restrictions.

1. Verbs were substituted by their lemmas.
2. Contracted prepositions were substituted by their base forms.
3. Proper names were substituted by their name class tag (PERSON, LOCATION, COMPANY).
4. Pronouns (in PP complement position) were substituted by a pronoun tag.
5. Numbers (in PP complement position) were substituted by a number tag.

6. Compound nouns were substituted by their short lemma, and regular nouns by their lemma.

7. Test cases with pronominal adverbs, comparative particles and circumpositions were skipped.

This means we now use 5803 NEGRA quadruples with their given attachment decisions as training material for the Back-off model. We then apply the Back-off decision algorithm to determine the attachments for the 4469 test cases in the CZ corpus. Table 7.4 shows the results. Due to the default attachment step in the algorithm the coverage is 100%. The accuracy is close to 74% with noun attachment accuracy being 10% better than verb attachment.

	correct	incorrect	accuracy
noun attachment	2291	677	77.19%
verb attachment	1015	486	67.62%
total	3306	1163	73.98%
decidable test cases	4469 (of 4469) coverage: 100%		

Table 7.4: Back-off results for the CZ test set based on training over the NEGRA test set.

A closer look reveals that the attachment accuracy for quadruples (100%) and triples (88.7%) is highly reliable (cf. table 7.5) but only 7.5% of the test cases can be resolved in this way. The overall accuracy is most influenced by the accuracy of the pairs (that account for 68% of all attachments with an accuracy of 75.66%) and by the attachment tendency of the preposition alone which resolves 24.1% of the test cases but results in a low accuracy of 64.66%.

decision level	number	coverage	accuracy
quadruples	8	0.2%	100.00%
triples	329	7.3%	88.75%
pairs	3040	68.0%	75.66%
preposition	1078	24.1%	64.66%
default	14	0.3%	64.29%
total	4469	100%	73.98%

Table 7.5: Attachment accuracy for the Back-off method split on decision levels.

In a second experiment we exchanged the roles of training and test corpus. We now use the CZ test set as training material with the same restrictions as above and the NEGRA test set for the evaluation. That means, we now have only 4469 training quadruples to resolve the attachment in 5803 test cases. Of course, the result is worse than before. The attachment accuracy is 68.29% (see table 7.6). Quadruples and triples cover only 6%, pairs only 60%

	correct	incorrect	accuracy
noun attachment	2543	1045	70.87%
verb attachment	1420	795	64.11%
total	3963	1840	68.29%

Table 7.6: Back-off results for the NEGRA test set based on training over the CZ test set.

of the decisions. Too many cases are left for the uncertain decision levels of prepositional tendency and default.

This result indicates that the size of the training corpus has a strong impact on the disambiguation quality. Since we do not have access to any larger treebank for German, we used cross validation on the CZ test set in a third experiment. We evenly divided this test corpus in 5 parts of 894 test sentences each. We added 4 of these parts to the NEGRA test set as training material. The training material thus consists of 5803 quadruples from the NEGRA test set plus 3576 quadruples from the CZ test set. We then evaluated against the remaining part of 894 test sentences. We repeated this 5 times with the different parts of the CZ test set and summed up the correct and incorrect attachment decisions.

	correct	incorrect	accuracy
noun attachment	2402	546	81.48%
verb attachment	1146	375	75.35%
total	3548	921	79.39%

Table 7.7: Back-off results for the CZ test set based on training over the NEGRA test set and 4/5th of the CZ test set using cross-validation.

The result from cross-validation is 5% better than using the NEGRA corpus alone as training material (cf. table 7.6). This could be due to the enlarged training set or to the domain overlap of the test set with part of the training set. We therefore did an evaluation taking only the 4 parts of the CZ test set as training material. If the improved accuracy were a result of the increased corpus size, we would expect a worse accuracy for this small training set. But in fact, training with this small set resulted in around 77% attachment accuracy. This is better than training on the NEGRA test set alone. This indicates that the domain overlap is the most influential factor.

7.2.2 The Transformation-based Approach

In section 2.2 we presented the Transformation-based approach as introduced by [Brill and Resnik 1994]. In a greedy process a rule learning algorithm compiles transformation rules according to predefined rule templates. In the application phase these rules will be used to decide the attachments.

The learner starts with "noun attachment" as default in all cases. In each step it determines the rule that contributes most to the correction of the training set. The rule templates can access one specific word of the quadruple V, N_1, P, N_2 (4 templates), or any combination of two words (6 templates), or any triple that includes the preposition (3 templates). Any rule can change the attachment from noun to verb or vice versa.

As examples consider the topmost rules learned from the NEGRA test corpus with their score.

```
 1 change attachment from N to V if  N1 = <Person>            111
 2 change attachment from N to V if  P  = auf                  92
 3 change attachment from N to V if  N1 = Uhr                  52
 4 change attachment from N to V if  N1 = Jahr                 42
 5 change attachment from N to V if  N1 = <Location>           38
 6 change attachment from N to V if  P  = durch                23
 7 change attachment from N to V if  N2 = <Pronoun>            21
 8 change attachment from N to V if  N2 = Verfügung            17
 9 change attachment from V to N if  N1 = <Person> && P = von  13
10 change attachment from N to V if  P  = wegen                12
```

The first rule says that it is most profitable to change the decision from noun attachment (the default) to verb attachment if the reference noun N_1 is a person name. This is a very intuitive rule since person names are less likely to have modifiers than regular nouns and therefore a PP following a person name is more likely to attach to the verb than to the person name.

The second rule states a strong tendency for *auf*-PPs to attach to the verb rather than to the noun. This same rule is also the second rule learned from the CZ test set (with a score of 78). Temporal nouns like *Uhr* or *Jahr* are bad reference nouns for PPs and thus trigger verb attachment.

Rules 7 and 8 are based on the PP noun N_2. The noun *Verfügung* often occurs in support verb units like *zur Verfügung stellen/stehen* and is thus a typical indicator of verb attachment. Below are the topmost rules learned from the CZ test set.

```
 1 change attachment from N to V if  N1 = <Company>                     146
 2 change attachment from N to V if  P  = auf                            78
 3 change attachment from N to V if  N2 = Verfügung                      44
 4 change attachment from N to V if  N1 = <Location>                     39
 5 change attachment from N to V if  N1 = Jahr                           30
 6 change attachment from N to V if  N2 = <Pronoun>                      20
 7 change attachment from N to V if  N1 = Internet                       18
 8 change attachment from N to V if  N1 = <Product>                      17
 9 change attachment from V to N if  N1 = Zugriff                        16
10 change attachment from V to N if  N1 = <Company> &&  N2 = <Location> 15
```

It is striking how similar the topmost rules learned from both corpora are. Rule 10 of the CZ rule set shows a particular strength of Transformation-based learning, it undoes some of the transformations from rule 1. If a company name is followed by a PP denoting a

location, this PP should be attached to the noun, although in general a company name is a bad reference noun for any PP according to rule 1.

In a first experiment we trained on the NEGRA test set and evaluated against the CZ test set.[3] For the compilation of the training set we used the same restrictions as in the experiments with the Back-off model (section 7.2.1). Based on the 5803 quadruples, the Transformation-based learner collects 1297 rules. We apply all rules to the 4469 test cases of the CZ test set. Table 7.8 shows the results.

	correct	incorrect	accuracy
noun attachment	2249	708	76.06%
verb attachment	984	528	65.08%
total	3233	1236	72.34%

Table 7.8: Transformation-based results for the CZ test set based on training over the NEGRA test set.

The accuracy is 72.34% and thus about 1.5% lower than for the Back-off model (cf. table 7.4). Verb attachment accuracy is particularly low with 65.08%. These results confirm the reported results for English in that the Back-off model outperforms the Transformation-based approach. For the Penn data set the Back-off model achieved 84% accuracy and the Transformation-based approach 81%.

In order to get a complete comparison we increased the training material for the Transformation-based learner by using cross-validation over the CZ test set, as we did for the Back-off method. We split the CZ test set in 5 parts of equal size and used 4 parts together with the NEGRA material as training material. We evaluated against the fifth part of the CZ test set. This was repeated for all five parts. The combined results are listed in table 7.9.

	correct	incorrect	accuracy
noun attachment	2368	647	78.54%
verb attachment	1045	409	71.87%
total	3413	1056	76.37%

Table 7.9: Transformation-based results for the CZ test set based on training over the NEGRA test set and 4/5th of the CZ test set using cross-validation.

Using the enlarged training set and cross validation leads to an improvement in the attachment accuracy of 4% to 76.37%. So again we notice a considerable impact of the size of the training material as well as of the proximity of the training data to the test data. However, the Transformation-based approach loses ground against the Back-off model and is 2% below the corresponding Back-off accuracy (in table 7.7). We can conclude safely that the Back-off method is to be prefered for the PP attachment task.

[3]We used the original programs for rule learning and application as distributed by Eric Brill at `www.cs.jhu.edu/~brill/`.

This decision is backed by the implementation and application conditions. The Transformation-based approach is computationally much more costly. It is a matter of hours to compute the transformation rules from a few thousand training cases while it takes only seconds to compute the probabilities for the Back-off model.

7.3 Combining Unsupervised and Supervised Methods

Now, that we have seen the advantages of the supervised approaches, but lack a sufficiently large treebank for training, we suggest combining the unsupervised and supervised information. With the experiments on cooccurrence values and the Back-off method we have worked out the quality of the various decision levels within these approaches, and we will now order the decision levels according to the reliability of the information sources.

We reuse the triple and pair cooccurrence values that we have computed for the experiments in section 4.12. That means that we will also reuse the respective noun factors and thresholds. In addition, we use the NEGRA test set as supervised training corpus for the Back-off method.

The disambiguation algorithm will now work in the following manner. It starts off with the support verb units as level 1, since they are known to be very reliable (leading to 100% accuracy for the CZ test set). As long as no attachment decision is taken, the algorithm proceeds to the next level. Next is the application of supervised quadruples (level 2), followed by supervised triples (level 3). In section 7.2.1 we had seen that there is a wide gap between the accuracy of supervised triples and pairs. We fill this gap by accessing unsupervised information, i.e. triple cooccurrence values followed by pair cooccurrence values (level 4 and 5). Even threshold comparison based on one cooccurrence value is usually more reliable than supervised pairs and therefore constitutes levels 6 and 7. If still no decision has been reached, the algorithm continues with supervised pair probabilities followed by pure preposition probabilities. The left-over cases are handled by default attachment. Below is the complete disambiguation algorithm in pseudo-code:

```
if ( support_verb_unit(V,P,N2) ) then  verb attachment

elsif ( supervised(V,N1,P,N2) ) then
  if ( prob(noun_attach | V,N1,P,N2) >= 0.5 ) then  noun attachment
  else  verb attachment

elsif ( supervised( (V,P,N2) or (N1,P,N2) or (V,N1,P) ) then
  if ( prob(noun_attach | triple) >= 0.5 ) then  noun attachment
  else  verb attachment

elsif ( cooc(N1,P,N2) && cooc(V,P,N2) ) then
  if ( (cooc(N1,P,N2) * noun_factor) >= cooc(V,P,N2) ) then  noun attachment
  else  verb attachment

elsif ( cooc(N1,P) && cooc(V,P) ) then
  if ( (cooc(N1,P) * noun_factor) >= cooc(V,P) ) then  noun attachment
  else  verb attachment
```

```
elsif ( cooc(N1,P) > threshold(N) ) then  noun attachment

elsif ( cooc(V,P) > threshold(V) ) then  verb attachment

elsif ( supervised( (V,P) or (N1,P) or (P,N2) ) then
  if ( prob(noun_attach | pair) >= 0.5 ) then  noun attachment
  else  verb attachment

elsif ( supervised(P) ) then
  if ( prob(noun_attach | P) >= 0.5 ) then  noun attachment
  else  verb attachment

else default  verb attachment
```

And indeed, this combination of unsupervised and supervised information leads to improved attachment accuracy. For complete coverage we get an accuracy of 80.98% (cf. table 7.10). This compares favorably to the accuracy of the cooccurrence experiments plus default attachment (79.14%) reported in table 4.30 on page 140 and to the Back-off results (73.98%) reported in table 7.4 on page 173. We obviously succeeded in combining the best of both worlds into an improved behaviour of the disambiguation algorithm.

	factor	correct	incorrect	accuracy	threshold
noun attachment	5.47; 5.97	2400	469	83.65%	0.020
verb attachment		1219	381	76.19%	0.109
total		3619	850	80.98%	
decidable test cases		4469 (of 4469) coverage: 100%			

Table 7.10: Results for the combination of Back-off and cooccurrence values for the CZ test set (based on training over the NEGRA test set).

A look at the decision levels in table 7.11 reveals that the bulk of the attachment decisions still rests with the cooccurrence values, mostly pair value comparisons (59.9%) and triple value comparisons (18.9%). But the high accuracy of the supervised triples and, equally important, the graceful degradation in stepping from threshold comparison to supervised pairs (resolving 202 test cases with 75.74% accuracy) help to improve the overall attachment accuracy.

We have plotted the contributions of all decision levels in figure 7.1 on the facing page. The cumulative curves show the coverage and accuracy accumulated from decision level 1 to the current decision level. The split on decision levels illustrates that it is possible to achieve a certain level of accuracy if one is willing to sacrifice some coverage. Through the cumulative accuracy curve we see at decision level 8 that the combined disambiguation algorithm leads to over 82% accuracy at a coverage of 95%.

Since the application of the supervised probabilities for prepositions leads to an accuracy of only 60.48%, we exchanged this decision level for interpolation values from the Lexical Association score (as used by [Hindle and Rooth 1993] and described above in section 7.1.1).

decision level		number	coverage	accuracy
1	support verb units	97	2.2%	100.00%
2	supervised quadruples	6	0.1%	100.00%
3	supervised triples	269	6.0%	86.62%
4	cooccurrence triples	845	18.9%	84.97%
5	cooccurrence pairs	2677	59.9%	80.39%
6	$cooc(N_1, P) >$ threshold	71	1.6%	85.51%
7	$cooc(V, P) >$ threshold	81	1.8%	82.72%
8	supervised pairs	202	4.5%	75.74%
9	supervised prepositions	210	4.7%	60.48%
10	default	11	0.3%	54.55%
total		4469	100.0%	80.98%

Table 7.11: Attachment accuracy based on decision levels.

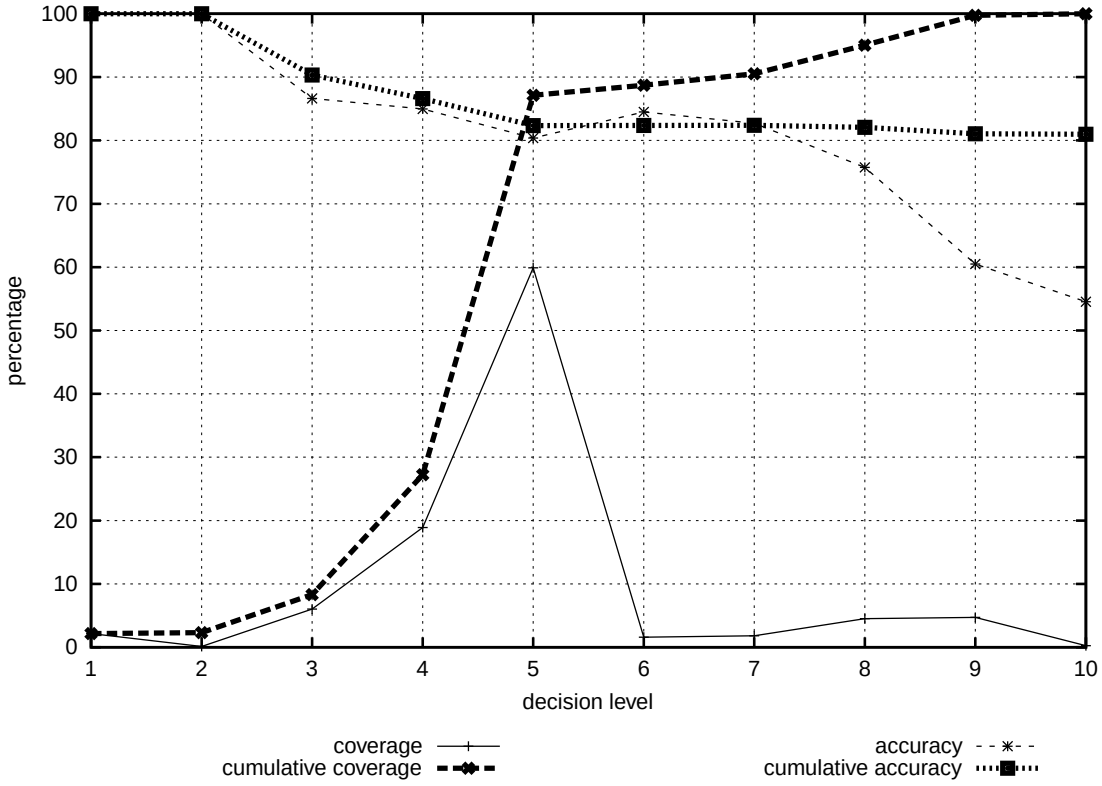


Figure 7.1: Coverage and accuracy at the decision levels

But it turned out that the interpolation values in this position only lead to an accuracy of 58.55%. So, the supervised preposition probabilities are to be prefered.

We also checked whether the combination of unsupervised and supervised approaches leads

to an improvement for the NEGRA test set. We exchanged the corpus for the supervised training (now the CZ test set) and evaluated over the NEGRA test set. This results in an accuracy of 71.95% compared to 68.29% for pure application of the supervised Back-off method (cf. table 7.6). That means, the combination leads to an improvement of 3.66% in accuracy. If we use the cooccurrence values derived from the NZZ (as in chapter 5) instead of those from the CZ corpus, the combined approach leads to another improvement of 1.24% to 73.19% correct attachments.

This chapter has shown that unsupervised approaches to PP attachment disambiguation are about as good as supervised approaches over small training sets. Both unsupervised and supervised methods will profit from training sets from the same domain as the test set. The combination of unsupervised and supervised information sources leads to the best results.

Chapter 8

Conclusions

8.1 Summary of this Work

We have presented an unsupervised method for PP attachment disambiguation. The method is based on learning cooccurrence values from a shallow parsed corpus. To build such a corpus we have compiled a cascade of corpus processing tools for German, including proper name recognition and classification, part-of-speech tagging, lemmatization, NP/PP chunking and clause boundary detection.

The method has been evaluated against two different German training corpora (Computer-Zeitung and Neue Zürcher Zeitung) and two different test sets, the NEGRA test set, derived from a 10,000 sentences treebank, and the CZ test set from our own 3,000 sentences special purpose treebank. Our tests showed that statistical methods for PP attachment are dependent on the subject domain of the training corpus. We observed better results if the training corpus and the test set were from the same domain.

We have explored the use of linguistic information with statistical evidence. We found that some linguistic information is advantagous such as the distinction between sure and possible attachment in training or the use of support verb units in the disambiguation algorithm. Other linguistic distinctions (such as reflexive verbs and PPs in idioms) did not lead to improvements.

We have shown that the unsupervised approach is competitive with the supervised approaches if supervised learning is limited by a small amount of manually annotated training material. Most interestingly, we have demonstrated that an intertwining of our unsupervised method and the supervised Back-off method is possible and leads to the best attachment results both in terms of coverage and accuracy. These results are slightly worse than those reported for English using the same resources. This is due to the strong impact of *of*-PPs in English which are very frequent and almost exclusively need to be attached to nouns.

As a sidestep, we have experimented with frequency counts from WWW search engines. They constitute the easiest way of obtaining cooccurrence frequencies over a vast corpus. Since the query formulation is imprecise in linguistic terms, these frequency counts need to be employed with restrictions.

We will make the 4562 test cases from the CZ test set available through the WWW so that they can be used as a benchmark for more experiments on PP attachment for German (`www.ifi.unizh.ch/CL/NLP_Resources.html`). The clause boundary detector can be tested

over the WWW in combination with our tagger.[1] The modules for corpus preparation will be made available to interested parties upon request. It should be noted that many of these modules rely on Gertwol, which is a commercial product licensed by Lingsoft Oy, Helsinki.

8.2 Applications of this Work

The proposed methods for corpus processing and the correct attachment of PPs will help in many areas of natural language processing.

Corpus annotation. Improved corpus annotation with proper names, NP/PP chunks, local and temporal PPs as well as PP attachments opens new opportunities for corpus searches. Our corpus annotation allows the linguist to query, for instance, for clauses with a person name in topic position and a temporal PP followed by a local PP. It will also provide a basis for improved computation of verbal and nominal subcategorization frames. Proper name recognition delimits the unknown word problem for subsequent processing modules and improves part-of-speech tagging.

Improving answer extraction. We stated at the beginning that our ultimate goal is the implementation of an answer extraction system. We will include a parser for German to determine the relationships of the phrasal constituents within each sentence. We see correct PP attachment as an important step from chunk parsing to full parsing.

Improving machine translation. PP attachment is a problem for machine translation systems (as exemplified in section 1.4). Our disambiguation algorithm alleviates the resolution of such ambiguities.

As every scientific endeavour this work has brought up more new questions than it answered. We see various ways in which the current work can be extended.

8.3 Future Work

8.3.1 Extensions on PP Attachments

One attachment option for PPs ignored in this book is the attachment of the PP to an adjective. In comparison to noun and verb attachment, adjective attachment is rare and in many cases not ambiguous. As mentioned in section 1.3, PP ambiguities between verb and adjective attachment occur most often for deverbal adjectives (i.e. participle forms used as adjectives). Our cooccurrence-based approach ought to work for these ambiguities in the same manner as for noun-verb ambiguities.

Another aspect that we have only touched on are circumpositions and postpositions and the attachment of the respective phrases. Once such phrases have been recognized, the attachment problem is the same as for PPs. However, circumpositions are often semantically more restricted and may thus provide more clues to the correct attachment than is available for PPs. For example, the preposition *zu* can introduce local, temporal, modal and other PPs, but in the circumposition *zu ... hin* it is constrained to denote a local phrase.

[1] See `www.ifi.unizh.ch/CL/tagger`.

We have reduced the PP attachment problem to a classification task over the quadruple (V, N_1, P, N_2). But, as [Franz 1996a] remarks, looking only at two possible attachment sites makes PP attachment appear easier than it is. Often a sentence contains a sequence of two or more PPs. Example sentence 8.1 contains seven prepositions in one clause with five PPs in immediate sequence. The *von*-PP is a clear noun attachment because of its position in the *Vorfeld*. The *zur*-PP is ambiguous between adjective attachment and verb attachment. The *ab*-PP has three possible attachment sites, the noun *Ausgabe* or the genitive noun *Blattes* or the verb. The *in*-PP has the same possible attachment sites as the preceding *ab*-PP plus the noun *Uhr* from that PP. Consequently, the *auf*-PP has five possible attachment sites and the *über*-PP has six possible attachment sites, although the attachment to the first nouns in the sequence with three or four intervening constituents is highly unlikely.

(8.1) *Die Abonnenten* ***von*** *Chicago Online können parallel* ***zur*** *gedruckten Ausgabe ihres Blattes* ***ab*** *8.00 Uhr morgens* ***in*** *einer inhaltlich gleichen elektronischen Version* ***auf*** *dem Computerbildschirm* ***über*** *ein Stichwort gezielt* ***nach*** *Artikeln suchen.*

But the choice of attachments in such a PP sequence is not independent. If the system determines that the *ab*-PP is a temporal PP and should therefore be attached to the verb, the subsequent PPs cannot be attached to nouns that precede the *ab*-PP.

The dependence is also evident for typical PP pairs. Some PPs often cooccur to denote, for instance, a local or temporal range. Examples are (*von - nach, von - bis, von - zu*) sometimes including an intermediate step with *über* (see the example sentences 8.2 and 8.3). 8.4 is a counterexample to illustrate that not all *von-nach*-PP sequences can be interpreted as denoting a range. As additional condition the PPs need to belong to the same semantic class.

(8.2) *... durch die am 4. März erfolgte Inbetriebnahme der ersten High-Speed-Verbindung über Lichtwellenleiter* ***von Hongkong nach Peking****.*

(8.3) *... reicht* ***von*** *einfachen MIS-Systemen* ***über*** *ambitionierte "Management-by"-Modelle* ***bis hin zu*** *radikalen Lean-Enterprise-Lösungen.*

(8.4) *Mit rund 30 Unternehmensberatern von Jay Alix holte sich Unruh eine teure Truppe* ***von Turnaround-Spezialisten nach Pennsylvania****.*

In order to take such interdependies into account, we will have to enlarge the disambiguation context. At least we will have to move from quadruples to quintuples (V, P_1, N_1, P_2, N_2). This will also help to identify frozen PPs (e.g. *im Gegensatz zu, mit/ohne Rücksicht auf*) and to systematically treat them as noun attachments.

Another argument for the usage of a larger context comes from passive sentences. In German passive sentences the subject of the corresponding active sentence is realized by a *von*-PP. We suspect that we could exploit this regularity if the passive information were represented in the test quadruples. In example 8.5 the *von*-PP is in an ambiguous position and could be attached to the verb based on the information that it occurs in a passive sentence. But example 8.6 indicates that this heuristic is not always correct. The PP *von IBM* is truely ambiguous even for the human reader and the passive mood of the sentence does not make it a clear case for verb attachment.

(8.5) *Nach eigenen Angaben werden rund 60 Prozent aller in Deutschland ausgegebenen Visa-Kartenprogramme* ***von B+S*** *betreut.*

(8.6) *Nach einem Bericht des Wall Street Journals wird die langfristige Strategie* ***von IBM*** *in Frage gestellt.*

(8.7) *Diese Projektaufgaben wurden* ***von der FIBU-Abteilung*** *übernommen.*

(8.8) *Als Kaufpreis wird* ***von Knowledge Ware*** *eine Spanne von 18 bis 30 Millionen US-Dollar angegeben.*

In fact, most often the subject-bearing *von*-PP in a passive sentence will be positioned right after the finite verb (i.e. not in an ambiguous position; see 8.7 and 8.8). In the latter example sentence there is a second *von*-PP within a *von-bis* pair which is noun attached.

Finally, we noted that prepositions, although very short words, are sometimes abbreviated. Our NZZ corpus contains, for instance, *Affoltern a. A., Frankfurt a. M., Wangen b. Olten, Aesch b. N., Burg i. L.* These abbreviated prepositions occur mostly with city names and the PP should be treated as part of a complex name.

8.3.2 Possible Improvements in Corpus Processing

We have devoted large efforts to annotate our training corpora through automatic corpus processing. Corpus annotation was governed by the task at hand, i.e. learning cooccurrence values for PP attachment disambiguation. But of course, the annotations can also be used for other information extraction tasks. For example, if we search information about companies, we might be interested in the company location, its managers, its products, its relations to other companies, and its financial standing. Towards this goal corpus processing could be enhanced in a number of ways.

Use of morpho-syntactic features

The most notable omission in our corpus processing scheme is the lack of morpho-syntactic agreement features. This may be puzzling at first sight since Gertwol outputs number, case and gender for any of its known nouns and corresponding features for known adjectives, determiners and verbs. The difficulty lies in compacting this information to a manageable format. If Gertwol states that a noun form could be nominative, genitive and accusative, we need to apply unification of feature structures with the other words in the NP in order to narrow down the set of possible values.

The use of such features will help to avoid incorrect NPs and PPs in NP/PP chunking if the features are contradictory. And it will also help to identify genitive NPs so that we may attach them as noun attributes.

Coreference identification

As part of corpus processing we recognized and classified proper names of persons, locations and companies. If we were to use the entities for knowledge extraction, it would be helpful to identify the coreference relations. This means that we identify various forms that refer to the same object. Some coreference relations fall out of our learning procedures:

1. the relation between a full person name (*Dr. Erich Roeckner*) and a person last name (*Roeckner*),
2. the relation between a name in base form and its genitive form (*Kanther, Kanthers; Hamburg, Hamburgs*),
3. the relation between a complex company name and its core name (*die Münchner Inplus GmbH → Inplus*), and
4. the relation between a complex company name and its acronym if the acronym is part of the complex name (*UBS Securities Asia Ltd. → UBS*).

Other relations could be inferred as well if the learning algorithm is appropriately adapted.

1. Often a company name is followed by its abbreviation in parentheses when it is first introduced. So our program could learn the abbreviation and establish the relation between the full company name and its abbreviation.

   (8.9) ***Nippon Telegraph und Telephone (NTT)*** *rechnet für das Geschäftsjahr 1993/94 mit ...*

2. The location of a company can be inferred from the geographical adjective in the pattern which we use for company name classification (*die Münchner Ornetix → Ornetix* is located in *München*).
3. The affiliation of a person to a company is often added as an apposition with the person's function description (*Innenminister, Geschäftsführer*). From the following example sentence the relations between a person and her company and between the company and its location could be infered.

   (8.10) *Für Ulrike Poser, Geschäftsführerin der Industrie-Service Tonträger GmbH (IST) im baden-württembergischen Reute gibt es nichts Besseres.*

4. The relation between a geographical name in its base form and its adjective forms (*Hamburg, Hamburger; Deutschland, deutsche*).

Proper name classification

The hypothesis that all names of a semantic class behave the same with respect to any given preposition is plausible and our test results lend some evidence to it. But it is not proven in this book. Maybe full person names behave differently from person last names. But if the hypothesis is true, one could also explore the reverse direction. If an unknown word W behaves similar to the members of the name class C, we might conclude that W is a member of C.

Proper name recognition and classification are important parts of corpus processing. We see the following directions for improvements in precision and recall.

1. We could apply Gertwol before name recognition so that Gertwol's information on proper names (via the EIGEN tag) could be used as part of the judgement (to increase confidence in a name recognition hypothesis).

2. The interaction between the recognition modules needs to be improved. As it stands, the modules for proper name recognition work independently, starting with the most reliable: person name recognition, then geographical names and company names. If a name is classified, the classification will not be overwritten by a subsequent module. This leads to errors like the classification of a person name within a company name (*des Münchner Anbieters* ***Otto Förg*** *Groupware*). We would rather have all modules compete with one another about the classification of a name.

3. Coordinated constituents need to be exploited. For example, our name recognition module learned that *Ernst & Young* is a company name but it did not classify *Knowledge Ware*. From the coordination in example 8.11 it could infer that *Knowledge Ware* is also a company name.

   (8.11) *Etwas anders verhält es sich bei dem ebenfalls noch im Dezember letzten Jahres ausgehandelten Deal zwischen* ***Knowledge Ware*** *und* ***Ernst & Young****.*

4. Other name types (product names, organization names, event names) need to be included.

8.3.3 Possible Improvements in the Disambiguation Algorithm

In chapter 7 we have described an intertwined disambiguation algorithm that uses both supervised and unsupervised information. We have observed that the decision levels 8 through 10 (supervised pairs, supervised prepositions and default) lead to low attachment accuracies. There are a number of alternatives for these decision levels that need to be tried.

- It might be advantagous to use triple frequencies from the WWW.
- For test cases with rare verbs we might employ the CELEX subcat information if the verb is listed in CELEX as having only one reading with an obligatory prepositional object.
- If applicable, we might use the transfer of verb cooccurrence values to deverbal nouns, and we might use the information that deverbal nouns often require the preposition *von* (cf. section 4.7).
- We might try to recognize systematically ambiguous cases and leave them undecided (cf. section 1.3).

We have computed the cooccurrence values for N+P and V+P pairs and the corresponding triples with a maximum likelihood estimate. This estimate leaves no probability mass for unseen events and accordingly assigns zero probability to such events. [Manning and Schütze 2000] (section 6.2) describe a number of methods to reserve some probability mass for unseen events by systematically decreasing the probability of seen events (Laplace's Law, Lidstone's Law, Good-Turing estimation). These should also lead to smoothing the probabilities of low frequency events. This needs to be tested, but we doubt that it will have a substantial impact on the disambiguation results. The interpolation experiments in sections 7.1.1 and 7.3 (as suggested by [Hindle and Rooth 1993]) did not lead to any improvements.

8.3.4 Integrating PP Attachment into a Parser

This work has paved the road to disambiguate PP attachments. To make full use of the opportunities, we need to integrate the disambiguation algorithm into a parser. First, we could add PP attachment as a decision procedure in shallow parsing. After NPs and PPs have been recognized, the disambiguator could mark all PPs as belonging to the noun or to the verb and integrate them into the respective phrases.

Second, the PP attachment disambiguator can be an integral part of a probabilistic parser. The subcategorization constraints within each clause could be fed back to the PP disambiguator to restrict its operations. The cooccurrence values could be integrated into the computation of the overall sentence probability. More research is necessary to determine the effects of such an integration. One should not forget that the noun factor as introduced in chapter 4 drives the cooccurrence value beyond the scope of probability theory. It can easily lead to a cooccurrence value greater than one.

8.3.5 Transfer to Other Disambiguation Problems

PP attachment ambiguities are a prominent ambiguity class. But others such as coordination disambiguation or word sense disambiguation are of similar importance. We claim that these could also be tackled with cooccurrence values although we know that more factors will come into play.

Let us look at a specific instance of a coordination ambiguity. In a sequence ($adj, N_1, coord$, N_2) it is possible to attach the adjective only to N_1 or to the coordinated sequence. If we determine that the adjective has a high cooccurrence value with N_1 and a low value with N_2, we might conclude that it should only modify N_1. Factors like the syntactic and semantic symmetry between N_1 and N_2 need also be considered.

Other attachment ambiguities arise from pre- vs. post-NP genitives (*Deutschlands Beitrag in der EU* vs. *der Beitrag Deutschlands in der EU*) or from relative clause or apposition attachments. Our claim is that constituent attachment in language understanding is inherently determined by the frequency of cooccurrence of the constituents. And therefore all kinds of attachment ambiguities can be solved through appropriate cooccurrence frequencies.

A similar approach is possible for word sense disambiguation. If a noun N_1 cooccurs frequently with another noun N_2, they will constrain each others senses. [Manning and Schütze 2000] (chapter 7) give an overview of word sense disambiguation methods that are based on statistical evidence.

The methods, tools and resources of our project will not only be useful for our specific task of answer extraction but also for neighboring fields in Computational Linguistics. The shallow parser which identifies noun phrases can also be applied to term extraction which is an important sub-task in the compilation of terminology databases (used for human or automatic translation of texts). The parser can also be employed in grammar checking or in computer-aided language learning programs, in mail-routing systems or in fact extraction systems.

Looking back over the activities to resolve natural language ambiguities in the last 40 years, the following pattern emerges. In the early stages of NLP one tried to apply the computer to a wide range of language phenomena and failed because of a lack of computational and linguistic resources. Subsequently, there was a period of using deep knowledge for a small set

of words which resulted in systems for limited domains. Since the beginning of the 90s the focus has switched again. We are working on broad coverage NLP, since now we do have the computational power and the necessary linguistic resources (corpora, test suites, lexicons). In text corpora a wealth of knowledge lies before us that is still largely untapped.

Appendix A

Prepositions in the Computer-Zeitung Corpus

This appendix lists all prepositions of the ComputerZeitung (1993-95+1997). We have added the classification as either primary or secondary preposition. Our list comprises 21 primary prepositions. The debatable ones are *ohne* and *wegen*. Since they do not form pronominal adverbs, it is not obvious that they can be used for prepositional objects. But as we show in appendix C there are rare pronominal adverb forms with *wegen*, and *ohne* is listed twice in the CELEX database (as prepositional object requirement for *auskommen* and *sich behelfen*). [Helbig and Buscha 1998] also mention *während* as a primary preposition. Since we are not aware of examples that this preposition introduces an object PP, we prefer to treat it as a secondary preposition.

Furthermore we have added the case requirement (accusative, dative, genitive), contracted forms that occur in our corpus, pronominal adverb forms and special notes. In the notes column we mark if the preposition can be used as a postposition (pre/post) and if it combines with other prepositions. Pure postpositions and circumpositions are not listed. The prepositions *bis* and *seit* can be combined with another preposition (marked as '+ prep'). The preposition *seiten* (rank 62) is unusual. It occurs only in combinations like *auf seiten* or *von seiten* and can be considered a dependent element of a complex preposition. It is related to *seitens* (rank 54) and similar in meaning.

Finally, we note all prepositions that can cooccur with the preposition *von*, in particular the following preposition families:

- local prepositions:
  - *fern, längs, unweit*
  - *oberhalb, unterhalb, innerhalb, ausserhalb*
  - *jenseits, abseits, diesseits, beiderseits, seitlich*
  - *südlich, westlich, östlich, nördlich, nordöstlich, nordwestlich, südöstlich, südwestlich*
- PP-based prepositions: *anhand, anstatt, anstelle, aufgrund, infolge, inmitten, zugunsten, zuungusten*
- (seldom with *von*): *abzüglich, anläßlich, bezüglich, hinsichtlich, vorbehaltlich*

- (seldom with *von*): *einschliesslich, ausschliesslich, inklusive, exklusive*

The English preposition *for* (rank 33) is included since it occurs so frequently in this corpus and was recognized as preposition by the part-of-speech tagger.

rank	preposition	frequency	type	case	contr.	pron. adv	special
1	*in*	84662	prim.	acc/dat	*im/ins*	*darin*	
2	*von*	71685	prim.	dat	*vom*	*davon*	
3	*für*	64413	prim.	acc	*fürs*	*dafür*	
4	*mit*	61352	prim.	dat		*damit*	
5	*auf*	49752	prim.	acc/dat	*aufs*	*darauf*	
6	*bei*	27218	prim.	dat	*beim*	*dabei*	
7	*über*	19182	prim.	acc/dat	*überm/s*	*darüber*	pre/post
8	*an*	18256	prim.	acc/dat	*am/ans*	*daran*	
9	*zu*	17672	prim.	dat	*zum/zur*	*dazu*	
10	*nach*	15298	prim.	dat		*danach*	pre/post
11	*aus*	13949	prim.	dat		*daraus*	
12	*durch*	12038	prim.	acc	*durchs*	*dadurch*	(pre/post)
13	*bis*	11253	sec.	acc			(+ prep)
14	*unter*	10129	prim.	acc/dat	*unterm/s*	*darunter*	
15	*um*	9880	prim.	acc	*ums*	*darum*	
16	*vor*	9852	prim.	acc/dat	*vorm/s*	*davor*	
17	*zwischen*	5079	prim.	acc/dat		*dazwischen*	
18	*seit*	4194	sec.	dat		*(seitdem)*	(+ prep)
19	*pro*	4175	sec.	/			
20	*ohne*	3007	prim.	acc			
21	*neben*	2733	prim.	acc/dat		*daneben*	
22	*laut*	2438	sec.	dat			
23	*gegen*	2127	prim.	acc		*dagegen*	
24	*per*	2011	sec.	/			
25	*ab*	1884	sec.	acc/dat			
26	*gegenüber*	1707	sec.	dat			pre/post
27	*innerhalb*	1509	sec.	gen			(+ *von*)
28	*trotz*	1260	sec.	dat/gen		*(trotzdem)*	
29	*wegen*	1048	prim.	dat/gen		*(deswegen)*	pre/post
30	*aufgrund*	949	sec.	gen			(+ *von*)
31	*während*	747	sec.	dat/gen		*(w.-dessen)*	
32	*hinter*	721	prim.	acc/dat	*hinterm/s*	*dahinter*	
33	*for*	676	sec.				
34	*statt*	611	sec.	gen		*(s.-dessen)*	
35	*angesichts*	553	sec.	gen			(+ *von*)

rank	preposition	frequency	type	case	contr.	pron. adv	special
36	*außer*	446	sec.	dat		*(außerdem)*	(+ *von*)
37	*dank*	414	sec.	dat/gen			
38	*je*	390	sec.	/			
39	*mittels*	380	sec.	dat/gen			
40	*hinsichtlich*	354	sec.	gen			(+ *von*)
41	*namens*	341	sec.	gen			
42	*außerhalb*	310	sec.	gen			(+ *von*)
43	*inklusive*	293	sec.	gen			(+ *von*)
44	*einschließlich*	284	sec.	gen			(+ *von*)
45	*anhand*	258	sec.	gen			(+ *von*)
46	*samt*	164	sec.	dat			
47	*gemäß*	153	sec.	dat/gen			pre/post
48	*bezüglich*	148	sec.	gen			(+ *von*)
49	*zugunsten*	136	sec.	gen			(+ *von*)
50	*anläßlich*	132	sec.	gen			(+ *von*)
51	*binnen*	120	sec.	dat/gen			
52	*anstelle*	105	sec.	gen			(+ *von*)
53	*infolge*	103	sec.	gen		*(i.-dessen)*	(+ *von*)
54	*seitens*	95	sec.	gen			
55	*jenseits*	90	sec.	gen			(+ *von*)
56	*entgegen*	76	sec.	dat			
57	*entlang*	64	sec.	acc/gen			pre/post
58	*unterhalb*	58	sec.	gen			(+ *von*)
59	*anstatt*	56	sec.	gen			(+ *von*)
60	*nahe*	49	sec.	gen			
61	*mangels*	44	sec.	gen			
62	*seiten*	39	sec.	gen			*von/auf* +
63	*versus*	32	sec.	gen			
64	*nebst*	31	sec.	dat			
65	*wider*	26	sec.	acc			
66	*oberhalb*	23	sec.	gen			(+ *von*)
67	*ob*	21	sec.	gen		*darob*	
68	*mitsamt*	21	sec.	dat			
69	*ungeachtet*	20	sec.	gen			(+ *von*)
70	*abseits*	20	sec.	gen			(+ *von*)
71	*zuzüglich*	18	sec.	gen			(+ *von*)
72	*zwecks*	17	sec.	gen			
73	*ähnlich*	15	sec.	gen			
74	*inmitten*	12	sec.	gen			(+ *von*)
75	*eingangs*	9	sec.	gen			
76	*südlich*	8	sec.	gen			(+ *von*)

rank	preposition	frequency	type	case	contr.	pron. adv	special
77	*vorbehaltlich*	7	sec.	gen			(+ *von*)
78	*nördlich*	7	sec.	gen			(+ *von*)
79	*kontra*	6	sec.	gen			
80	*gen*	6	sec.	acc			
81	*entsprechend*	6	sec.	dat/gen			pre/post
82	*westlich*	5	sec.	gen			(+ *von*)
83	*fern*	5	sec.	gen			(+ *von*)
84	*abzüglich*	5	sec.	gen			(+ *von*)
85	*diesseits*	4	sec.	gen			(+ *von*)
86	*beiderseits*	4	sec.	gen			(+ *von*)
87	*zuungunsten*	3	sec.	gen			(+ *von*)
88	*unweit*	3	sec.	gen			(+ *von*)
89	*längs*	3	sec.	gen			(+ *von*)
90	*ausschließlich*	2	sec.	gen			(+ *von*)
91	*anfangs*	2	sec.	gen			
92	*vermittels*	1	sec.	gen			
93	*unbeschadet*	1	sec.	gen			(+ *von*)
94	*südöstlich*	1	sec.	gen			(+ *von*)
95	*seitlich*	1	sec.	gen			(+ *von*)
96	*östlich*	1	sec.	gen			(+ *von*)
97	*nordöstlich*	1	sec.	gen			(+ *von*)
98	*minus*	1	sec.	dat/gen			
99	*kraft*	1	sec.	gen			
100	*exklusive*	1	sec.	gen			(+ *von*)

Appendix B

Contracted Prepositions in the Computer-Zeitung Corpus

This appendix lists all contracted prepositions of the Computer-Zeitung (1993-95+1997). The table includes contracted forms for the prepositions *an, auf, bei, durch, für, hinter, in, über, um, unter, von, vor, zu*. In order to illustrate the usage tendency we added the frequencies for the non-contracted forms.

rank	contracted prep.	frequency	prep. + det.	frequency	prep. + det.	frequency
1	*im*	40940	*in dem*	857	*in einem*	2365
2	*zum*	14225	*zu dem*	330	*zu einem*	1578
3	*zur*	13537	*zu der*	219	*zu einer*	986
4	*vom*	6299	*von dem*	534	*von einem*	1061
5	*am*	6136	*an dem*	442	*an einem*	506
6	*beim*	4641	*bei dem*	551	*bei einem*	759
7	*ins*	2155	*in das*	1053	*in ein*	521
8	*ans*	199	*an das*	611	*an ein*	171
9	*fürs*	154	*für das*	3787	*für ein*	879
10	*aufs*	125	*auf das*	1281	*auf ein*	600
11	*übers*	109	*über das*	1598	*über ein*	684
12	*ums*	60	*um das*	302	*um ein*	372
13	*durchs*	53	*durch das*	645	*durch ein*	373
14	*unterm*	36	*unter dem*	1062	*unter einem*	102
15	*unters*	10	*unter das*	27	*unter ein*	6
16	*vors*	4	*vor das*	20	*vor ein*	44
17	*hinterm*	4	*hinter dem*	102	*hinter einem*	5
18	*überm*	2	*über dem*	142	*über einem*	50
19	*vorm*	1	*vor dem*	598	*vor einem*	263
20	*hinters*	1	*hinter das*	3	*hinter ein*	0

Appendix C

Pronominal Adverbs in the Computer-Zeitung Corpus

This appendix lists all pronomial adverbs of the Computer-Zeitung (1993-95+1997) sorted by the cumulated frequency of the corresponding preposition.

rank	prep.	freq.	*da*-form	freq.	*hier*-form	freq.	*wo*-form	freq
1	*bei*	6929	*dabei*	5861	*hierbei*	381	*wobei*	687
2	*mit*	6446	*damit*	6332	*hiermit*	36	*womit*	78
3	*zu*	3508	*dazu*	3099	*hierzu*	348	*wozu*	61
4	*für*	2767	*dafür*	2410	*hierfür*	309	*wofür*	48
5	*von*	1777	*davon*	1708	*hiervon*	20	*wovon*	49
6	*über*	1783	*darüber*	1766	*hierüber*	5	*worüber*	12
7	*durch*	1601	*dadurch*	1385	*hierdurch*	54	*wodurch*	162
8	*gegen*	1420	*dagegen*	1397	*hiergegen*		*wogegen*	23
9	*auf*	1324	*darauf*	1267	*hierauf*	19	*worauf*	38
10	*an*	789	*daran*	737	*hieran*	9	*woran*	43
11	*in*	738	*darin*	685	*hierin*	18	*worin*	35
12	*nach*	613	*danach*	531	*hiernach*	3	*wonach*	79
13	*unter*	601	*darunter*	587	*hierunter*	6	*worunter*	8
14	*aus*	463	*daraus*	432	*hieraus*	18	*woraus*	13
15	*um*	377	*darum*	367	*hierum*		*worum*	10
16	*neben*	331	*daneben*	331	*hierneben*		*woneben*	
17	*vor*	148	*davor*	146	*hiervor*		*wovor*	2
18	*hinter*	135	*dahinter*	135	*hierhinter*		*wohinter*	
19	*zwischen*	26	*dazwischen*	26	*hierzwischen*		*wozwischen*	

All primary prepositions are represented except for *ohne* and *wegen*. Queries to an internet search engine[1] reveal that pronominal adverb forms for *wegen* do exist albeit with low frequencies (*dawegen* 8, *hierwegen* 82, *wowegen* 3!). The internet search engine also finds examples for those forms with zero frequency in the Computer-Zeitung (*hiergegen* being by far the most frequent form).

[1] We used `www.google.com`.

Special forms

There are a number of special forms that can be regarded as pronominal adverbs or as related to them. First, there is the pronominal adverb *darob* which sounds rather old-fashioned. In the second block we list combinations of pronominal adverbs with the particle *hin* (*daraufhin* and *woraufhin*) which can be considered frozen circumpositional phrases since the particle is a typical right element of a circumposition.

The third block lists pronominal adverb forms with a vowel ellision in the first syllable. And the final block lists combinations of prepositions (or postpositions) with a form of the definite determiner (or of the corresponding interrogative form *wes*) which were marked as pronominal adverbs by our part-of-speech tagger. Since all of them serve other functions as well (as adverb or conjunction), the frequency counts are not very reliable and should not be taken as giving more than a rough idea of their usage.

pronominal adverb	frequency
darob	1
daraufhin	138
woraufhin	1
dran	14
drauf	32
draus	1
drin	13
drum	3
drunter	2
außerdem	2020
dementsprechend	100
demgegenüber	53
demgemäß	1
demnach	21
demzufolge	52
deshalb	2127
deswegen	94
infolgedessen	4
seitdem	127
stattdessen	16
trotzdem	570
währenddessen	16
weshalb	88
weswegen	2

Appendix D

Reciprocal Pronouns in the Computer-Zeitung Corpus

This appendix lists all prepositional reciprocal pronouns of the Computer-Zeitung (1993-95+1997). The table includes the pure pronoun *einander* (rank 7).

rank	reciprocal pronoun	frequency
1	*miteinander*	609
2	*untereinander*	187
3	*voneinander*	161
4	*aufeinander*	91
5	*auseinander*	66
6	*nebeneinander*	58
7	*einander*	47
8	*zueinander*	43
9	*gegeneinander*	37
10	*hintereinander*	28
11	*nacheinander*	20
12	*durcheinander*	14
13	*aneinander*	13
14	*ineinander*	12
15	*beieinander*	12
16	*übereinander*	7
17	*füreinander*	1

Five primary prepositions do not have reciprocal pronouns in this corpus. But for all of them we find usage examples in the internet (with *wegeneinander* being the least frequent).

(D.1) *Nach langen Streitereien stellen sie fest, dass sie* ***ohneeinander*** *nicht leben wollen ...*

(D.2) *Wie fünf Sterne, die* ***umeinander*** *kreisen.*

(D.3) *Zusammenleben in Achtung* ***voreinander****.*

(D.4) *... auf diese Weise die pädagogische Tagesarbeit miteinander und* ***wegeneinander*** *zu vertiefen*

(D.5) *Konkret, handelt es sich um eine "Brücke", die den zwei Applikationen* ***zwischeneinander*** *oder mit einem Hardwareelement zu komunizieren erlaubt.*

Bibliography

[Abney et al. 1999] Steven P. Abney, Robert E. Schapire, and Yoram Singer. 1999. Boosting applied to tagging and PP attachment. In Pascale Fung and Joe Zhou, editors, *Proc. of Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*, pages 38–45, University of Maryland, College Park.

[Abney 1989] Stephen Paul Abney. 1989. A computational model of human parsing. *Journal of Psycholinguistic Research*, 18(1):129–144.

[Abney 1997] Steven P. Abney. 1997. Stochastic attribute-value grammars. *Computational Linguistics*, 23(4):597–618.

[Agricola 1968] Erhard Agricola. 1968. *Syntaktische Mehrdeutigkeit (Polysyntaktizität) bei der Analyse des Deutschen und des Englischen.* Schriften zur Phonetik, Sprachwissenschaft und Kommunikationsforschung. Akademie Verlag, Berlin.

[Alegre et al. 1999] M.A. Alegre, J.M. Sopena, and A. Lloberas. 1999. PP-attachment: A committee machine approach. In Pascale Fung and Joe Zhou, editors, *Proc. of Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*, pages 231–238, University of Maryland, College Park.

[Aliod et al. 1998] Diego Mollá Aliod, Jawad Berri, and Michael Hess. 1998. A real world implementation of answer extraction. In *Proc. of 9th International Conference and Workshop on Database and Expert Systems. Workshop "Natural Language and Information Systems" (NLIS'98)*, Vienna.

[Arnold et al. 2001] T. Arnold, S. Clematide, R. Nespeca, J. Roth, and M. Volk. 2001. LUIS - ein natürlichsprachliches, universitäres Informationssystem. In H.-J. Appelrath, R. Beyer, U. Marquardt, H.C. Mayr, and C. Steinberger, editors, *Proc. of "Unternehmen Hochschule" (Symposium UH 2001)*, volume P-6 of *Lecture Notes in Informatics (LNI) - Proceedings*, pages 115–126, Wien.

[Baayen et al. 1995] R. H. Baayen, R. Piepenbrock, and H. van Rijn. 1995. The CELEX lexical database (CD-ROM). Linguistic Data Consortium, University of Pennsylvania.

[Behl 1999] Heike K. Behl. 1999. Word order and preposition attachment in English-German MT systems. In Claudia Gdaniec, editor, *Problems and Potential of English-to-German MT systems. Workshop at the 8th International Conference on Theoretical and Methodological Issues in Machine Translation. TMI-99*, Chester.

[Biber et al. 1998] D. Biber, S. Conrad, and R. Reppen. 1998. *Corpus Linguistics. Investigating Language Structure and Use.* Cambridge University Press.

[Black et al. 1993] Ezra Black, Roger Garside, and Geoffrey Leech, editors. 1993. *Statistically-driven computer grammars of English: The IBM/Lancaster approach.* Language and Computers. Rodopi, Amsterdam.

[Boland 1998] Julie E. Boland. 1998. Lexical constraints and prepositional phrase attachment. *Journal of Memory and Language*, 39(4):684–719.

[Bowen 2001] Rhonwen Bowen. 2001. Nouns and their prepositional phrase complements in English. In *Proc. of Corpus Linguistics*, Lancaster.

[Brants et al. 1997] T. Brants, W. Skut, and B. Krenn. 1997. Tagging grammatical functions. In *Proc. of EMNLP-2*, Providence, RI.

[Breindl 1989] Eva Breindl. 1989. *Präpositionalobjekte und Präpositionalobjektsätze im Deutschen*, volume 220 of *Linguistische Arbeiten*. Niemeyer, Tübingen.

[Brill and Resnik 1994] E. Brill and P. Resnik. 1994. A rule-based approach to prepositional phrase attachment disambiguation. In *Proceedings of COLING*, pages 1198–1204, Kyoto. ACL.

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

[Britt 1994] M. Anne Britt. 1994. The interaction of referential ambiguity and argument structure in the parsing of prepositional phrases. *Journal of Memory and Language*, 33:251–283.

[Burke et al. 1997] R.D. Burke, K.J. Hammond, V.A. Kulyukin, S.L. Lytinen, N. Tomuro, and S. Schoenberg. 1997. Question answering from frequently-asked question files: Experiences with the FAQ finder system. Technical Report TR-97-05, The University of Chicago. Computer Science Department.

[Bußmann 1990] Hadumod Bußmann. 1990. *Lexikon der Sprachwissenschaft*. Kröner Verlag, Stuttgart, 2. revised edition.

[Carbonell and Hayes 1987] Jaime G. Carbonell and Philip J. Hayes. 1987. Robust parsing using multiple construction-specific strategies. In Leonard Bolc, editor, *Natural Language Parsing Systems*, pages 1–32. Springer, Berlin.

[Chen and Chang 1995] Mathis H.C. Chen and Jason J.S. Chang. 1995. Structural ambiguity and conceptual information retrieval. In *PACLIC-10, Kowloon*, Hongkong.

[Chen and Chen 1996] Kuang-Hua Chen and Hsin-Hsi Chen. 1996. A rule-based and MT-oriented approach to prepositional phrase attachment. In *Proc. of COLING-96*, pages 216–221, Copenhagen.

[Clematide and Volk 2001] Simon Clematide and Martin Volk. 2001. Linguistische und semantische Annotation eines Zeitungskorpus. In *Proc. of GLDV-Jahrestagung*, Giessen, March.

[Collins and Brooks 1995] Michael Collins and James Brooks. 1995. Prepositional phrase attachment through a backed-off model. In *Proc. of the Third Workshop on Very Large Corpora*.

[Crain and Steedman 1985] S. Crain and M. Steedman. 1985. On not being led up the garden path: the use of context by the psychological syntax processor. In D.R. Dowty, L. Karttunen, and A.M. Zwicky, editors, *Natural language parsing. Psychological, computational, and theoretical perspectives*, chapter 9, pages 320–358. Cambridge University Press, Cambridge.

[Cucchiarelli et al. 1999] A. Cucchiarelli, D. Luzi, and P. Velardi. 1999. Semantic tagging of unknown proper nouns. *Natural Language Engineering*, 5:171–185.

[Dahlgren 1988] K. Dahlgren. 1988. *Naive Semantics for Natural Language Understanding*. Kluwer, Boston.

[de Lima 1997] Erika F. de Lima. 1997. Acquiring German prepositional subcategorization frames from corpora. In J. Zhou and K. Church, editors, *Proc. of the Fifth Workshop on Very Large Corpora*, pages 153–167, Beijing and Hongkong.

[Domenig and ten Hacken 1992] M. Domenig and P. ten Hacken. 1992. *Word Manager: A system for morphological dictionaries*. Olms Verlag, Hildesheim.

[Drosdowski 1995] Günther Drosdowski, editor. 1995. *DUDEN. Grammatik der deutschen Gegenwartssprache*. Bibliographisches Institut, Mannheim, 5. edition.

[Ejerhed 1996] Eva Ejerhed. 1996. Finite state segmentation of discourse into clauses. In A. Kornai, editor, *ECAI Workshop: Extended Finite State Models of Language*.

[Eroms 1981] Hans-Werner Eroms. 1981. *Valenz, Kasus und Präpositionen. Untersuchungen zur Syntax und Semantik präpositionaler Konstruktionen in der deutschen Gegenwartssprache*. Carl Winter, Heidelberg.

[Fang 2000] Alex Chengyu Fang. 2000. A lexicalist approach towards the automatic determination for the syntactic functions of prepositional phrases. *Natural Language Engineering*, 6:183–201.

[Fleischer and Barz 1995] W. Fleischer and I. Barz. 1995. *Wortbildung der deutschen Gegenwartssprache*. Niemeyer, Tübingen, 2. edition.

[Franz 1996a] Alex Franz. 1996a. *Automatic Ambiguity Resolution in Natural Language Processing. An Empirical Approach*, volume 1171 of *Lecture Notes in Artificial Intelligence*. Springer, Berlin.

[Franz 1996b] Alex Franz. 1996b. Learning PP attachment from corpus statistics. In S. Wermter, E. Riloff, and G. Scheler, editors, *Connectionist, Statistical, and Symbolic Approaches to Learning for Natural Language Processing*, volume 1040 of *Lecture Notes in AI*, pages 188–202. Springer Verlag, Berlin.

[Frazier 1978] L. Frazier. 1978. *On comprehending sentences: syntactic parsing strategies*. PhD dissertation, University of Connecticut.

[Gale et al. 1992] W.A. Gale, K.W. Church, and D. Yarowsky. 1992. One sense per discourse. In *Proc. of DARPA speech and Natural Language Workshop*, Harriman, NY, February.

[Gaussier and Cancedda 2001] Eric Gaussier and Nicola Cancedda. 2001. Probabilistic models for PP-attachment resolution and NP analysis. In *Proc. of ACL-2001 CoNLL-2001 Workshop*, Toulouse. ACL.

[Gazdar et al. 1985] Gerald Gazdar, Ewan Klein, Geoffrey Pullum, and Ivan Sag. 1985. *Generalized phrase structure grammar*. Harvard University Press, Cambridge, MA.

[Götz et al. 1993] D. Götz, G. Hänsch, and H. Wellmann, editors. 1993. *Langenscheidts Großwörterbuch Deutsch als Fremdsprache*. Langenscheidt, Berlin.

[Greenbaum 1996] Sidney Greenbaum. 1996. *The Oxford English Grammar*. Oxford University Press.

[Grefenstette 1999] Gregory Grefenstette. 1999. The World Wide Web as a resource for example-based machine translation tasks. In *Proc. of Aslib Conference on Translating and the Computer 21*, London, November.

[Griesbach and Uhlig 1994] H. Griesbach and G. Uhlig. 1994. *Die starken Verben im Sprachgebrauch. Syntax - Valenz - Kollokationen*. Langenscheidt, Leipzig.

[Griesbach 1986] Heinz Griesbach. 1986. *Neue deutsche Grammatik*. Langenscheidt, Berlin.

[Haapalainen and Majorin 1994] Mariikka Haapalainen and Ari Majorin, 1994. *Gertwol. Ein System zur automatischen Wortformerkennung deutscher Wörter*. Lingsoft Oy, Helsinki, September.

[Hanrieder 1996] Gerhard Hanrieder. 1996. PP-Anbindung in einem kognitiv adäquaten Verarbeitungsmodell. In S. Mehl, A. Mertens, and M. Schulz, editors, *Präpositionalsemantik und PP-Anbindung*, number SI-16 in Schriftenreihe Informatik, pages 13–22. Gerhard-Mercator-Universität, Duisburg.

[Hartrumpf 1999] Sven Hartrumpf. 1999. Hybrid disambiguation of prepositional phrase attachment and interpretation. In Pascale Fung and Joe Zhou, editors, *Proc. of Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*, pages 111–120, University of Maryland, College Park.

[Heid 1999] Ulrich Heid. 1999. Extracting terminologically relevant collocations from German technical texts. In *Proceedings of the TKE '99 International Congress on Terminology and Knowledge Engineering*, pages 241 – 255, Innsbruck.

[Helbig and Buscha 1998] G. Helbig and J. Buscha. 1998. *Deutsche Grammatik. Ein Handbuch für den Ausländerunterricht.* Langenscheidt. Verlag Enzyklopädie, Leipzig, Berlin, 18. edition.

[Helbig and Schenkel 1991] G. Helbig and W. Schenkel. 1991. *Wörterbuch zur Valenz und Distribution deutscher Verben.* Niemeyer, Tübingen, 8 edition.

[Helbig et al. 1994] H. Helbig, A. Mertens, and M. Schulz. 1994. Disambiguierung mit Wortklassenagenten. Informatik-Bericht 168, Fernuniversität Hagen.

[Hindle and Rooth 1993] D. Hindle and M. Rooth. 1993. Structural ambiguity and lexical relations. *Computational Linguistics*, 19(1):103–120.

[Hirst 1987] Graeme Hirst. 1987. *Semantic interpretation and the resolution of ambiguity.* Studies in Natural Language Processing. Cambridge University Press, Cambridge, New York.

[Hoeppner 1980] Wolfgang Hoeppner. 1980. *Derivative Wortbildung der deutschen Gegenwartssprache und ihre algorithmische Analyse.* Gunter Narr Verlag, Tübingen.

[Jaworska 1999] E. Jaworska. 1999. Prepositions and prepositional phrases. In K. Brown and J. Miller, editors, *Concise Encyclopedia of Grammatical Categories*, pages 304–311. Elsevier, Amsterdam.

[Jensen and Binot 1987] K. Jensen and J.-L. Binot. 1987. Disambiguating prepositional phrase attachments by using on-line dictionary definitions. *Computational Linguistics*, 13(3-4):251–260.

[Kennedy 1998] Graeme Kennedy. 1998. *An Introduction to Corpus Linguistics.* Addison Wesley Longman, London.

[Kermes and Evert 2001] H. Kermes and St. Evert. 2001. Exploiting large corpora: A circular process of partial syntactic analysis, corpus query and extraction of lexicographic information. In *Proc. of Corpus Linguistics*, Lancaster.

[Kimball 1973] J. Kimball. 1973. Seven principles of surface structure parsing in natural language. *Cognition*, 2:15–47.

[Klaus 1999] Cäcilia Klaus. 1999. *Grammatik der Präpositionen: Studien zur Grammatikographie; mit einer thematischen Bibliographie*, volume 2 of *Linguistik International.* Peter Lang, Frankfurt.

[Konieczny et al. 1991] L. Konieczny, B. Hemforth, and G. Strube. 1991. Psychologisch fundierte Prinzipien der Satzverarbeitung jenseits von Minimal Attachment. *Kognitionswissenschaft*, 1(2):58–70.

[Konradin-Verlag 1998] Konradin-Verlag. 1998. Computer Zeitung auf CD-ROM. Volltextrecherche aller Artikel der Jahrgänge 1993 bis 1998. Leinfelden-Echterdingen.

[Krenn and Evert 2001] B. Krenn and St. Evert. 2001. Can we do better than frequency? A case study on extracting PP-verb collocations. In *Proc. of the ACL Workshop on Collocation*, Toulouse.

[Krenn and Volk 1993] Brigitte Krenn and Martin Volk. 1993. DiTo-Datenbank. Datendokumentation zu Funktionsverbgefügen und Relativsätzen. DFKI-Document D-93-24, DFKI, Saarbrücken.

[Krenn 2000] Brigitte Krenn. 2000. Collocation Mining: Exploiting Corpora for Collocation, Identification and Representation. In *Proc. of Konvens-2000. Sprachkommunikation*, pages 209–214, Ilmenau. VDE Verlag.

[Langer et al. 1997] H. Langer, S. Mehl, and M. Volk. 1997. Hybride NLP-Systeme und das Problem der PP-Anbindung. In S. Busemann, K. Harbusch, and S. Wermter, editors, *Berichtsband des Workshops "Hybride konnektionistische, statistische und symbolische Ansätze zur Verarbeitung natürlicher Sprache" auf der 21. Deutschen Jahrestagung für Künstliche Intelligenz, KI-97 (auch erschienen als DFKI-Document D-98-03)*, Freiburg.

[Langer 1996] Hagen Langer. 1996. Disambiguierung von Präpositionalkonstruktionen mit einem syntaktischen Parser: Möglichkeiten und Grenzen. In S. Mehl, A. Mertens, and M. Schulz, editors, *Präpositionalsemantik und PP-Anbindung*, number SI-16 in Schriftenreihe Informatik, pages 23–31. Gerhard-Mercator-Universität, Duisburg.

[Langer 1999] Hagen Langer. 1999. *Parsing-Experimente*. Habilitationsschrift, Universität Osnabrück, Januar.

[Lemnitzer 1997] Lothar Lemnitzer. 1997. *Akquisition komplexer Lexeme aus Textkorpora*, volume 180 of *Germanistische Linguistik*. Niemeyer, Tübingen.

[Li and Abe 1998] Hang Li and Naoki Abe. 1998. Generalizing case frames using a thesaurus and the MDL principle. *Computational Linguistics*, 24(2):217–244.

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

[Mani and MacMillan 1996] Inderjeet Mani and T. Richard MacMillan. 1996. Identifying unknown proper names in newswire text. In B. Boguraev and J. Pustejovsky, editors, *Corpus Processing for Lexical Acquisition*, chapter 3, pages 41–59. MIT Press, Cambridge, MA.

[Manning and Schütze 2000] C. Manning and H. Schütze. 2000. *Foundations of Statistical Natural Language Processing*. MIT Press, Cambridge, MA, second printing with corrections edition.

[Mason 2000] Oliver Mason. 2000. *Java Programming for Corpus Linguistics*. Edinburgh Textbooks in Empirical Linguistics. Edinburgh University Press, Edinburgh.

[Mater 1969] Erich Mater. 1969. *Verhältnis zum Reflexivpronomen und Kompositionsbildung zu Grundwörtern*, volume 7 of *Deutsche Verben*. VEB Bibliographisches Institut, Leipzig.

[McDonald 1996] David D. McDonald. 1996. Internal and external evidence in the identification and semantic categorization of proper names. In B. Boguraev and J. Pustejovsky, editors, *Corpus Processing for Lexical Acquisition*, chapter 2, pages 21–39. MIT Press, Cambridge, MA.

[Mehl et al. 1996] S. Mehl, A. Mertens, and M. Schulz, editors. 1996. *Präpositionalsemantik und PP-Anbindung*. Number SI-16 in Schriftenreihe Informatik. Gerhard-Mercator-Universität, Duisburg.

[Mehl et al. 1998] S. Mehl, H. Langer, and M. Volk. 1998. Statistische Verfahren zur Zuordnung von Präpositionalphrasen. In B. Schröder, W. Lenders, W. Hess, and T. Portele, editors, *Computers, Linguistics, and Phonetics between Language and Speech. Proc. of the 4th Conference on Natural Language Processing. KONVENS-98*, pages 97–110, Bonn. Peter Lang. Europäischer Verlag der Wissenschaften.

[Mehl 1998] Stephan Mehl. 1998. Semantische und syntaktische Disambiguierung durch fakultative Verbkomplemente. In Petra Ludewig and Bart Geurts, editors, *Lexikalische Semantik aus kognitiver Sicht*. Narr Verlag, Tübingen.

[Meier 1964] H. Meier. 1964. *Deutsche Sprachstatistik*. Georg Olms Verlag, Hildesheim.

[Merlo et al. 1997] P. Merlo, M.W. Crocker, and C. Berthouzoz. 1997. Attaching multiple prepositional phrases: generalized backed-off estimation. In *Proceedings of the Second Conference on Empirical Methods in Natural Language Processing. Brown University, RI.*

[Meyer 1989] Kurt Meyer. 1989. *Wie sagt man in der Schweiz? Wörterbuch der schweizerischen Besonderheiten*. Duden Taschenbücher. Dudenverlag, Mannheim.

[Miller 1995] George A. Miller. 1995. WordNet: A lexical database for English. *CACM*, 38(11):39–41.

[MUC 1998] 1998. Message understanding conference proceedings: MUC7. http://www.muc.saic.com.

[Müller 1999] Stefan Müller. 1999. *Deutsche Syntax deklarativ. Head-Driven Phrase Structure Grammar für das Deutsche*, volume 394 of *Linguistische Arbeiten*. Niemeyer Verlag, Tübingen.

[Murray 1995] K. M. Elisabeth Murray. 1995. *Caught in the Web of Words: James Murray and The Oxford English Dictionary.* Yale University Press.

[Negra-Group 2000] Negra-Group. 2000. Negr@ corpus. A syntactically annotated Corpus of German Newspaper Texts. Saarland-University. Department of Computational Linguistics and Phonetics. http://www.coli.uni-sb.de/sfb378/negra-corpus/.

[Oakes 1998] Michael Oakes. 1998. *Statistics for Corpus Linguistics.* Edinburgh Textbooks in Empirical Linguistics. Edinburgh University Press, Edinburgh.

[Paik et al. 1996] W. Paik, E.D. Liddy, E. Yu, and M. McKenna. 1996. Categorizing and standardizing proper nouns for efficient information retrieval. In B. Boguraev and J. Pustejovsky, editors, *Corpus Processing for Lexical Acquisition*, chapter 4, pages 61–73. MIT Press, Cambridge, MA.

[Pantel and Lin 2000] Patrick Pantel and Dekang Lin. 2000. An unsupervised approach to prepositional phrase attachment using contextually similar words. In *Proc. of ACL-2000*, Hongkong.

[Piskorski and Neumann 2000] J. Piskorski and G. Neumann. 2000. An intelligent text extraction and navigation system. In *Proc. of 6th International Conference on Computer-Assisted Information Retrieval (RIAO-2000)*, Paris, April.

[Pollard and Sag 1994] Carl Pollard and Ivan Sag. 1994. *Head-Driven Phrase Structure Grammar.* University of Chicago Press, Chicago.

[Ratnaparkhi et al. 1994] A. Ratnaparkhi, J. Reynar, and S. Roukos. 1994. A maximum entropy model for prepositional phrase attachment. In *Proceedings of the ARPA Workshop on Human Language Technology*, Plainsboro, NJ, March.

[Ratnaparkhi 1998] Adwait Ratnaparkhi. 1998. Statistical models for unsupervised prepositional phrase attachment. In *Proceedings of COLING-ACL-98*, Montreal.

[Resnik 1993] Philip Resnik. 1993. *Selection and Information: A Class-Based Approach to Lexical Relationships.* Ph.D. thesis, University of Pennsylvania, December. (Institute for Research in Cognitive Science report IRCS-93-42); includes a chapter on PP attachment.

[Richter and Sailer 1996] F. Richter and M. Sailer. 1996. Syntax für eine unterspezifizierte Semantik: PP-Anbindung in einem deutschen HPSG-Fragment. In S. Mehl, A. Mertens, and M. Schulz, editors, *Präpositionalsemantik und PP-Anbindung*, number SI-16 in Schriftenreihe Informatik, pages 39–47. Gerhard-Mercator-Universität, Duisburg.

[Roth 1998] Dan Roth. 1998. Learning to resolve natural language ambiguities: A unified approach. In *Proc. of AAAI-98.*

[Roth 2001] Jeannette Roth. 2001. Automatische Erkennung von Produktenamen. Programmierprojekt, Universität Zürich.

[Schaeder 1998] Burkhard Schaeder. 1998. Die Präpositionen in Langenscheidts Großwörterbuch Deutsch als Fremdsprache. In Herbert E. Wiegand, editor, *Perspektiven der pädagogischen Lexikographie des Deutschen. Untersuchungen anhand von "Langenscheidts Großwörterbuch Deutsch als Fremdsprache"*, volume 86 of *Lexicographica. Series Maior*, pages 208–232. Niemeyer Verlag, Tübingen.

[Schäuble 1997] Peter Schäuble. 1997. *Multimedia Information Retrieval. Content-based Information Retrieval from Large Text and Audio Databases.* Kluwer Academic Publishers, Boston.

[Schierholz 2001] Stefan J. Schierholz. 2001. *Präpositionalattribute. Syntaktische und semantische Analysen*, volume 447 of *Linguistische Arbeiten.* Niemeyer Verlag, Tübingen.

[Schiller et al. 1995] 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.

[Schmid and Kempe 1996] H. Schmid and A. Kempe. 1996. Tagging von Korpora mit HMM, Entscheidungsbäumen und Neuronalen Netzen. In H. Feldweg and E.W. Hinrichs, editors, *Wiederverwendbare Methoden und Ressourcen zur linguistischen Erschliessung des Deutschen*, volume 73 of *Lexicographica. Series Maior*, pages 231–244. Niemeyer Verlag, Tübingen.

[Schmied and Fink 1999] J. Schmied and B. Fink. 1999. Corpus-based contrastive lexicology: the case of English *with* and its German translation equivalents. In S.P. Botley, A.M. McEnery, and A. Wilson, editors, *Multilingual Corpora in Teaching and Research*, chapter 11, pages 157–176. Rodopi, Amsterdam.

[Schröder 1990] Jochen Schröder. 1990. *Lexikon deutscher Präpositionen.* Verlag Enzyklopädie, Leipzig.

[Schulz et al. 1995] M. Schulz, A. Mertens, and H. Helbig. 1995. Dynamische Präpositionsanalyse mit Hilfe von Lexikon und Wortagenten im System LINAS. In James Kilbury and Richard Wiese, editors, *Integrative Ansätze in der Computerlinguistik*, pages 96–101, Universität Düsseldorf.

[Schulz et al. 1997] M. Schulz, A. Mertens, and H. Helbig. 1997. Präpositionsanalyse im System LINAS. In D. Haumann and S.J. Schierholz, editors, *Lexikalische und grammatische Eigenschaften präpositionaler Elemente*, volume 371 of *Linguistische Arbeiten*, pages 105–121, Tübingen. Niemeyer.

[Schumacher 1986] Helmut Schumacher, editor. 1986. *Verben in Feldern. Valenzwörterbuch zur Syntax und Semantik deutscher Verben.* Walter de Gruyter Verlag, Berlin.

[Schütze 1995] Carson T. Schütze. 1995. PP-attachment and argumenthood. Technical Report 26, MIT Working Papers in Linguistics.

[Schütze 1997] Hinrich Schütze. 1997. *Ambiguity Resolution in Language Learning: computational and cognitive models*, volume 71 of *Lecture Notes.* CSLI, Stanford.

[Schweisthal 1971] Klaus G. Schweisthal. 1971. *Präpositionen in der maschinellen Sprachbearbeitung. Methoden der maschinellen Inhaltsanalyse und der Generierung von Präpositionalphrasen, insbesondere für reversible Maschinenübersetzung.* Dümmler, Bonn.

[Skut and Brants 1998] Wojciech Skut and Thorsten Brants. 1998. A maximum-entropy partial parser for unrestricted text. In *Proc. of Sixth Workshop on Very Large Corpora*, Montréal.

[Skut et al. 1997] W. Skut, B. Krenn, T. Brants, and H. Uszkoreit. 1997. An annotation scheme for free word order languages. In *Proceedings of the 5th Conference on Applied Natural Language Processing*, pages 88–95, Washington, DC.

[Skut et al. 1998] Wojciech Skut, Thorsten Brants, Brigitte Krenn, and Hans Uszkoreit. 1998. A linguistically interpreted corpus of German newspaper text. In *Proc. of ESSLLI-98 Workshop on Recent Advances in Corpus Annotation*, Saarbrücken.

[Small and Rieger 1982] Steven L. Small and Chuck Rieger. 1982. Parsing and comprehending with word experts (a theory and its realization). In Wendy G. Lehnert and Martin H. Ringle, editors, *Strategies for Natural Language Processing*, pages 89–147. Lawrence Erlbaum, Hillsdale.

[Springer 1987] Danuta Springer. 1987. *Valenz der Verben mit präpositionalem Objekt "von", "mit": eine kontrastive Studie.* Wydawnictwo Wyzszej Szkoly Pedagogicznej, Zielona Gora.

[Stetina and Nagao 1997] J. Stetina and M. Nagao. 1997. Corpus based PP attachment ambiguity resolution with a semantic dictionary. In J. Zhou and K. Church, editors, *Proc. of the 5th Workshop on Very Large Corpora*, pages 66–80, Beijing and Hongkong.

[Stevenson and Gaizauskas 2000] Mark Stevenson and R. Gaizauskas. 2000. Using corpus-derived name lists for named entity recognition. In *Proc. of ANLP*, Seattle.

[Thielen and Schiller 1996] C. Thielen and A. Schiller. 1996. Ein kleines und erweitertes Tagset fürs Deutsche. In H. Feldweg and E.W. Hinrichs, editors, *Wiederverwendbare Methoden und Ressourcen zur linguistischen Erschliessung des Deutschen*, volume 73 of *Lexicographica. Series Maior*, pages 193–203. Niemeyer Verlag, Tübingen.

[Uszkoreit 1987] Hans Uszkoreit. 1987. *Word order and constituent structure in German.* Number 8 in CSLI, Lecture notes. University of Chicago Press, Stanford.

[Volk and Clematide 2001] Martin Volk and Simon Clematide. 2001. Learn-filter-apply-forget. Mixed approaches to named entity recognition. In Ana M. Moreno and Reind P. van de Riet, editors, *Applications of Natural Language for Information Systems. Proc. of 6th International Workshop NLDB'01*, volume P-3 of *Lecture Notes in Informatics (LNI) - Proceedings*, pages 153–163, Madrid.

[Volk and Richarz 1997] Martin Volk and Dirk Richarz. 1997. Experiences with the GTU grammar development environment. In D. Estival, A. Lavelli, K. Netter, and F. Pianesi, editors, *Workshop on Computational Environments for Grammar Development and Linguistic Engineering at the ACL/EACL Joint Conference*, pages 107–113, Madrid.

[Volk and Schneider 1998] Martin Volk and Gerold Schneider. 1998. Comparing a statistical and a rule-based tagger for German. In B. Schröder, W. Lenders, W. Hess, and T. Portele, editors, *Computers, Linguistics, and Phonetics between Language and Speech. Proc. of the 4th Conference on Natural Language Processing - KONVENS-98*, pages 125–137, Bonn.

[Volk et al. 1995] M. Volk, M. Jung, and D. Richarz. 1995. GTU - A workbench for the development of natural language grammars. In *Proc. of the Conference on Practical Applications of Prolog*, pages 637–660, Paris.

[Volk 1992] Martin Volk. 1992. The role of testing in grammar engineering. In *Proceedings of the 3rd Conference on Applied Natural Language Processing*, pages 257–258, Trento, Italy.

[Volk 1995] Martin Volk. 1995. *Einsatz einer Testsatzsammlung im Grammar Engineering*, volume 30 of *Sprache und Information.* Niemeyer Verlag, Tübingen.

[Volk 1996a] Martin Volk. 1996a. Die Rolle der Valenz bei der Auflösung von PP-Mehrdeutigkeiten. In S. Mehl, A. Mertens, and M. Schulz, editors, *Präpositionalsemantik und PP-Anbindung*, number SI-16 in Schriftenreihe Informatik, pages 32–38. Gerhard-Mercator-Universität, Duisburg.

[Volk 1996b] Martin Volk. 1996b. Parsing with ID/LP and PS rules. In D. Gibbon, editor, *Natural Language Processing and Speech Technology. Results of the 3rd KONVENS Conference (Bielefeld)*, pages 342–353, Berlin. Mouton de Gruyter.

[Volk 1998] Martin Volk. 1998. Markup of a test suite with SGML. In John Nerbonne, editor, *Linguistic Databases*, volume 77 of *CSLI Lecture Notes*, pages 59–76. CSLI.

[Volk 1999] Martin Volk. 1999. Choosing the right lemma when analysing German nouns. In *Multilinguale Corpora: Codierung, Strukturierung, Analyse. 11. Jahrestagung der GLDV*, pages 304–310, Frankfurt. Enigma Corporation.

[Volk 2000] Martin Volk. 2000. Scaling up. Using the WWW to resolve PP attachment ambiguities. In *Proc. of Konvens-2000. Sprachkommunikation*, pages 151–156, Ilmenau. VDE Verlag.

[Volk 2001] Martin Volk. 2001. Exploiting the WWW as a corpus to resolve PP attachment ambiguities. In *Proc. of Corpus Linguistics 2001*, Lancaster, March.

[Wahrig 1978] G. Wahrig, editor. 1978. *Der kleine Wahrig. Wörterbuch der deutschen Sprache.* Bertelsmann Lexikon Verlag, 1994 edition.

[Wauschkuhn 1999] Oliver Wauschkuhn. 1999. *Automatische Extraktion von Verbvalenzen aus deutschen Textkorpora.* Berichte aus der Informatik. Shaker, Aachen.

[Weisweber 1987] Wilhelm Weisweber. 1987. Ein Dominanz-Chart-Parser für generalisierte Phrasenstrukturgrammatiken. KIT Report 45, TU Berlin.

[Wilks and Stevenson 1997] Y.A. Wilks and M. Stevenson. 1997. Combining independent knowledge sources for word sense disambiguation. In *Proceedings of the Conference on Recent Advances in NLP*, Tzigov Lhask, Bulgaria.

[Wilks and Stevenson 1998] Y.A. Wilks and M. Stevenson. 1998. Word sense disambiguation using optimized combinations of knowledge sources. In *Proc. of ACL-COLING 98*, volume II, pages 1398–1402, Montréal.

[Winograd 1973] Terry Winograd. 1973. A procedural model of language processing. In R.C. Schank and K.M. Colby, editors, *Computer Models of Thought and Language*, pages 152–186. W.H. Freeman, San Francisco.

[Wu and Furugori 1996] H. Wu and T. Furugori. 1996. A hybrid disambiguation model for prepositional phrase attachment. *Literary and Linguistic Computing*, 11(4):187 – 192.

[Yeh and Vilain 1998] A.S. Yeh and M.B. Vilain. 1998. Some properties of preposition and subordinate conjunction attachments. In *Proceedings of COLING-ACL-98*, pages 1436–1442, Montreal.

[Zavrel et al. 1997] Jakub Zavrel, Walter Daelemans, and Jorn Veenstra. 1997. Resolving PP attachment ambiguities with memory-based learning. In Mark Ellison, editor, *Proc. of the Workshop on Computational Natural Language Learning*, Madrid. http://lcg-www.uia.ac.be/conll97/-proceedings.htm.

[Zifonun et al. 1997] Gisela Zifonun, Ludger Hoffmann, and Bruno Strecker. 1997. *Grammatik der deutschen Sprache*, volume 7 of *Schriften des Instituts für deutsche Sprache*. de Gruyter, Berlin.