SHALMANESER– A Toolchain For Shallow Semantic Parsing

Katrin Erk and Sebastian Padó

Computational Linguistics
Saarland University
Saarbrücken, Germany
erk,pado@coli.uni-sb.de

Abstract

This paper presents SHALMANESER, a software package for *shallow semantic parsing*, the automatic assignment of semantic classes and roles to free text. SHALMANESER is a toolchain of independent modules communicating through a common XML format. System output can be inspected graphically. SHALMANESER can be used either as a "black box" to obtain semantic parses for new datasets (classifiers for English and German frame-semantic analysis are included), or as a research platform that can be extended to new parsers, languages, or classification paradigms.

1. Introduction

The last decade has seen immense successes in syntactic analysis, in which the data-driven acquisition of information from annotated corpora (i.e., treebanks) has played a pivot role. The same development is currently gaining momentum in the area of predicate-argument structure, which models the relationship between *predicates* (e.g., verbs) and their semantic arguments or *semantic roles*. Often, the predicate is first assigned a sense or *semantic class*, which is followed by the assignment of roles appropriate for this sense. See Fig. 1 for a predicate-argument level analysis of a sentence: The predicate "pass" is used in its `Giving` sense with three arguments: an agent, realized as its (deep) subject, a theme, realized as object, and a manner, realized as adjunct. For the sentence "Fred passed the test quickly", a different sense of "pass" would be appropriate.

A number of projects have annotated large corpora with this kind of information, such as PropBank (Palmer et al., 2005) and FrameNet (Fillmore et al., 2003) for English, SALSA (Erk et al., 2003) for German, and the Prague Dependency Treebank (Hajičová, 1998) for Czech. The availability of data has kickstarted research on the use of predicate-argument structure, whose attractiveness lies in its intermediate position between syntax and "deep" semantics. It normalizes across more or less meaning-preserving syntactic transformations such as passivization (e.g. "The butter was passed quickly by Fred" would receive the same representation as in Fig. 1), and to some degree also across languages. Most paradigms provide role labels such as `Actor` or `Theme`, which characterize the relationship between predicate and argument as well as the relationship between arguments. This provides a handle on modeling inferences about role-fillers: for example, the `Theme` of a `Giving` event is the object that changes possessors. Conversely, predicate-argument structure ignores problems of deep semantic analysis such as modality, negation, or scope ambiguity.

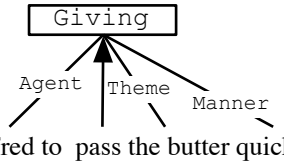

Figure 1: Role-semantic analysis of a short sentence.

These properties have generated interest in predicate-argument structure for content-related natural language processing tasks, such as Question Answering (Narayanan and Harabagiu, 2004) or Information Extraction (Moschitti et al., 2003). However, the serious use of predicate-argument information in NLP hinges on the ability to robustly and accurately label new, unrestricted text with semantic class and role information, a task also known as *shallow semantic parsing*. Even though there has been a lot of fundamental research on the task, starting from Gildea and Jurafsky (2002), up to the shared tasks at CoNLL (Carreras and Màrquez, 2004; 2005) and SENSEVAL 3 (Mihalcea and Edmonds, 2004), we are not aware of any robust shallow semantic parser which is freely available.

We address this problem by presenting SHALMANESER, a SHALlow seMANtic parSER which can be downloaded from our web site (see Sec. 4 for details). We conceptualize semantic analysis to be decomposable into a number of subproblems, which can be solved fairly independently; therefore, SHALMANESER is designed as a loosely coupled toolchain. Currently, the toolbox contains three modules: a preprocessor to parse plain-text input into the interchange format, a module for sense-disambiguation of predicates (FRED), and one for the assignment of semantic roles (ROSY). The modularity furthermore allows easy integration with other NLP tools. For example, the XML output of any module of SHALMANESER can be visualized directly with the SALTO tool (Burchardt et al., 2006), which is also available at our web site.

SHALMANESER is designed both as a platform for research in shallow semantic parsing, and as a "black box" to produce data with role-semantic annotation. In an "end user scenario", pre-trained classifiers for English and German are available for exploring the use of role-semantic information in different NLP settings. In a "research scenario", the modular architecture enables the integration of additional processing modules; furthermore, we have kept the processing components encapsulated to make them easily adaptable to new features, parsers, languages, or classification algorithms.

Frame: STATEMENT		
This frame contains verbs and nouns that communicate the act of a SPEAKER to address a MESSAGE to some ADDRESSEE using language. A number of the words can be used performatively, such as *declare* and *insist.*		
Frame Elements	SPEAKER	**Evelyn** said she wanted to leave.
	MESSAGE	Evelyn said **she wanted to leave**.
	ADDRESSEE	Evelyn told **me** about her past.
	TOPIC	Evelyn told me **about her past.**
	MEDIUM	Evelyn preached to me **over the phone**.
Predicates	acknowledge.v, acknowledgment.n, add.v, address.v, admission.n, admit.v, affirm.v, affirmation.n, allegation.n, allege.v, announce.v, announcement.n, assert.v, assertion.n, attest.v, aver.v, avow.v, avowal.n, caution.v, claim.n, claim.v, comment.n, comment.v, ...	

Table 1: Example frame from the FrameNet database

Plan of the paper. In Section 2, we give some background on Frame Semantics, the predicate-argument paradigm primarily used in SHALMANESER. Section 3 discusses our modeling of shallow semantic parsing as a set of loosely coupled problems. Section 4 gives a high-level overview of SHALMANESER's features, and Section 5 adds details about the individual components. Finally, Section 6 evaluates SHALMANESER quantitatively and qualitatively.

2. Frame Semantics

Frame Semantics (Fillmore, 1985) is a used-based theory of meaning and, more specifically, predicate-argument structure. In Frame Semantics the meaning of a predicate is described by reference to a *frame*, a conceptual structure describing a situation. Semantic roles, called *frame elements*, are local to frames and represent the agents and objects involved in that particular situation.

The Berkeley FrameNet project (Fillmore et al., 2003) is constructing a frame-semantic lexicon for English, which currently contains some 600 frames with 8,700 predicates, of which 5,700 are exemplified in annotated sentences from the British National Corpus. Table 1 shows the `Statement` frame as an example, which describes a communication situation. The frame definition contains a natural-language description, a list of frame elements, and the predicates which can introduce the frame.

3. Semantic analysis with a loosely coupled toolchain

Semantic parsing divides naturally into two subtasks, *class assignment* and *role assignment*. The task can in principle be modeled either as one integrated process, or as two separate modules, with class assignment preceding role assignment. Experiments by Gildea and Jurafsky (2002) with interleaved processing showed a small gain in accuracy at a huge processing cost; similar experiments by Erk (2005), who fed role assignment information back to class assignment, resulted in no improvement. So, given the large advantages of a modular architecture in general – where individual components can be exchanged, added, and removed without the need to change or even know the other components – we have modeled semantic parsing as a *loosely coupled chain of modules*. The components of SHALMANESER are connected only by a common interface format which enables annotation at different linguistic layers.

In addition to the advantages on the technical level, a modular architecture addresses the more fundamental question of a suitable framework for semantic processing in general. Semantic classes and roles are just one particular type among the many kinds of semantic information that are potentially helpful in NLP applications, such as lexical information (ontological status, lexical relations, polarity), structural information (scope, anaphoric links, modality, discourse structure), or proposition-level information (factivity). Currently, there is no comprehensive theoretical account of interaction between different kinds of information, even less a theory of processing.

However, the last years have seen impressive progress in the accurate computation of individual kinds of semantic information. This is why we believe that the best-suited architecture for semantic processing is a loosely coupled toolchain architecture with a flexible number of individual modules which work more or less independently to solve particular subproblems of the task. Which modules are necessary or helpful is very much a matter of the application. Crucial for the manageability of this kind of system is a well-defined interchange format which can represent the different kinds of information.

4. Features of SHALMANESER

Overview. SHALMANESER is a shallow semantic parsing tool. It assigns semantic classes (senses) to words, and it assigns semantic roles. Both sense and role assignment are modeled as supervised learning tasks. The system can be used with arbitrary sense inventories and semantic role paradigms, as long as training data is available.

As interchange format, we use SALSA/TIGER XML (Erk and Pado, 2004), an XML format designed for the representation of multi-level annotation. Extending TIGER XML (Mengel and Lezius, 2000), which conceptualizes syntax as a directed graph and is expressive enough to represent the output of many parsers, it allows annotation to refer to different layers of linguistic analysis by global node IDs. Input to the toolchain can be plain text, TIGER XML, SALSA/TIGER XML, or FrameNet XML.

A number of additional applications are either already SALSA/TIGER XML-aware, or will become so in the near future. Most importantly, the SALTO annotation tool (Burchardt et al., 2006) reads SALSA/TIGER XML and can therefore be used to inspect and manually modify the assigned frames and roles within a graphical interface. Figure 2 shows an example of SHALMANESER output as visualized by the SALTO tool.

For researchers primarily interested in a robust system for shallow semantic analysis, SHALMANESER comes with pre-trained classifiers for English and German. A single command starts the complete analysis of plain text input, encompassing syntactic analysis, frame assignment and

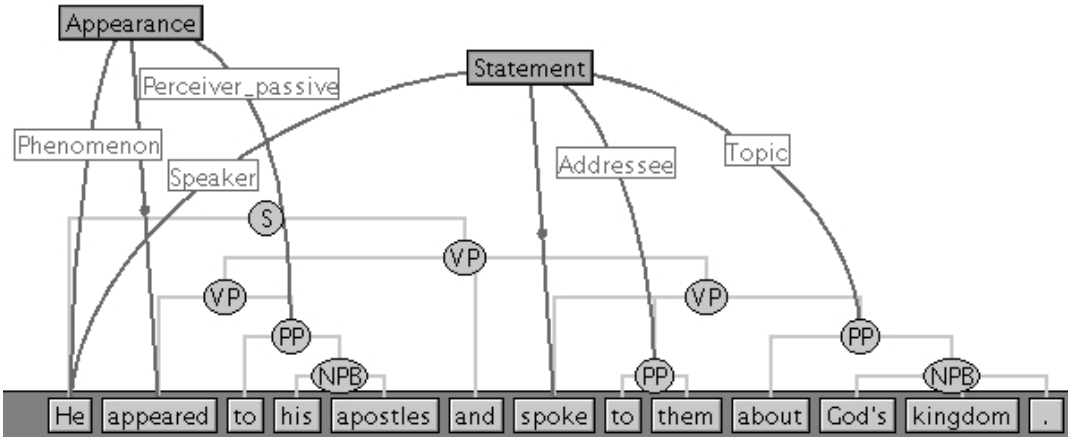

Figure 2: Example output of SHALMANESER: Acts 1:3b

role assignment. More specifically, the training data for English is the FrameNet release 1.2 dataset, consisting of 133,846 annotated BNC examples for 5,706 predicates. For German, the training data is the currently annotated portion of the SALSA/TIGER corpus (Erk et al., 2003), 17,743 annotated instances covering 485 predicates.

Flexibility. One aim of SHALMANESER is to allow research in semantic role assignment on a high level of abstraction and control. Studies in this area typically involve a comparative evaluation of different experimental conditions, e.g. the activation and deactivation of model features. These and other conditions are be specified declaratively in *experiment files*. SHALMANESER is designed to enable easy adaptability to new languages, integration of additional syntactic parsers and machine learning systems, and addition of new features.

Architecture. SHALMANESER is realized as a loosely coupled toolchain, as described in Sec. 3. The architecture is shown is Figure 3. The components are: (a), a *preprocessing* module, to analyze plain text input (lemmatization, part of speech tagging, and parsing); (b) FRED, a FRame Disambiguator (detection and sense-disambiguation of known predicates); and (c), ROSY, a ROle assignment SYstem (identification and labeling of semantic roles in the linguistic context of each predicate introducing a frame).

Obtaining and using SHALMANESER. SHALMANESER is written in Ruby, an object-oriented scripting language. The additional requirements are as follows: For preprocessing, external NLP tools for linguistic analysis (see Section 5.1). FRED is self-contained. For ROSY, an installed MySQL database server for data storage, and one of the supported classification toolkits (see Section 5.3). The complete system can be downloaded from `http://www.coli.uni-saarland.de/projects/salsa/page.php?id=software`.

5. Component details

5.1. Preprocessing

SHALMANESER includes a preprocessing component, called FRPREP, which combines external parsers, lemmatizers and part-of-speech taggers to produce

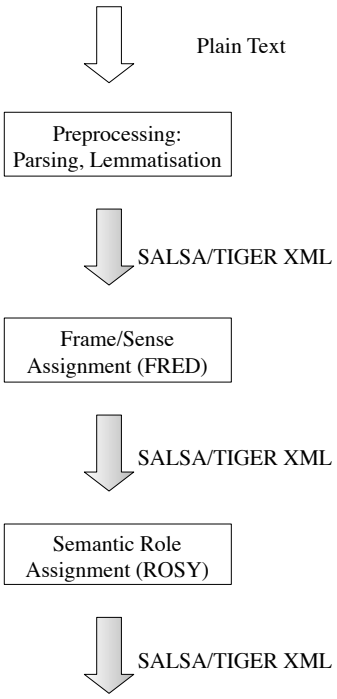

Figure 3: SHALMANESER: A loosely coupled toolchain

SALSA/TIGER XML. To keep the system flexible and extensible, FRPREP interacts with all syntactic analysis components through a common abstract interface instantiated for individual systems. Currently, we support the Collins (Collins, 1997) and Minipar (Lin, 1993) parsers for English and the Sleepy parser for German (Dubey, 2005); furthermore we support Treetagger (Schmid, 1994) for both English and German lemmatization and the TNT part-of-speech tagger (Brants, 2000).

5.2. Fred

FRED is a system for supervised Word Sense Disambiguation. It uses a rich set of features consisting of

- a bag-of-words context, with a window size of one or more sentences;
- bigrams and trigrams centered on the target word;

- grammatical functions of the target word: function labels, head words, and a combination of both. In addition, the concatenation of all function labels is used as a feature. For PPs, function labels are extended by the preposition.
- for verb targets, the target voice.

The feature set is based on Florian et al. (2002), but extends the list of syntax-related features. All word-related features exist in three variations, for the word itself, its lemma, and its part of speech.
Currently, FRED uses a Naive Bayes classifier, estimating feature probabilities using (smoothed) maximum likelihood estimation on weighted features. However, the machine learning component is completely encapsulated, which makes it easily replaceable by other, external machine learning toolkits.

5.3. Rosy

ROSY assigns semantic roles to the linguistic context of a predicate, based on the semantic class assigned to the predicate. ROSY offers a high degree of flexibility in modeling the task: The task can be performed in one step, or it can be split into *argument recognition* (*argrec*) and *argument labeling* (*arglab*). The first step, *argrec*, distinguishes only between roles and non-roles; *arglab* performs a more detailed classification on the instances recognized as roles in the first step. Furthermore, classifiers can be trained on different *data groups*, e.g. frame-wise or by target part-of-speech. This flexibility is accomplished by using a database as back end to store the features, which allows different "views" on the data.
The current implementation of ROSY includes some 30 features, mostly motivated by current state-of-the-art systems (see e.g. (Carreras and Màrquez, 2005)).
ROSY assigns semantic roles to constituents of the syntactic structure. Only a small fraction of all constituents can potentially bear a role, and still fewer are actual role-bearers. To make the classification task easier, ROSY can apply the pruning scheme proposed by Xue and Palmer (2003), which uses parse tree structure to decide if constituents can possibly bear a role.
To keep the system independent from particular machine learning paradigms, we have implemented an interface to external machine learning toolkits. We currently support Mallet (McCallum, 2002), TiMBL (Daelemans et al., 2003), and Malouf's `estimate` Maximum Entropy learner (Malouf, 2002).

Clearing up syntax-semantics mismatches. During preprocessing, the span of semantic roles in the training corpora is projected onto the output of the syntactic parser by assigning each role to the set of maximal constituents covering its word span. If the word span of a role does not coincide with parse tree constituents, e.g. due to misparses, the role is "spread out" across several constituents. This leads to idiosyncratic paths between predicate and semantic role in the parse tree. Figure 4 shows an example where a parser error has included the *will* into the NP *This delightful, animated musical*, which has led to the STIMULUS role being assigned to multiple constituents. We would rather not have the classifier learn to copy this particular parser mistake. ROSY offers an option to make role spans *in the training data* simpler and more uniform, in our example replacing the multiple STIMULUS labels by the single circled one, using the following **span standardization algorithm**:

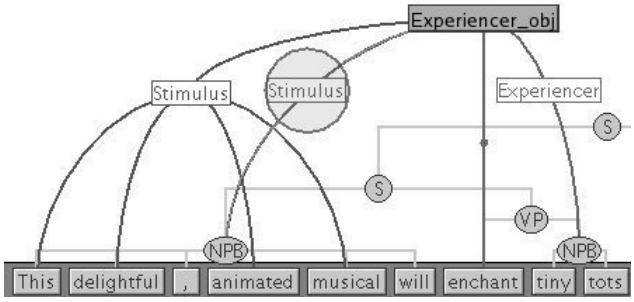

Figure 4: Clearing up a syntax/semantics mismatch

> Given a role r that has been assigned, let N be the set of *terminal* nodes of the syntactic structure that are covered by r.
>
> Iteratively compute the maximal projection of N in the syntactic structure:
>
> 1. If n is a node such that all of n's children are in N, then remove n's children from N and add n instead.
> 2. If n is a node with 3 or more children, and all of n's children except one are in N, then remove n's children from N and add n instead.
> 3. If n is an NP with 2 children, and one of them, another NP, is in N, and the other, a relative clause, is not, then remove n's children from N and add n instead.
>
> If none of the rules is applicable to N anymore, assign r to the nodes in N.

Rule 1 implements normal maximal projection. Rule 2 "repairs" parser errors where all children of a node but one have been assigned the same role. Rule 3 addresses a problem of the FrameNet data, where relative clauses have been omitted from roles assigned to NPs.

6. Evaluation

Our main purpose in building SHALMANESER was not to surpass the accuracy of the systems reported in the literature, most of which were optimized on one particular dataset. Instead, we aim at providing a robust system which can be adapted to different users' needs. Still, this section provides an evaluation for SHALMANESER's modules on English and German data for a realistic assessment of the performance level to be expected from the system. We first give a quantitative evaluation against gold-standard annotation, then we discuss impressions from an experiment in which we applied SHALMANESER to free text from a different genre.

6.1. Quantitative Evaluation

We evaluate FRED and ROSY against manually annotated data. The English evaluation is based on FrameNet release 1.2 (preprocessing with Treetagger, TNT, and the Collins parser), the German evaluation on the SALSA corpus (preprocessing with Treetagger and the Sleepy parser). For each language, we split the data sets randomly into a training (90%) and test (10%) portion.

FRED. Table 2 shows the overall system accuracy, compared to a "most frequent sense" baseline. The high baseline for English is due to the fact that FrameNet, which progresses one frame at a time, provides an incomplete sense inventory for many words. The German data, constructed in a different fashion, has on average about twice as many senses per lemma as the English data (Erk, 2005).

ROSY. The classification task was split into *argrec* and *arglab*. *argrec* used only syntactic features, while *arglab* used syntactic as well as lexical features. For a realistic setting, argument labeling was evaluated on the result of the *argrec* step rather than perfect argument boundaries. The Mallet maximum entropy package was used as a classifier.

The results can be seen in Table 3. For *argrec*, which is a binary decision between *role* and *no-role*, the table gives precision, recall and F-score for the class *role*. For the n-way decision task of *arglab*, the table gives the overall accuracy. The lower overall results for the German data reflect the smaller size of the training set.

We tested Xue and Palmer's (2003) pruning scheme on both sets prior to processing. The percentage of roles retained after pruning is comparable for English and German (86.2% vs. 81.9%), but while classifier results for English profit from pruning by 3 points in F-score, the classifiers for German suffer a drop of 3.2 points in F-score through pruning. We surmise that this is due to the overall sparseness of the data for German. We therefore retained pruning for English, but skipped it for German.

The optimal evaluation of the *span standardization algorithm* described above would be against a gold standard corpus annotated with perfect syntax/semantics correspondence. In the absence of such a corpus, we can evaluate the algorithm only against the unchanged test data with all its syntax/semantics mismatches. Although this will probably result in a systematic underestimation of the algorithm's contribution, we have repeated the *argrec* task on span-standardized training English data and evaluated it against the same test data as above. The result achieves a precision of 0.868 (as opposed to 0.855 without standardization), i.e. the roles assigned by the classifier tend to be more reliable. Recall drops from 0.751 to 0.641, as was to be expected, since the syntax/semantics mismatches in the gold annotation of the test data are duplicated by the classifier.

In fact, the classifications produced the standardized and the non-standardized classifier differed in a significant number of cases, namely for 2,887 out of 13,396 test sentences (21.6%). A manual inspection of individual differences showed that the span-standardized classifier assigns less roles in total, but those which are assigned are generally more appropriate.

Data	Acc.	Baseline
English (FrameNet data)	0.932	0.888
German (Salsa data)	0.790	0.751

Table 2: FRED evaluation results: overall accuracy and baseline

	argrec			*arglab*
Data	Prec.	Rec.	F	Acc.
English	0.855	0.669	0.751	0.784
German	0.761	0.496	0.600	0.673

Table 3: ROSY evaluation results

6.2. Qualitative Evaluation

To obtain an impression of the current state of shallow semantic parsing models when applied to free text, we consider a text from a rather different domain, the Bible. More specifically, we have run SHALMANESER on the Acts of the Apostles in the Contemporary English Version (Luke, 1995), applying the same preprocessing as before (TreeTagger, TNT, Collins Parser), and using the FrameNet classifiers. Verse 1:3b, shown in Fig. 2, illustrates both successes and problems.

With respect to accuracy, we find a large number of correct assignments. In our example, the frame STATEMENT has been assigned correctly, and all roles have been recognized correctly, even the nonlocal subject. This mirrors the high precision we find in the quantitative evaluation.

Coverage, however, is limited: we obtain on average only 3 frames per sentence, at an average sentence length of 19 words. Since many predicates are not listed in FrameNet, they can receive neither semantic class nor roles (the *predicate coverage problem*). In our example, at least "kingdom" could reasonably receive a frame, and a role to model its relation to its argument "God".

However, even if a predicate is covered by FrameNet, there is no guarantee that all senses are listed (the *sense coverage problem*). Consider the predicate "appeared" in our example. The sense assigned by FRED is actually not appropriate: the FrameNet-frame APPEARANCE refers to situations where some PHENOMENON exhibits properties described through a CHARACTERIZATION, such as:

$$[_{Phen.}\ \text{The violins}]\ \textbf{sounded}\ [_{Char.}\ \text{horrible}].$$

where the "appeared" in our example has the sense "to become apparent", for which no frame currently exists in FrameNet. Recent advances in recognizing missing senses by outlier detection (Erk, 2006) provide a way of identifying such cases automatically. Also, on the positive side, even the assigned frame is very similar to the ideal frame as we envisage it – both are perception-type frames – so that even the misclassification of the predicate has led to reasonable role assignment, which can form the basis for inferences about the participants.

Another serious issue that becomes apparent in qualitative evaluation is the treatment of multiword expressions, which are frequent in this text, and even more frequent in newspaper text. SHALMANESER does not support multiword predicates at the moment, and while FrameNet lists some

multiword expressions, it does not offer a full account of multiword predicates in all syntactic and lexical variations.

7. Conclusion

We have presented SHALMANESER, a flexible toolchain for the automatic assignment of senses and semantic roles to text. SHALMANESER was designed with two main usage scenarios in mind: on the one hand, research on the shallow semantic parsing task itself can use the software as a platform that offers flexibility with respect to parsers, languages, classification paradigms, and conceptualizations of the task. On the other hand, end users can use it "out of the box" as a frame-semantic parser for English and German. We hope that SHALMANESER can facilitate the further study of practical uses of predicate-argument structure analyses, and that it can support further theoretical investigations into semantic phenomena and how they can or should be treated in shallow semantic representations.

Acknowledgments. Work in the project Salsa-II has been funded by DFG (Grant PI 154/9-2).

8. References

T. Brants. 2000. TnT – a statistical part-of-speech tagger. In *Proceedings of ANLP 2000*, Seattle, WA.

A. Burchardt, K. Erk, A. Frank, A. Kowalski, S. Pado, and M. Pinkal. 2006. SALTO – a versatile multi-level annotation tool. In *Proceedings of LREC 2006*, Genoa, Italy.

X. Carreras and L. Màrquez, editors. 2004. *Proceedings of the CoNLL shared task: Semantic role labelling*.

X. Carreras and L. Màrquez, editors. 2005. *Proceedings of the CoNLL shared task: Semantic role labelling*.

M. Collins. 1997. Three generative, lexicalised models for statistical parsing. In *Proceedings of ACL/EACL 1997*, pages 16–23, Madrid, Spain.

W. Daelemans, J. Zavrel, K. van der Sloot, and A. van den Bosch. 2003. Timbl: Tilburg memory based learner, version 5.0, reference guide. Technical Report ILK 03-10, Tilburg University. Available from `http://ilk.uvt.nl/downloads/pub/papers/ilk0310.ps.gz`.

A. Dubey. 2005. What to do when lexicalization fails: parsing German with suffix analysis and smoothing. In *Proceedings of ACL 2005*, Ann Arbor, Michigan.

K. Erk and S. Pado. 2004. A powerful and versatile XML format for representing role-semantic annotation. In *Proceedings of LREC 2004*, Lisbon, Portugal.

K. Erk, A. Kowalski, S. Pado, and M. Pinkal. 2003. Towards a resource for lexical semantics: A large German corpus with extensive semantic annotation. In *Proceedings of ACL 2003*, pages 537–544, Sapporo, Japan.

K. Erk. 2005. Frame assignment as word sense disambiguation. In *Proceedings of IWCS 2005*, Tilburg, The Netherlands.

K. Erk. 2006. Unknown word sense detection as outlier detection. In *Proceedings of NAACL 2006*, New York, NY.

C. Fillmore, C. Johnson, and M. Petruck. 2003. Background to FrameNet. *International Journal of Lexicography*, 16:235–250.

C. Fillmore. 1985. Frames and the semantics of understanding. *Quaderni di Semantica*, IV(2).

R. Florian, S. Cucerzan, C. Schafer, and D. Yarowsky. 2002. Combining classifiers for word sense disambiguation. *Journal of Natural Language Engineering*, 8(4):327–341.

D. Gildea and D. Jurafsky. 2002. Automatic labeling of semantic roles. *Computational Linguistics*, 28(3):245–288.

E. Hajičová. 1998. Prague Dependency Treebank: From Analytic to Tectogrammatical Annotation. In *Proceedings of TSD'98*, pages 45–50, Brno, Czech Republic.

D. Lin. 1993. Principle-based parsing without overgeneration. In *Proceedings of ACL-93*, Columbus, OH, USA.

Luke. 1995. The Acts of the Apostles. In *The Bible (Contemporary English Version)*. American Bible Society.

R. Malouf. 2002. A comparison of algorithms for maximum entropy parameter estimation. In *Proceedings of CoNLL 2002*, Taipei, Taiwan.

A. McCallum. 2002. Mallet: A machine learning for language toolkit. Available from `http://mallet.cs.umass.edu`.

A. Mengel and W. Lezius. 2000. An XML-based encoding format for syntactically annotated corpora. In *Proceedings of LREC 2000*, Athens, Greece.

R. Mihalcea and P. Edmonds, editors. 2004. *Proceedings of Senseval-3: The Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text*, Barcelona, Spain.

A. Moschitti, P. Morarescu, and S. Harabagiu. 2003. Open-domain information extraction via automatic semantic labeling. In *Proceedings of the 14th Florida AI Conference*, pages 397–401, St. Augustine, FL.

S. Narayanan and S. Harabagiu. 2004. Question answering based on semantic structures. In *Proceedings of COLING 2004*, pages 693–701, Geneva, Switzerland.

M. Palmer, D. Gildea, and P. Kingsbury. 2005. The proposition bank: An annotated corpus of semantic roles. *Computational Linguistics*, 31(1).

H. Schmid. 1994. Probabilistic part-of-speech tagging using decision trees. In *Proceedings of NeMLaP 1994*.

N. Xue and M. Palmer. 2003. Calibrating features for semantic role labeling. In *Proceedings of EMNLP 2004*, Barcelona, Spain.