Core Natural Language Processing Technology Applicable to Multiple Languages

Workshop '98

Center for Language and Speech Processing
Johns Hopkins University
Baltimore, MD, USA

Final Report

Jan Hajic, Eric Brill, Michael Collins, Barbora Hladká, Douglas Jones, Cynthia Kuo, Lance Ramshaw, Oren Schwartz, Christoph Tillmann, Daniel Zeman

Introduction

Parsing natural languages is a task approached by many people in the past and present and undoubtedly also in the future. It is widely believed that if one can obtain automatically a good parse for a given input sentence (i.e. functions of the words and relations between them), then all the "language understanding" tasks can profit from it.

English has been studied most for this purpose and many good results have been obtained, especially using statistically-based techniques coupled with automatic learning from large annotated corpora. However, these techniques have not been applied to other languages than English very much, especially to languages which display substantially different behavior (both morphologically and syntactically) than English (such as Slavic languages, spoken mostly in Europe by approx. 350 million people).

The Task

Czech has been selected as the "example" language for the Workshop'98 parsing task mainly for two reasons: (1) it is substantially different from English (highly inflectional morphology, free word order), and (2) at least some resources (data, mainly the so-called Prague Dependency Treebank (PDT), and tools, such as morpholgical dictionary and taggers) exist for automatic learning techniques to work reasonably.

We have worked on the following parsers before and during the workshop: Michael Collins' state-of-the-art parser for English (which was being adapted for Czech), Dan Zeman's "direct" dependency parser for Czech, and Ciprian Chelba's and Fred Jelinek's Structured Language Model (as a parser). Also, we were exploring the possibility (and discovering the difficulties) of parsing an "unknown" language using the "Parsing by translation" approach.

On top of those parsers, we were working on the "Classifier Combination" method for combining the results of several parsers to obtain still better results. Although we could not test this method on really different parsers, interesting results have been obtained using various variants of a single parser.

Evaluation Criteria

In order to compare the results among different parsers, we have set up an evaluation scheme which would be applicable to all parsers. As the test data consist of dependency trees, it was natural to choose the following evaluation metric:

An error occurs if and only if a dependency link goes to a wrong governing node.

Every sentence contains one artificial node (tree root), thus the number of nodes representing "real" nodes equals the number of dependency links (counting also links leading to the artificial root node). Parsing accuracy can thus be defined very simply:

Accuracy = number of correct dependency links / number of words in a sentence

provided that each node has been supplied by one and only one upgoing link by the parser being evaluated. Precision and recall can be easily defined, should some parser(s) provide partial parses or multiple parses.

Special evaluation tools have been developed, and all parser "developers" have been instructed to provide the output of their respective parsers in the test data format for automatic evaluation.

There were two test sets of data. One of them (the development test set) has been freely available, and people could in fact use it in any way they wanted. The final evaluation test set has been kept separate for final and completely unbiased evaluation.
Both sets consisted of over 3500 sentences, whereas the training set contained over 19 thousand sentences. The original partitioning of the Prague Dependency Treebank into training and the two test sets is still kept in the freely available version of the PDT (ufal.ms.mff.cuni.cz), so that results obtained later and/or by other people are directly comparable.

Data Preparation

The Prague Dependency Treebank (see also ufal.ms.mff.cuni.cz -- then go to Projects, and to PDT) has been prepared for the Workshop with full annotations on level 1 (morphology) and level 2 (dependency syntax). Although the PDT is still being developed (final version shoud be available before the end of 1999), we were able to obtain almost half a million of words annotated on these two levels at the start of the Workshop. Substantial effort has been spent before the workshop and even in the first three weeks of the workshop to check the PDT and merge the annotation levels 1 and 2 into a single data resource, which would otherwise be necessary only sometimes during 1999 when all the manual annotation work is finished.

The sizes of the data:

	Sentences	Words
Training	19126	327597
Development test	3697	63718
Evaluation test	3787	65390

Resources and Tools

The following resources were available for the workshop (part of them has been developed in the past without direct relation to the workshop, but some, such as the evaluation tool and the tree viewer have been developed by the team members berfore and during the workshop):

- the Prague Dependency Treebank (now dubbed "PDT version 0.5", almost 450,000 words, or about 26,000 sentences), with each word annotated with its original textual form, lemma, morphosyntactic tag, analytical (syntactic) function and its governing node. The original position of the word in the input sentence was also preserved.
- Bilingual English-Czech data (2 mil. words) from the Czech edition of Reader's Digest, sentence-aligned, for the "parsing by translation" experiments
- Czech morphological dictionary and morphological analyzer, which for (almost) each word from a Czech text produces a list of possible grammatical meanings (on the morphological level) and all possible lemmas. The coverage of the dictionary is about 98% of a newspaper text. The tagset is very detailed (as the inflectional nature of Czech dictates) and currently contains 3128 tags.
- Czech tagger (so called "exponential" tagger), which can disambiguate the tags as provided by the morphlogical dictionary. The error rate of the tagger is at about 92% (flat, using all morphological categories), but for example for the POS alone it is 99%.
- Evaluation tool for comparison of two files, counting differences in the dependent - governor relation and computing the accuracy, precision and recall at exit.
- Java-based dependency tree viewer for simultaneous viewing of several dependency trees. This tools is used for "manual" evaluation of parsing results. It can also, e.g., highlight the differences between two parses.

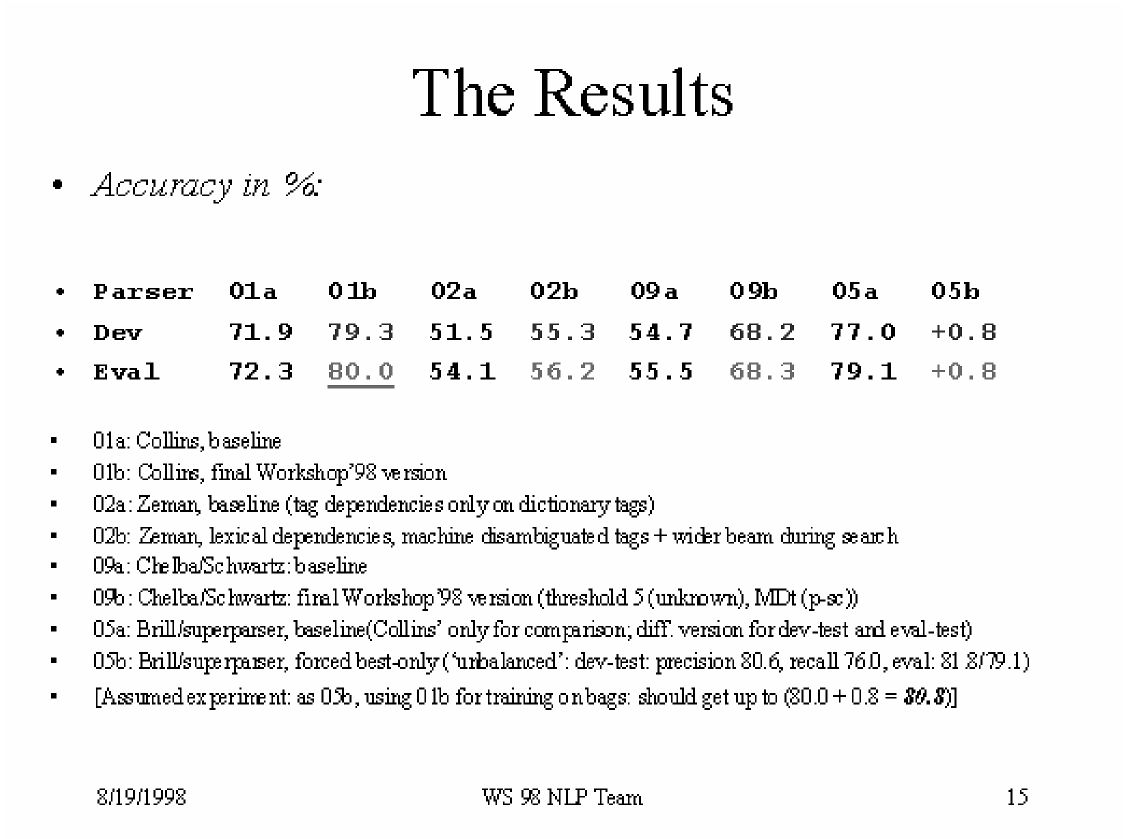
Results and Achievements (an overview)

We believe that the main achievements of the summer Workshop'98 in the area of parsing are:

- 3 parsers working with a language different from English, all of which have improved upon a baseline set before the workshop

- confirmation of the hypothesis that the lexicalized CF parsing can be adapted for handling other languages, albeit with somewhat lower accuracy
- a set of observations regarding the "Parsing-by-translation" approach
- implemented the "Superparser" method for combining the ouput of several parsers, working at the moment by "bagging"
- data availability for further experiments by the team or by anybody else (the Treebank is now freely available)
- improved Czech tagger
- identified open problems regarding parsing Czech as a representative of highly inflective and free-word order languages.

The most important quantitative results obtained at the end of the workshop are summarized here:



The Results

- *Accuracy in %:*

Parser	01a	01b	02a	02b	09a	09b	05a	05b
Dev	71.9	79.3	51.5	55.3	54.7	68.2	77.0	+0.8
Eval	72.3	80.0	54.1	56.2	55.5	68.3	79.1	+0.8

- 01a: Collins, baseline
- 01b: Collins, final Workshop'98 version
- 02a: Zeman, baseline (tag dependencies only on dictionary tags)
- 02b: Zeman, lexical dependencies, machine disambiguated tags + wider beam during search
- 09a: Chelba/Schwartz: baseline
- 09b: Chelba/Schwartz: final Workshop'98 version (threshold 5 (unknown), MDt (p-sc))
- 05a: Brill/superparser, baseline(Collins' only for comparison; diff. version for dev-test and eval-test)
- 05b: Brill/superparser, forced best-only ('unbalanced': dev-test: precision 80.6, recall 76.0, eval: 81.8/79.1)
- [Assumed experiment: as 05b, using 01b for training on bags: should get up to (80.0 + 0.8 = ***80.8***)]

8/19/1998 WS 98 NLP Team 15

The Parsers

In this section we describe briefly each of the five parsing efforts. More detailed accounts can be found in the respective chapters of this report.

Lexicalized Context-Free Parser

(Chapter 1)

Recent work in statistical parsing of English has used lexicalized trees as a representation, and has exploited parameterizations that lead to probabilities which are directly associated with dependencies between pairs of words in the tree structure. Typically, a corpus such as the Penn treebank is used as training and test data: hand-coded rules are used to assign head-words to each constituent in the tree, and the dependency structures are then implicit in the tree.

In the Czech PDT corpus we have dependency annotations, but no tree structures. For parsing Czech we considered a strategy of converting dependency structures in training data to lexicalized trees, then running the parsing algorithms originally developed for English. Crucially, the mapping from dependencies to trees is one-to-many. The choice of tree structure is important in that it determines the parameterization of the model: that is, the independence assumptions that the parsing model makes. There are at least 4 degrees of freedom when deciding on the tree structures:

1) How ‘‘flat’’ should the trees be? The trees could be maximally flat (one phrasal level per head-word), binary branching, or anything between these two extremes.

2) What set of part of speech (POS) tags should be used?

3) What non-terminal labels should the internal nodes have?

4) How to deal with crossing dependencies?

As a baseline system we: 1) made the trees as flat as possible; 2) chose just the major categories (noun, verb etc.) as parts of speech; 3) derived non-terminal labels from the POS tag of the headword (for example, a phrase headed by a noun would be labeled NP); 4) effectively ignored the crossing dependencies problem.

During the workshop we refined this conversion process in several ways: modifying the tree structures for linguistically special cases such as coordination and relative clauses; experimenting with different POS tag-sets; and modifying the parsing model (for example, extending it to handle bigram dependencies at the phrasal level).

The baseline results on the final test set were 72.4% accuracy. The final system, with all refinements, recovered dependencies with 80.0% accuracy. Results on the development set showed that newspaper-style text (75% of the sentences in test data) were parsed at around 2% greater accuracy than this averaged, 80% result. These results therefore compare favorably to those on English Wall Street Journal text: that is, around 90% accuracy but with considerably more training data (890,000 words vs. 347,000 for Czech).

A "Direct" Dependency Parser

(Chapter 2)

This Workshop "sub"-project involved a parser for Czech based on direct statistical modeling of tag / word dependencies in Czech. In comparison to the Lexicalized CF parser, this does not use any grammar, and works directly with dependency trees instead of parse trees.

Several techniques were developed in preparation for this workshop, that had not been published before and that help the tag-based part of the parser. Although they were prepared before the workshop, the workshop brought a great deal to their implementing for the Prague Dependency Treebank, and to testing them thoroughly.

Then the second part of the workshop was devoted to lexicalizing the parser, i.e. to developing a new statistical model that deals with dependencies between words rather than between the morphological tags. It is true that this part did not help the parser as much as expected (and as reported for other parsers for English) but the outcomes are still challenging and the model developed here enables to continue with the research in future.

Let us very briefly look at the structure of the parser. Its main task can be characterized by the following expression:

$$\arg\max_{T} p(T|S) = \arg\max_{T} \left(p(S|T) \cdot p(T)\right)$$

It means that the parser wants to find the dependency tree T that maximizes p(T|S) where S is the sentence being parsed. In other words, we want to construct the tree that most likely is a dependency structure of the sentence S. Because in no way we are able to decide among all possible trees in the universe, we have to decompose the tree probability into edge probabilities. These can be estimated from the relative frequencies of dependencies in training data. Then, using the Viterbi search algorithm, we try to find the tree with the highest possible product of probabilities of its edges. Here we take a significant simplification that the dependency probabilities are statistically independent, i.e.

$$p(T) = \prod_{i=1}^{n} p(d_i)$$

This obviously is not true and weakens the parser so that we had to introduce various constraints, additional models and techniques that help us a little to work around this weakness. A list of them follows; a more detailed description will be given later in this report.

- Crossing dependencies (so-called non-projective constructions) are not allowed.
- A supervised reduction of morphological tag set was done. The number of different tags occurring in corpus decreased from about 1000 to about 400.
- A new model for valency was added. It says the parser how likely a node with a given tag has a particular number of child nodes.
- We take into account whether the words forming a dependency are adjacent in the sentence or not.
- We take into account whether the dependency goes to the right (the governing node precedes the dependent one in the sentence) or to the left.

The following table gives a brief summary of the results in terms of parsing accuracy. That is, each number is the percentage of dependencies generated by the parser that were correct. Unless stated otherwise, all the numbers characterize parsers trained on over 19000 sentences (approx. 300000 words) and tested on one of the two test data sets, the development test data, and the evaluation test data. The e-test data was used only at the very end of the workshop to cross-validate the results, so most stages have been tested with the d-test only. The training set and both the test sets contained texts from three different sources, from a daily newspaper, from a business weekly, and from a scientific magazine. It turned out to be much more difficult to parse the last one (mainly because of the sentence length) so it seems reasonable to give separate results for the scientific magazine (labeled "sci") and for the rest (labeled "norm").

The baseline parser includes all the techniques whose development started before the workshop so "baseline" may be read as "before lexicalization". A deeper description of the techniques and a more diversified summary of their contribution to the parsing accuracy will be given later in this report. The final results include the lexical part of the parser as well as some minor improvements that will be described later, too.

	D-test			E-test		
	norm	sci	all	norm	sci	all
Baseline	54	48	51	57	51	54
Final	57	52	55	58	53	56

The "Parsing by Translation" Approach

(Chapter 3)

We conducted a preliminary survey of the prospects of using bilingual corpus to generate a grammar for monolingual parsing. The purpose of the survey was primarily educational. We wanted to learn about the current state of the art in parsing as applied to large-scale corpora. We examined the Czech Readers Digest corpus, which consists of around 2 million words of aligned Czech and English sentences and has been preprocessed in Prague. Since we had parsers available for both languages, we parsed both sides of the corpus and surveyed the parse structures. Although we did not have time over the summer for a large-scale project to automatically generate one grammar from the other, we felt that there were enough systematic structural correspondences to warrant further work in this direction. The purpose of this report is to describe some details about this very valuable bilingual corpus and to comment on the tools we used for our survey.

We conducted a preliminary survey of how we might use a bilingual corpus to generate a grammar for monolingual parsing. In particular, we examined the Czech *Readers Digest* corpus, which consists of around 2 million words of aligned Czech and English sentences. These sentences were taken from the Czech *Readers Digest* articles paired with the original English *Readers Digest* articles from which they were translated. The essential idea is that correspondences in the bitext imply structure for the two monolingual halves of the text. In the ideal case, we would like to infer the structure itself, based on the knowledge that the aligned sentences are translations of each other, or perhaps to parse the English side and use those structures to infer the Czech parses. In these cases, we would have a way to build up a new treebank, based in part on an analysis of the English side of the bilingual corpus. Furthermore, since we do have the Czech treebank, we could compare any new results with it as a reference point. Our somewhat more modest goal was to see to what extent the English parses and Czech parses correspond, given the corpus, parsers, and other resources that are available at the workshop. Since very large Czech and English treebanks are now available, we are able to train parsers to parse both sides of the corpus.

Our original motivation for trying this experiment at the workshop is to provide a very different source of grammatical information for Eric’s "Super Parser". Our expectation is that the Super Parser will make the best improvement over the individual parsers when the individual parsers behave very differently. Being able to infer the Czech parses from the English side of course would have met the criterion of being a very different source of grammatical information. Moreover, to the extent that such an exercise is possible, we would have a means of building trainable parsers for languages for which treebanks are not available, but for which bilingual texts are available.

Adaptation of the "Structured Language Model" for Parsing Czech

(Chapter 4)

The Structured Language Model (SLM) was introduced by Ciprian Chelba and Frederick Jelinek as a language model that uses a statistical parser in a left to right manner in order to exploit syntactic information for use in a language model as part of a speech recognition system. In traversing a sentence, the Structured Language Model produces a series of partial lexical parses whose exposed headwords are used instead of the previous two words in a trigram-like prediction process. It is conjectured that this language model could be easily modified to produce good complete parses for Czech. One of the attractive features of the parsing method used in this model is that it can be easily modified to handle "crossing" or non-projective dependencies, a feature of the Prage Dependency Treebank that our other parsers currently ignore. Chelba, who implementated his Structured Language Model in C++, was gracious enough to allow us access to his code in order to modify it for our purposes. During the workshop, with time constraints and the usual range of difficulties encountered, we had time only to modify this Structured Language Model to work properly with the Czech data, to experiment with unknown word statistics, and to use part of speech tags generated by a version of Jan Hajic’s exponential model statistical tagger. The results obtained during the workshop are encouraging as they were obtained with a version of the SLM which, though modified, is still not optimized for parsing.

The Structured Language Model consists of two main parts: a parser and a predictor. The model proceeds along a sentence in a left to right manner, constructing partial parses for the sentence prefix available at each word. These partial parses consist of binary branching lexical parse trees where each node is associated with a lexical item and a nonterminal or part of speech (POS) label. Each nonterminal label and lexical item pair that covers a partial parse is referred to as a headword. The predictor uses the last two exposed headwords over a sentence prefix to predict the next word in the sentence. With parameter reestimation Chelba and Jelinek report results that this technique does achieve significantly lower perplexities for the UPenn Treebank of Wall Street Journal text.

Partial parses are constructed in a binary manner by considering the last two headwords and their associated tag information. The parser can choose from three moves at any given point: ADJOIN-RIGHT, ADJOIN-LEFT, or NULL. An ADJOIN-RIGHT move creates a new nonterminal that covers the last two words, percolating the right word up the parse as the headword of the new phrase. ADJOIN-LEFT acts similarly, except that the left word is percolated up. After each adjoin operation, headwords are renumbered. The parser continues to build syntactic structure over a sentence prefix in this manner until the most probable move is the NULL move, which does not change the parse structure, but passes control to the predictor, which then predicts the next word and its POS tag from the previous two headwords. The model proceeds down each sentence in this manner until the end of sentence marker is reached. Because of the large (exponential) number of possible parse trees associated with any given sentence, this model uses a multiple stack search with pruning through the space of sentence parse tree hypotheses (see Chelba, Jelinek 1998). In this manner, hypotheses with equal numbers of parser operations and predictions are compared against each other.

Baseline results:

No unknown word statistics. No use of external (MDt = Machine Disambiguated tags) POS tags.

devel	54.7%
eval	55.5%

Use MDt tags for:

unknown word threshold:3	none	unk	all
devel	57.91%	67.42%	67.54%
eval	57.70%	67.16%	67.45%

unknown word threshlod:5	none	unk	all
devel	--	68.04%	68.18%
eval	57.29%	68.12%	68.32%

Various directions of future research are also discussed, such as handling crossing dependencies, parse closing strategies, optimization of the objective function, predictior probability decomposition, changes in data the annotation scheme, and optimization of the POS tagset.

The Superparser

(Chapter 5)

The goal of the superparser group was to explore the efficacy of combining the output of multiple parsers in hopes of generating a dependency structure of higher quality than that achieved by any single parser. Classifier combination has proven to be a powerful tool for improving the performance of machine learning algorithms and has recently been applied with success in natural language processing to a number of tasks, including part of speech tagging, word sense disambiguation and named entity recognition.

Our ultimate goal is to combine a diverse array of mature parsers. While we are currently developing a number of parsers for Czech (see elsewhere this report), at the moment one parser performs significantly better than the others, and appears to outperform the other parsers across all linguistic environments (reference table in superparsing section). Therefore, we will have to wait until the other parsers improve before we can expect gains through combination.

We instead focused primarily on diversification and combination of a single parsing algorithm, namely the Collins Parser (Chapter 1 of this report) ported to Czech. To generate a set of variants of the parser, we employed a technique known as bagging. Given a training set with N instances (sentences), we generate a single bag by randomly drawing instances from the training set N times with replacement. Once we have a training bag, we can then train the parser on that bag. We can generate as many bags as we wish, with each bag containing a slightly different version of the original training corpus, containing multiple copies of some sentences and no copies of others.

In the text of Chapter 5 we show the results achieved from generating multiple parsers via bagging and then combining the outputs of these parsers. We were consistently able to achieve an improvement in accuracy over the single parser trained from the original training set.

Future work

Several important problems have been identified during the work on the various parsers which warrant future research. The most challenging problem is the phenomenon of non-projectivity, or crossing dependency, which makes any parser based on a Context-Free Grammar theoretically insufficient. Also, the number of tags used for languages such as Czech is so high that we will have to work more on tag clustering or classification to obtain various clustering schemes suitable for various parsing techniques. Furthermore, the Czech tagger needs to be improved, especially for the number, gender and case categories, where it’s error rate of 4-6% is still too high for parsing purposes.

Certain theoretical questions will have to be solved, too: for example, is the tree representation as chosen really the best for statistical parsing? It seems that at least for some of the parsers, some changes would help, while preserving the dependency structure which we still believe is best suited for further processing in natural language understanding tasks (i.e. *after* parsing is done).

Adn of course, the more data, the better: the annotation of the PDT continues and we will certainly re-run the experiments we did during the Workshop ’98 when the PDT is finished and there is more than 1 mil. words for training.

Acknowledgements

This material is based upon work supported by the National Science Foundation under Grant No. (#IIS-9732388), and was carried out at the 1998 Workshop on Language Engineering, Center for Language and Speech Processing, Johns Hopkins University. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation or The Johns Hopkins University.

The resources (data and tools) used throughout the workshop are the results (in part or in whole, and on top of the funding provided by the workshop as described in the previous paragraph) of the following institutional grants (granted to the Institute of Formal and Applied Linguistcs, Faculty of Mathematics and Physics, Charles University, Prague, and lead by Prof. Eva Hajicova): Project No. VS96151 of the Ministry of Education of the Czech Republic, Grants No. 405/96/0198 and 405/96/K214 of the Grant Agency of the Czech Republic. The following individual grants have also contributed to the development of tools used at the workshop: Grant No. 39/94 of the Charles University, Prague, Czech Republic, and the Grant No. 195/1995 of the Research Support Scheme/HESP of the Soros' Open Society Fund.

The bilingual data used for the "Parsing by translation" task have been kindly prepared for use at the workshop by Martin Cmejrek and Jan Curin, now graduate students at the Institute of Formal and Applied Linguistics, Faculty of Mathematics and Physics, Charles University, Prague, Czech Republic. Special thanks to Reader's Digest VYBER (Prague, Czech Republic) for granting the licence for using their textual material.

The Prague Dependency Treebank data used at the workshop has been prepared and annotated by a team of people under the supervision of Jan Hajic (in alphabetical order): Alla Bemova, Eva Buranova, Jiri Karnik, Michal Kren, Petr Pajas, Jarmila Panevova, Jan Stepanek, and Zdenka Uresova. The morphological annotation of the data has been provided by a separate team, supervised by Barbora Hladka (in alphabetical order): Jiri Hana, Lenka Kebortova, Kristyna Kupkova, Pavel Kveton, Emil Jerabek, Jiri Mirovsky, Karel Skoupy and Stepan Verecky. Most of the technical work of the actual preparation of the data for the workshop has been done by (in alphabetical order): Alena Bohmova, Jan Hajic, Barbora Hladka, Kiril Ribarov and Dan Zeman, all of IFAL, Faculty of Mathematics and Physics, Charles University, Prague, before or during the workshop.

The Prague Dependency Treebank is based on raw texts provided by the Institute of Czech National Corpus, Faculty of Arts, Charles University, Prague, Czech Republic, which come from the following sources: Lidove Noviny Publishers, Mlada Fronta Publishers, Ringier CR, Academy of Science of the Czech Republic.

Last but not least, we are very grateful to the local organizers (above all to Jacob Laderman and Kimberly S. Petropoulos) who made all the technical and organizational arrangements necessary for a well-functioning and scientifically stimulating environment and thus made the Workshop'98 a very pleasant experience.