Towards English-to-Czech MT via Tectogrammatical Layer

Ondřej Bojar, Silvie Cinková, Jan Ptáček
Institute of Formal and Applied Linguistics
ÚFAL MFF UK, Malostranské náměstí 25
CZ-11800 Praha, Czech Republic
{bojar,cinkova,ptacek}@ufal.mff.cuni.cz

Abstract

We present an overview of an English-to-Czech machine translation system. The system relies on transfer at the tectogrammatical (deep syntactic) layer of the language description. We report on the progress of linguistic annotation of English tectogrammatical layer and also on first end-to-end evaluation of our syntax-based MT system.

1 Introduction

Current state of the art machine translation (MT) systems are statistical and mostly phrase-based[1]. In recent years the performance of (surface) syntax-based systems has improved and as a result are approaching state of the art performance levels (Zollmann and Venugopal, 2006; Quirk and Menezes, 2006; Chiang, 2005).

Our long-term goal is to improve English-Czech MT quality by introducing a transfer step at a deep syntactic layer, making explicit use of linguistic theories and annotated data. For the time being, parts of the annotated data as well as the whole pipeline of automatic deep syntactic analysis, syntactic transfer and a generation component are still very much work in progress. Nevertheless, we are able to deliver first end-to-end evaluation that will serve as a baseline for the future improvements of the system.

In Section 2, we give a brief overview of the tectogrammatical representation. Section 3 summarizes our ongoing efforts in developing and annotating English texts at the tectogrammatical layer. In Section 4, we describe both formal and implementational aspects of our MT system and Section 5 compares and discusses automatically assessed translation quality of several configurations of our system.

[1] See NIST evaluation: http://www.nist.gov/speech/tests/mt/

2 Overview of the Tectogrammatical Representation

2.1 Functional Generative Description and Treebank Annotation

The tectogrammatical language representation is an implementation of the Functional Generative Description (FGD, Sgall et al. (1986)). FGD has been implemented in treebank annotations. The Prague Dependency Treebank (PDT 2.0, Hajič et al. (2006)) consists of three interlinked annotation layers, corresponding to the three FGD-original levels: the morphological layer (m-layer; 2 million words), the analytical layer (a-layer, describing the surface syntax; 1.5 million words) and the tectogrammatical layer (t-layer; 0.8 million words).

The FGD as well as the treebank annotation focus on the tectogrammatical language (t-) level. Being a transition between syntax and semantics (sometimes also referred to as *underlying syntax/deep syntax*), the tectogrammatical language level captures the linguistic meaning of each sentence, describing mutual syntactic and semantic relations between the respective words in a sentence, including those of coreference and topic-focus articulation in a broader context scope. FGD has a strong valency theory (Panevová, 1980, 1974, 1975). The valency theory of FGD assigns valency frames to verbs, nouns, adjectives and certain types of adverbs, assigning semantic roles to their complementations.

2.2 Trees, Nodes and Edges

In the treebank annotation, every sentence is represented as a rooted dependency tree with labeled nodes and edges. The tree reflects the underlying (deep) structure of the sentence. Several types of edges specify whether the relation between two nodes is a dependency relation or not (e.g. the relation between the sentence predicate and an interjection or a disjunct is not that of dependency, although the predicate and the other node are connected by an edge).

Unlike the surface-syntax representation (a-layer), only autosemantic words[2] have their own nodes in the tectogrammatical tree structures. Function words like auxiliaries, subordinating conjunctions and prepositions as well as several cognitive, syntactic and morphological categories are attached to the respective nodes as a set of attribute-value pairs. The presence or absence of an attribute in a given node is determined by its node type.

2.3 Valency

Each occurrence of a part of speech that is considered to have valency is assigned a valency frame from a valency lexicon, interlinked with the data[3]. Obligatory complementations that are not present in the surface representation of the sentence get

[2] Several artificially generated complementary nodes for coordination, apposition, reciprocity, etc., and the technical root node also have their own t-nodes, although they do not necessarily have a corresponding node in the surface structure.

[3] This is restricted to verbs and certain types of nouns in the current annotation.

their tectogrammatical representations by means of artificially added nodes. These nodes specify whether the missing information can be retrieved from the context (anaphora/cataphora, textual ellipsis) or whether it is only implied by common knowledge.

2.4 Machine Translation via Tectogrammatical Layer

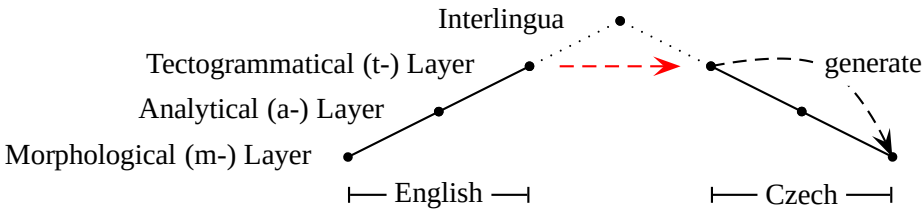
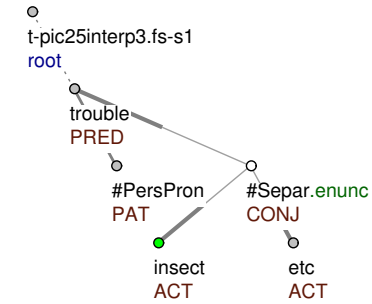


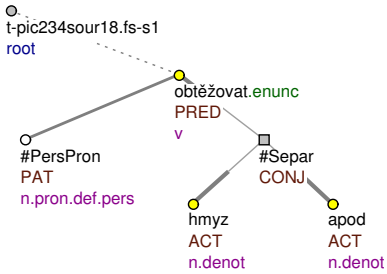
Figure 1: MT via tectogrammatical annotation.

Figure 1 illustrates the big picture of our MT system. The rationale to introduce additional layers of formal language description is to bring the source and target language closer to each other (see Figure 2). If the layers are designed appropriately, the transfer step will be easier to implement because (among others):

- t-structures exhibit less divergences, fewer structural changes will be needed in the transfer step.
- t-nodes correspond to autosemantic words only, all auxiliary words are identified in the source language and generated in the target language using language-dependent grammatical rules between t- and a- layers.
- t-nodes contain word lemmas, the whole morphological complexity of either of the languages is handled between m- and a- layers.
- t-layer abstracts away word-order issues, explicitly encoding topic-focus articulation (given/new) in node order.



He was troubled by insects etc.



Obtěžoval ho hmyz apod.
Troubled him insects etc.

Figure 2: A pair of English and Czech t-trees of the same sentence.

3 English Tectogrammatical Layer: Ongoing Work

3.1 Prague Czech-English Dependency Treebank

The tectogrammatical representation remains in many concrete annotation decisions language-specific. Though, its basic concepts are believed to be applicable to most languages. To prove this assumption, a parallel Czech-English treebank is being built. The Prague Czech-English Dependency Treebank (PCEDT 2.0) is based on PCEDT 1.0 (Cuřín et al., 2004), which comprises the Penn Treebank II - Wall Street Journal section (Marcus et al., 1994) converted into dependency trees on the a-layer, and a corpus of its Czech translations, parsed in the same way as PDT 1.0 (Hajič et al., 2001) was. As PDT 2.0 came into existence, the parallel texts were re-parsed to comply with the new format of PDT 2.0, and manual annotation of the automatically pre-processed t-layer trees was launched for both languages.

3.2 Prague English Dependency Treebank

The English counterpart (referred to as the Prague English Dependency Treebank, PEDT) comprises approx. 50 000 dependency trees, which have been obtained by an automatic conversion of the original Penn Treebank II constituency trees into FGD-compliant a-layer trees. These a-layer trees have been automatically converted into t-layer trees. EngVallex (Cinková, 2006), a valency lexicon of verbs contained in PTB-WSJ, was obtained by a semi-automatic conversion of the PropBank-Lexicon (Palmer et al., 2005, 2004) into an FGD-compliant valency lexicon (following the structure of the Czech PDT-Vallex (Hajič et al., 2003)) and its manual adjustment.

3.3 Annotation Manual

Three annotators and a coordinator have been working on the adaptation of the Czech annotation guidelines into English. Recently an annotation manual for the English tectogrammatical representation was released (Cinková et al., 2006)[4]. So far, the annotation has concentrated on the following issues:

1. correct tree structure, including but not limited to:
   (a) rules for coordination, apposition, parenthesis
   (b) some specific constructions like comparison, restriction, consecutive clauses with quantifiers etc.
   (c) determination of function words
2. assigning and completing valency frames in verbs
3. correct semantic labels (functors) in nodes
4. correct t-lemmas
5. correct links to a-layer

[4] http://ufal.mff.cuni.cz/~cinkova/TR_En.pdf

The following issues have been left aside for the moment:

1. coreference
2. topic-focus articulation
3. more fine-grained attributes in nodes (subfunctors, grammatemes)

3.4 Annotation Process

Three Czech annotators had first been trained in the Czech annotation and their proficiency in English had been checked before entering the English annotation. The annotation tool TrEd[5], used in the Czech annotation, was adopted to the specific features of the English annotation. Later on, the two configurations were re-unified to make it possible for the annotators to switch languages without having to learn two different ways of annotation with TrEd. This preparatory stage lasted from spring to fall 2006. The actual annotation was launched in September 2006.

The annotators are supposed to deliver 500 trees per month including the test files for agreement measurements, which should ensure about one half of PTB-WSJ to be manually annotated by 2008. Being slightly behind the schedule, we decided to appoint another annotator, who is now being trained. Simultaneously, special attention is being paid to tree pre-processing in order to decrease the extent of the manual annotation work. As the annotation manual has become quite stable now it is possible to formulate additional rules for the conversion of the original constituency trees into tectogrammatical trees, exploiting the rich original linguistic markup of PTB-WSJ in more depth than done so far, e.g. regarding cleft sentences and verb control.

4 Tree-to-tree Transfer

4.1 Synchronous Tree Substitution Grammars

Synchronous Tree Substitution Grammars (STSG) were introduced by Hajič et al. (2002) and formalized by Eisner (2003) and Čmejrek (2006). They formally capture the basic assumption of syntax-based MT that a valid translation of an input sentence can be obtained by local structural changes of the input syntactic tree (and translation of node labels). Some training sentences may violate this assumption because human translators do not always produce literal translations but we are free to ignore such sentences.

As illustrated in Figure 3, STSG describe the tree transformation process using the basic unit of *treelet pair*. Both source and target trees are decomposed into treelets that fit together. Each treelet can be considered as representing the minimum translation unit. A treelet pair such as depicted in Figure 4 represents the structural and lexical changes necessary to transfer local context of a source tree into a target tree.

[5] `http://ufal.mff.cuni.cz/pdt2.0/doc/tools/tred/`

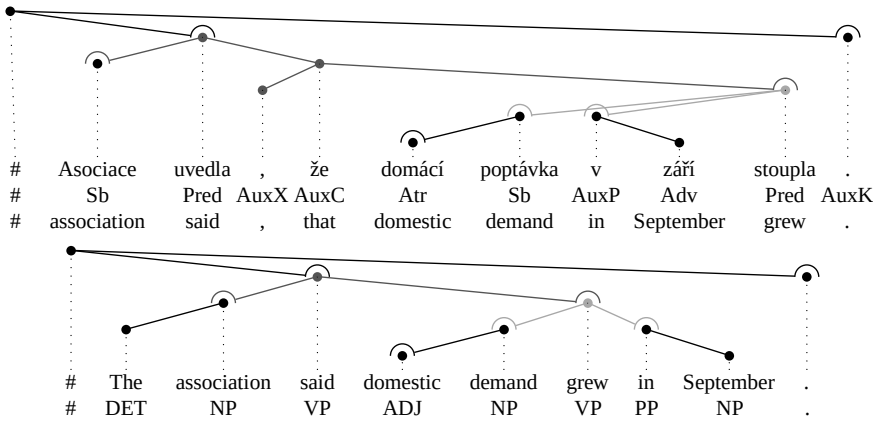


Figure 3: A sample pair of analytical trees synchronously decomposed into treelets.

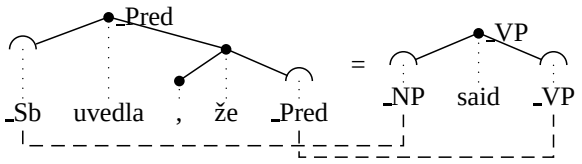


Figure 4: Sample analytical treelet pair.

Each node in a treelet is either *internal* (• , constitutes treelet internal structure and carries a lexical item) or *frontier* (⌒ , represents an open slot for attaching another treelet). Frontier nodes are labelled with *state labels* (such as "_Sb" or "_NP"), as is the root of each treelet. A treelet can be attached at a frontier node only if its root state matches the state of the frontier.

A *treelet pair* describes also the *mapping* of the frontier nodes. A pair of treelets is always attached synchronously at a pair of matching frontier nodes.[6]

Depending on our needs, we can encode ordering of nodes as part of each treelet. If only local ordering is used (i.e. we record the position of a parent node among its sons), the output tree will be always projective. If we record global ordering of all nodes in a treelet, the final output tree may contain non-projectivities introduced by non-projective treelets (the attaching operation itself is assumed to be projective).

STSG is generic enough to be employed at or across various layers of annotation (e.g. English t-tree to Czech t-tree or English a-tree to Czech a-tree). Our primary goal is to perform transfer at the tectogrammatical layer.

[6]We depart from Čmejrek (2006) in a few details of the definition. Most notably, we require (1) each treelet to contain at least one internal node and (2) all frontier nodes in a treelet pair to be mapped, i.e. the left and right treelets must contain the same number of frontier nodes.

4.2 STSG Decoder

The task of STSG "decoder" is to find the most likely target tree, given a source tree and a dictionary of treelet pairs.

Our current version of the decoder considers all possible decompositions of input tree. We traverse the input tree top-down, using the dictionary of treelet pairs to produce the output tree by attaching corresponding right hand treelets to open frontiers. Another option is to traverse the tree in bottom-up fashion in a parsing-like algorithm, as sketched in Čmejrek (2006).

4.3 Estimating STSG Model Parameters

Eisner (2003) and Čmejrek (2006) provide formal details and expectation-maximization algorithms for training STSG using a parallel treebank. Our plan is to soon adopt this method, but for the time being we restrict our training method to a heuristic based on GIZA++ (Och and Ney, 2000) word alignments.

For each tree pair in the training data, we first read off the sequence of node labels and use GIZA++ tool to extract a possibly N-N node-to-node-alignment. Then we extract all treelet pairs from each aligned tree pair such that all the following conditions are satisfied:

- each treelet may contain at most 5 internal and at most 7 frontier nodes (the limits are fairly arbitrary),
- each internal node of each treelet, if aligned at all, must be aligned to a node in the other treelet,
- the mapping of frontier nodes has to be a subset of the node-alignment,
- each treelet must satisfy STSG property: if a node in the source tree is used as an internal node of the treelet, all immediate dependents of the node have to be included in the treelet as well (either as frontier or internal nodes). In other words, we assume no tree adjuction operation was necessary to construct the training sentence.

All extracted treelet pairs and basic co-occurrence statistics constitute our "translation table".

4.4 Methods of Back-off

As expected, and also pointed out by Čmejrek (2006), the additional structural information boosts data-sparseness problem. Many source treelets in the test corpus were never seen in our training data. To tackle the problem, our decoder utilizes a sequence of back-off models, i.e. a sequence of several translation tables where each subsequent table is based on less fine-grained description of the input tree.

Given a source treelet, we first search an "exact-match" translation table. If no translation candidate can be found, we disregard some of the detailed node attributes (such as verbal tense etc.) in the source treelet and search correspondingly reduced translation table. We also experiment with an alternative direction of

source treelet simplification: we keep the full detail of internal nodes but remove all frontier nodes. When a target treelet is found (with no frontier nodes, because the source treelet we searched for had no frontier nodes either), we insert the original number of frontier nodes on the fly, guessing both their position in the treelet and their label using simple local statistics. As a last resort back-off, we keep the internal nodes in the source treelet untranslated and just guess target-side labels of all frontiers. The order and level of detail of the back-off methods is fixed but easily customizable in a configuration file.

4.5 Generating Surface from Czech Tectogrammatical Trees

The purpose of the generation component is to express the meaning given by the target t-tree in a sentence of target language. In the terms of Figure 1, our objective is the transition given by the right side of the translation triangle.

We decompose the generation into sequence of seven linguistically motivated steps: Formeme Selection, Agreement, Adding Functional Words (prepositions, subordinating conjunctions and other auxiliaries), Inflexion, Word Order, Punctuation and Vocalization. During each step the input t-tree is gradually changing - new node attributes and/or new nodes are added. After the last step, the nodes are ordered appropriately and each node bears a computed word form. The resulting sentence is then simply obtained by concatenation.

The Formeme Selection phase is where the syntactic shape of the final sentence is grounded. The input t-tree is traversed in depth-first fashion and a suitable morphosyntactic (surface) form is selected for each node. From the full repertoire of surface forms available in Czech language, a subset was selected and is implemented in the generator. Surface forms are identified in the system by a distinguishable label, which we call *formeme*. The formeme is stored as an attribute of a t-node once particular surface realization is picked out. Possible formeme values are for instance: simple case *gen* (genitive case), prepositional case *pod+7* (preposition *pod/under* and instrumental case), *adj* (syntactic adjective), *že+v-fin* (subordinating clause introduced with subordinating conjunction *že*), etc.

Surface forms suitable for a particular t-node are restricted both by syntax and semantics. The syntactic nature is given by the governor's and its own part of speech. As far as semantics is concerned, a particular choice of meaning-bearing preposition or subordinate conjunction is determined by a attribute of t-node called functor. Additional constraints can also be specified in a valency frame of t-node's governor; the frame is picked up from a valency dictionary. The six remaining steps of generation procedure materialize the syntactic and morphological aspects prescribed by the formeme.

Computation of word forms is accomplished using morphological tools by Hajič (2004). Vocalization rules specifying whether to append a vowel *-e/-u* to selected prepositions for easier pronunciation are based on Petkevič (1995). A detailed description of the generation component is given in (Ptáček and Žabokrtský, 2006).

5 Experimental Results

Table 1 reports the BLEU (Papineni et al., 2002) scores of several configurations of our system. For the purposes of comparison with a phrase based system tuned for English-to-Czech, we train and test our system on the News Commentary corpus as available for the ACL 2007 workshop on machine translation (WMT)[7]. We report single-reference lowercased BLEU[8,9].

The values in column Generation indicate how strongly is the final production of string of words driven by an n-gram language model (LM). For phrase-based approaches, LM is a vital component. For our transfer to Czech a-layer, our decoder uses LM to score partial trees when enough consecutive internal nodes have been established. The generation component described in Section 4.5 employs no LM and has no access to the target side of the training corpus.

Transfer Mode	Generation	Dev	DevTest
English t → Czech t preserving structure	rule-based	5.38±0.43	5.12±0.49
English t → Czech t changing structure	rule-based	5.14±0.43	4.74±0.46
English t → Czech a	LM-guided	7.01±0.50	6.27±0.56
English a → Czech t	rule-based	3.21±0.37	3.18±0.35
English a → Czech a	LM-guided	9.88±0.58	8.61±0.57
Phrase-based as reported by Bojar (2007)			
Vanilla	LM-driven	-	12.9±0.6
Factored to improve target morphology	LM-driven	-	14.2±0.7

Table 1: Preliminary English-to-Czech BLEU scores for syntax-based MT evaluated on Dev and DevTest datasets of ACL 2007 WMT shared task.

5.1 Discussion and Future Research

At the first sight, our preliminary results support common worries that with a more complex system it is increasingly difficult to obtain good results. However, we are well aware of many limitations of our current experiments:

1. BLEU is known to favour methods employing n-gram based language models (LMs). In future experiments we plan to attempt both, employing some LM-based rescoring when generating from the t-layer, as well as using other automatic metrics of MT quality.

[7] `http://www.statmt.org/wmt07/`

[8] For methods using the generation system as described in section 4.5, we tokenize the hypothesis and the reference using the rules from the official NIST `mteval-v11b.pl` script. For methods that directly produce sequence of output tokens, we stick to the original tokenization.

[9] The reported ± bounds indicate empirical 95% confidence intervals obtained using bootstrapping method by Koehn (2004).

2. All components in our setup deliver only the single best candidate. Any errors will therefore accumulate over the whole pipeline. In future, we would like to pass and accept several candidates, allowing each step in the calculation to do any necessary rescoring.
3. The rule-based generation system was designed to generate from full-featured manual Czech tectogrammatical trees from the (monolingual) PDT. There are so far no manual Czech trees for a parallel corpus. Our target-side training trees are the result of an automatic analytical and tectogrammatical parsing procedure as implemented by McDonald et al. (2005) and Klimeš (2006), resp. The errors in automatic target-side training trees, together with errors in the tree-to-tree transfer process, pose new challenges to the generation system. A more thorough analysis of which component causes most frequent errors will still have to be done.
4. For the purposes of source-side English analysis, we still rely on simple rules similar to those used by Čmejrek et al. (2003) to convert Collins (1996) parse trees to analytical and tectogrammatical dependency trees. We hope to improve the English-side pipeline soon, using recent parsers and improved tectogrammatical analysis, based on the PEDT manual t-trees described above.

Surprisingly, preserving the structure of English t-tree achieves (insignificantly) better BLEU score than allowing the decoder to use larger treelets to produce structurally different Czech t-trees. One possible explanation is that our current heuristic tree-alignment method performs poorly for t-trees. For all other modes of transfer (t→a, a→t, a→a), tree structure modifications gain significant improvements and we use them.

6 Conclusion

We have described the current status of our ongoing effort to translate from English to Czech via deep syntactic (tectogrammatical) structure. The process involves adaptation of the tectogrammatical layer definition for English, parallel treebank annotation and automatic procedures of source sentence analysis, tree-based transfer and target sentence generation.

Our first empirical results do not reach the phrase-based benchmark and we give several reasons why this is the case. However, the presented system is a finished pipeline that establishes a baseline and allows to evaluate how modifications to individual components influence the end-to-end performance in syntax-based machine translation.

7 Acknowledgment

The work on this project was partially supported by the grants FP6-IST-5-034291-STP (EuroMatrix), GA405/06/0589, 1ET101120503, 1ET201120505, and GAUK 7643/2007. We would like to thank Zdeněk Žabokrtský for his rules performing automatic annotation of English t-layer.

References

Ondřej Bojar. 2007. English-to-Czech factored machine translation. In *Proc. of the ACL Second Workshop on Statistical Machine Translation*, pages 232–239.

David Chiang. 2005. A hierarchical phrase-based model for statistical machine translation. In *Proc. of ACL*, pages 263–270.

Silvie Cinková. 2006. From PropBank to EngValLex: Adapting the PropBank-Lexicon to the Valency Theory of the Functional Generative Description. In *Proc. of LREC*, pages 2170–2175.

Silvie Cinková, Jan Hajič, Marie Mikulová, Lucie Mladová, Anja Nedolužko, Petr Pajas, Jarmila Panevová, Jiří Semecký, Jana Šindlerová, Josef Toman, Zdeňka Urešová, and Zdeněk Žabokrtský. 2006. Annotation of English on the tectogrammatical level. Technical report, ÚFAL MFF UK.

Martin Čmejrek. 2006. *Using Dependency Tree Structure for Czech-English Machine Translation*. Ph.D. thesis, ÚFAL, MFF UK, Prague, Czech Republic.

Martin Čmejrek, Jan Cuřín, and Jiří Havelka. 2003. Czech-English Dependency-based Machine Translation. In *Proc. of EACL*, pages 83–90.

Michael Collins. 1996. A New Statistical Parser Based on Bigram Lexical Dependencies. In *Proc. of ACL*, pages 184–191.

Jan Cuřín, Martin Čmejrek, Jiří Havelka, Jan Hajič, Vladislav Kuboň, and Zdeněk Žabokrtský. 2004. Prague Czech-English Dependency Treebank Version 1.0. LDC2004T25, ISBN: 1-58563-321-6.

Jason Eisner. 2003. Learning non-isomorphic tree mappings for machine translation. In *Proc. of ACL, Companion Volume*, pages 205–208.

Jan Hajič, Eva Hajičová, Petr Pajas, Jarmila Panevová, Petr Sgall, and Barbora Vidová Hladká. 2001. Prague Dependency Treebank 1.0. LDC2001T10, ISBN: 1-58563-212-0.

Jan Hajič, Jarmila Panevová, Zdeňka Urešová, Alevtina Bémová, Veronika Kolářová-Řezníčková, and Petr Pajas. 2003. PDT-VALLEX: Creating a Large-coverage Valency Lexicon for Treebank Annotation. In J. Nivre and E. Hinrichs, editors, *Proc. of The Second Workshop on Treebanks and Linguistic Theories*, pages 57–68. Vaxjo University Press, Vaxjo, Sweden.

Jan Hajič. 2004. *Disambiguation of Rich Inflection – Computational Morphology of Czech*. Charles University – The Karolinum Press, Prague.

Jan Hajič, Jarmila Panevová, Eva Hajičová, Petr Sgall, Petr Pajas, Jan Štěpánek, Jiří Havelka, Marie Mikulová, Zdeněk Žabokrtský, and Magda Ševčíková Razímová. 2006. Prague Dependency Treebank 2.0. LDC2006T01, ISBN: 1-58563-370-4.

Jan Hajič, Martin Čmejrek, Bonnie Dorr, Yuan Ding, Jason Eisner, Daniel Gildea, Terry Koo, Kristen Parton, Gerald Penn, Dragomir Radev, and Owen Rambow. 2002. Natural Language Generation in the Context of Machine Translation. Technical report. NLP WS'02 Final Report.

Václav Klimeš. 2006. *Analytical and Tectogrammatical Analysis of a Natural Language*. Ph.D. thesis, ÚFAL, MFF UK, Prague, Czech Republic.

Philipp Koehn. 2004. Statistical Significance Tests for Machine Translation Evaluation. In *Proc. of EMNLP*.

M. Marcus, G. Kim, M. Marcinkiewicz, R. MacIntyre, A. Bies, M. Ferguson, K. Katz, and B. Schasberger. 1994. The Penn treebank: Annotating predicate argument structure. In *Proc. of ARPA Human Language Technology Workshop*.

Ryan McDonald, Fernando Pereira, Kiril Ribarov, and Jan Hajič. 2005. Non-Projective Dependency Parsing using Spanning Tree Algorithms. In *Proc. of HLT/EMNLP*.

Franz Josef Och and Hermann Ney. 2000. A comparison of alignment models for statistical machine translation. In *Proc. of ACL*, pages 1086–1090.

Martha Palmer, Daniel Gildea, and Paul Kingsbury. 2005. The Proposition Bank: An Annotated Corpus of Semantic Roles. *Computational Linguistics*, 31(1):71–106.

Martha Palmer, Paul Kingsbury, Olga Babko-Malaya, Scott Cotton, and Benjamin Snyder. 2004. Proposition Bank I. LDC2004T14, ISBN: 1-58563-304-6.

Jarmila Panevová. 1974. On verbal frames in Functional Generative Description I. *Prague Bulletin of Mathematical Linguistics*, (22):3–40.

Jarmila Panevová. 1975. On verbal frames in Functional Generative Description II. *Prague Bulletin of Mathematical Linguistics*, (23):17–52.

Jarmila Panevová. 1980. *Formy a funkce ve stavbě české věty*. Prague:Academia.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a Method for Automatic Evaluation of Machine Translation. In *Proc. of ACL*, pages 311–318.

Vladimír Petkevič, editor. 1995. *Linguistic Problems of Czech*, chapter Vocalization of Prepositions, pages 147–157.

Jan Ptáček and Zdeněk Žabokrtský. 2006. Synthesis of Czech Sentences from Tectogrammatical Trees. In *Proc. of TSD*, pages 221–228.

Christopher Quirk and Arul Menezes. 2006. Dependency treelet translation: the convergence of statistical and example-based machine-translation? *Machine Translation*, 20(1):43–65.

Petr Sgall, Eva Hajičová, and Jarmila Panevová. 1986. *The Meaning of the Sentence in Its Semantic and Pragmatic Aspects*. Dordrecht:Reidel Publishing Company and Prague:Academia.

Andreas Zollmann and Ashish Venugopal. 2006. Syntax Augmented Machine Translation via Chart Parsing. In *Proc. of the ACL Workshop on Statistical Machine Translation*, pages 138–141.