Annotation of Sentence Structure; Capturing the Relationship among Clauses in Czech Sentences

Markéta Lopatková and **Natalia Klyueva** and **Petr Homola**
Charles University in Prague, Institute of Formal and Applied Linguistics
Malostranské nám. 25, 118 00 Prague 1, Czech Republic
{lopatkova,kljueva,homola}@ufal.mff.cuni.cz

Abstract

The goal of the presented project is to assign a structure of clauses to Czech sentences from the Prague Dependency Treebank (PDT) as a new layer of syntactic annotation, a layer of clause structure. The annotation is based on the concept of segments, linguistically motivated and easily automatically detectable units. The task of the annotators is to identify relations among segments, especially relations of super/subordination, coordination, apposition and parenthesis. Then they identify individual clauses forming complex sentences.

In the pilot phase of the annotation, 2,699 sentences from PDT were annotated with respect to their sentence structure.

1 Motivation

Syntactic analysis of natural languages is the fundamental requirement of many applied tasks. Parsers providing automatic syntactic analysis are quite reliable for relatively short and simple sentences. However, their reliability is significantly lower for long and complex sentences, especially for languages with free word order; see, e.g., Zeman (2004) for results for Czech.

The identification of the overall structure of a sentence prior to its full syntactic analysis is a natural step capable to reduce the complexity of full analysis. Such methods brought good results for typologically different languages, see e.g. Jones (1994) for English or Ohno et al. (2006) for Japanese.

The goal of the presented project is to annotate a structure of clauses to Czech sentences from the Prague Dependency Treebank. The main idea is to reuse the already existing language resource and to enrich it with a new layer of annotation, a layer of clause structure.

We exploit a concept of segments, easily automatically detectable and linguistically motivated units, as they were defined by Lopatková and Holan (2009).[1] The annotation captures relationship among segments, especially subordination, coordination, apposition and parenthesis. Based on segment annotation, the annotators identify clauses forming (complex) sentences: they group the segments constituting individual clauses of complex sentences.

Contrary to such well known approaches as e.g. chunking, see Abney (1991) or cascaded parsing, see Abney (1995) or Ciravegna and Lavelli (1999), which group individual tokens into more complex structures as nominal or prepositional phrases, i.e., in a bottom-up direction, the proposed approach aims at determining a hierarchy of sentence parts in a 'top-down' way. Such an approach is quite novel not only for Czech, it has not been reported for other Slavic languages.

Prague Dependency Treebank[2] (PDT), see Hajič et al. (2006) is a large and elaborated corpus with rich syntactic annotation of Czech newspaper texts. As the dependency-based framework has been adopted for PDT, the treebank contains explicit information on mutual relations among individual tokens (words and punctuation marks). However, relations among more complex units, esp. clauses, are not explicitly indicated, see Figure 1.

Syntactic information stored in PDT can be used (at least to some extent) for the identification of individual clauses as well. Let us refer to the experiments described in the papers by Lopatková and Holan (2009) and Krůza and Kuboň (2009). In both papers, the authors designed well-developed procedures for identifying segments and their mu-

[1] We adopt the basic idea of segments introduced and used by Kuboň (2001) and Kuboň et al. (2007). We slightly modify it for the purposes of the annotation task.

[2] http://ufal.mff.cuni.cz/pdt2.0/

Proceedings of the Third Linguistic Annotation Workshop, ACL-IJCNLP 2009, pages 74–81,
Suntec, Singapore, 6-7 August 2009.

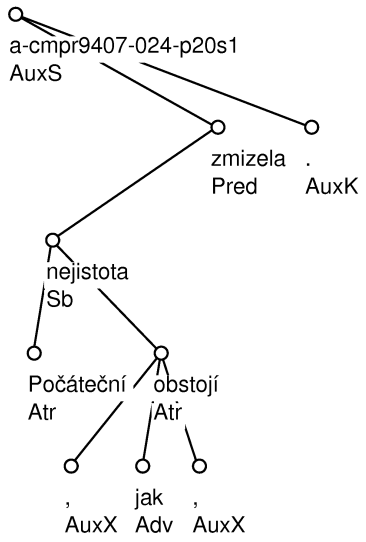

Figure 1: Analytic tree of the sentence *Počáteční nejistota, jak obstojí, zmizela.* 'Initial uncertainty, how it-will-do, vanished.'

tual relationship from the analytical layer of PDT (i.e., layer of surface syntax). However, they either do not identify individual clauses in the complex sentence at all, or their procedural definition of clause does not exactly model what a human would consider as a clause.

The previous experiments brought clear specification of segmentation charts describing the relation among individual segments. The results showed that for further research it is necessary to work with a large set of precisely annotated data. It has turned out that such data cannot be obtained without extensive (semi)manual annotation of a large set of sentences, see Lopatková and Holan (2009) and Krůza and Kuboň (2009).

In this article, we present a project of manual annotation of sentence structure for complex Czech sentences. In Section 2, we introduce the basic concepts, esp. boundaries, segments and segmentation charts. Then we focus on the annotation of basic linguistic phenomena (Section 3). Section 4 brings specification of a data format and an editor used for the annotation. Lastly, basic statistics of the annotated data are presented (Section 5).

2 Boundaries, Segments and Segmentation Charts

The aim of the annotation is to explicitly describe relations among clauses of (complex) Czech sentences. We focus on the annotation of (part of) Czech sentences from PDT. We take advantage of morphological analysis (m-layer) and partially also surface syntactic analysis (a-layer) stored in PDT.

All tokens from PDT are disjunctively divided into two groups – ordinary words and segment boundaries. *Segment boundaries* are tokens and their sequences that divide a sentence into individual units referred to as segments. As segment boundaries, the following tokens are considered:

- punctuation marks: comma, colon, semicolon, question mark, exclamation mark, dash (all types), opening and closing bracket (all kinds), and quotation mark (all types);
- coordinating conjunctions: tokens morphological tag of which starts with the pair J^ (e.g., *a* 'and', *ale* 'but', *nebo* 'or', *neboť* 'for', *ani* 'nor'), see Hajič (2004).

After the identification of boundaries, the input sentence is partitioned into individual segments – a *segment* is understood as a maximal non-empty sequence of tokens that does not contain any boundary.

This concept of the linear segment serves as a good basis for the identification of clauses, basic linguistically motivated syntactic units. We will see that a single *clause* consists of one or more segments; one or more clauses then create(s) a *complex sentence* (see Section 3).

The definition of segments adopted in this project is based on very strict rules for punctuation in Czech. Generally, beginning and end of each clause must be indicated by a boundary, i.e., sentence boundary (usually fullstop, question mark or exclamation mark), punctuation (mostly comma) or conjunction. This holds for embedded clauses as well. In particular, there are only very few exceptions to a general rule saying that there must be some kind of a boundary between two finite verb forms of meaningful verbs.

Segmentation Charts and Clauses

Relations between clauses, esp. super- or subordination, coordination, apposition or parenthesis, are described by so called *segmentation charts* (one or more, if we allow for ambiguous annotation) – segmentation chart captures the levels of embedding for individual segments, as described below.

The principal idea of the segmentation chart is quite clear – it can be described by the following basic instructions. (In examples, segments are marked by square brackets [and $]_k$, where k is a level of embedding. In addition, individual clauses are marked by brackets { and $\}_j$, where j is an index of a particular clause.)

Main clauses. Segments forming all main clauses[3] of a complex sentence belong to the basic level (level of embedding 0), as in the following sentence.
$\{[$*O studium byl velký zájem*$]_0\}_1$, $\{[$*v přijímacích pohovorech bylo vybráno 50 uchazečů*$]_0\}_2$. 'There was a lot of interest in studying, 50 applicants were selected in admission interviews.'

Dependent clauses. Segments forming clauses that depend on clauses at the k-th level obtain level of embedding $k + 1$ (i.e., the level of embedding for subordinated segments is higher than the level of segments forming their governing clause).
$\{[$*Potom zjistíte*$]_0\}_1$, $\{[$ *že vám nikdo nedá vstupní vízum*$]_1\}_2$. 'Then you realize that nobody gives you entrance visa.'

Coordination and apposition. Segments forming coordinated sentence members and coordinated clauses occupy the same level. The same holds for apposition.
$\{[$*Hra nám jde*$]_0\}_1$ *a* $\{[$*forma stoupá*$]_0\}_1$. 'We're getting on well in game and our form improves.'

Parenthesis. Segments forming parenthesis (e.g., sequence of wordforms within brackets) obtain the level of embedding $k + 1$ if the level of their neighboring segments is k .
$\{[$*Návrh mluví o dvou letech u mužů*$]_0$ *(* $\{[$*zvyšuje věk z 60 na 62*$]_1\}_1$ *)* *a* $[$*o čtyřech letech u žen*$]_0\}_2$.
'The proposal mentions two years for men (it raises the age from 60 to 62) and four years for women.'

Although this basic idea of segmentation charts seems simple, it appears that – working with 'real data' from newspaper corpus – detailed annotation guidelines are necessary for good and consistent annotation of specific linguistic phenomena and especially for their combination. We focus on some of them in the following section.

[3] As a main clauses, such clauses are considered that are *syntactically / formally* independent, see also Section 3.

3 Annotation of Complex Sentences

Segments can be divided into two main groups, mutually independent and mutually related segments.

Mutually independent segments. Mutually independent segments are, e.g., segments forming two dependent clauses, each of them modifying (different) part of the main clause, as segments *do které se zamiloval* 'with whom he felt in love' and *který zazvonil* 'that rang' in the following sentence.
$\{[$*Marie*$]_0$, $\{[$*do které se zamiloval*$]_1\}_1$, $\{[$*když ji potkal*$]_2\}_2$, $[$*zvedla telefon*$]_0\}_3$, $\{[$*který zazvonil*$]_1\}_4$. 'Mary, with whom he felt in love when he met her, answered the phone that rang.'

Such segments can have the same level of embedding (as the above mentioned segments) or they can belong to clauses with different levels of embedding (as segments *když ji potkal* 'when he met her' and *který zazvonil* 'that rang').

Mutually related segments. Mutually related segments either belong to different levels of embedding – they are super- or subordinated, we focus on this group in the following Section 3.1, or they have the same level of embedding – this type is described in Section 3.2.

Let us stress here that the segment annotation is based on *formally expressed* structures rather than on their semantic interpretation. For example, we do not interpret text enclosed in brackets – whether it is semantically apposition, sentence member or independent sentence part, see also the discussion in Kuboň et al. (2007). We annotate such text as parenthetical segment(s) on a lower level compared to the neighboring segments.

The annotators have been instructed to *disambiguate annotated* sentences – if more readings of a particular sentence are possible, they should respect the reading rendered in PDT.

3.1 Subordination and Superordination

The super- or subordinated mutually related segments capture primarily **relations between governing and dependent clauses**.

Identification of subordinated status of a particular segment is based on morphological properties of tokens forming this segment, i.e., on the presence of a token with 'subordinating function'.

'Subordinating tokens' are especially of the following types:

- subordinating conjunctions (e.g., *aby* 'in order that', *dokud* 'till', *kdyby* 'if', *protože* 'because', *přestože* 'although', *že* 'that');
- relative/interrogative pronouns and some types of numerals (e.g., *kdo* 'who', *co* 'what', *jaký* 'which', *kolik* 'how many');
- pronominal adverbs (e.g., *kde* 'where', *kdy* 'when', *jak* 'how', *proč* 'why').

In Czech, a subordinating token is usually at the beginning of the segment, as in the following sentence (marked pronoun *který* 'who' serves as subordinating token here).
{[*Klejch*]$_0$, {[*který dal devět ze dvanácti ligových gólů Zlína*]$_1$}$_1$, [*má vydatné pomocníky*]$_0$}$_2$. 'Klejch who scored nine goals out of twelve for Zlín has good helpers.'

A particular subordinated segment can precede or follow its superordinated segment or it can be placed between two superordinated segments (in case of a governing clause with embedded dependent clause, as in the previous example).

In addition to governing and dependent clauses, there are also other constructions that should evidently be captured as subordinated segments, especially:

- Segments representing **direct speech**:
  „{[*Kupříkladu závod Ejpovice projevil zájem dokonce o 150 pracovníků*]$_1$}$_1$,"{[*uvedl Ladislav Vltavský*]$_0$}$_2$. ' "For example Ejpovice company showed interest in 150 workers," said Ladislav Vltavský.'
- Some types of **parenthesis**, esp. those marked by brackets:
  {[*Guido Reni*]$_0$ *(* {[*1575 až 1642*]$_1$}$_1$ [*byl vynikající figuralista*]$_0$}$_2$. 'Guido Reni (1575 to 1642) was an outstanding figural painter.' In such cases, parenthetical expressions are captured as separate clauses even if they consist of fragmental expression.

3.2 Segments on the Same Level and Identification of Clauses

We can identify three main groups of structures where segments are mutually related and they share the same level of embedding:

- **segments forming a clause with embedded dependent clause**, as the attributive dependent clause in the following example.
  {[*V případě*]$_0$, {[*že se nedovoláte*]$_1$}$_1$, [*vytočte číslo ve večerních hodinách znovu*]$_0$}$_2$. 'In case that you will not succeed, redial the number again in the evening.'
- coordinated segments (see the corresponding section below);
- others, esp. segments in apposition and some types of parenthesis (see the corresponding section below).

In particular, segments on the same level – unlike the super/subordinated ones – can form a single clause. For the annotators, the important task is to *identify individual clauses*. They group those segments that constitute individual clauses of a complex sentence and thus mark them as a single syntactic unit of a higher level, level of clause structures. (Let us recall that clauses are marked here by brackets { and }$_j$ where j is an index of a particular clause).

Coordination of sentence members and coordination of clauses

The relation of coordination may occur between two (or more) sentence members or between two (or more) clauses, be they main clauses or dependent ones. The syntactic position of coordinated units is 'multiplied', that is, they share the same syntactic relations to other sentence members. The annotators have to identify segments containing coordinated sentence members and put them together into a single clause; coordinated clauses are marked as separate clauses sharing the same level of embedding,[4] as in the following sentence.
{[*Český prezident apeloval na Čechy*]$_0$ *a* [*na Němce*]$_0$}$_1$, {[*aby odpovědně zacházeli s minulostí*]$_1$}$_2$ *a* {[*aby posouvali vpřed dialog*]$_1$ *a* [*spolupráci.*]$_1$}$_3$ 'Czech president appealed to Czechs and Germans that they should treat their history responsibly and improve their mutual dialogue and cooperation.' This complex sentence consists of five segments (marked by [and]), which form three clauses (marked by { and }), namely one main clause (on the zero level) and two coordinated dependent clauses (first embedded level), see also Figure 3.

[4] In PDT, coordination of sentence members and coordination of clauses are not distinguished (at the analytical layer).

Segmentation is purely linear (on segment follows another); after the identification of segments, they are grouped into the clauses. As we have seen, a single clause consists (prototypically) of one or more segments. This is fully true for semantically and syntactically complete sentences, i.e. sentences without ellipses of different kinds.

Let us mention one construction where clauses identified by the annotators (i.e., clauses based on segments) do not conform with the linguistic intuition, namely the case of coordinated clauses sharing one (or more) sentence member(s) or a syntactic particle. We interpret such cases as cases of ellipses, i.e., a shared sentence member or particle is supposed to belong only to one of the clauses and to be elided in the second clause. Thus a shared sentence member or particle is annotated only as a part of one clause.
$\{[$*Neopravuje* *se*$]_0\}_1$ *a* $\{[$*neinvestuje*$]_0\}_2$, $\{[$*peníze stačí jen na běžný provoz*$]_0\}_1$. 'They do not renovate nor invest, there is enough money only for routine operation.' (The underlined reflexive particle belongs to both verbs *opravovat* 'to renovate' and *investovat* 'to invest' (reflexive passive forms of the verbs); in the segmentation chart, it is marked as a part of the first clause *Neopravuje se* and elided in the second clause *neinvestuje*.)

On the other hand, a basic rule was adopted saying that a single finite verb form indicates a single clause, i.e., verb constitutes (a core of) a sentence[5] (providing that other formal labels as, e.g., brackets do not indicate more levels). This rule implies that if the shared sentence member is a predicate, then the particular segments are joined together into a single clause, as in the following example.
$\{[$*Petr přišel včera*$]_0$ *a* $[$*babička dneska*$]_0\}_1$. 'Petr came yesterday and my grandma today.'

Other constructions

Apposition is a construction where the same 'idea' is rendered in different ways (the latter being an explanatory equivalent of the former), both having the same syntactic relation to other sentence members (e.g., a name and a function of particular person, as in the following sentence).
$\{[$*Oznámil to Václav Havel*$]_0$, $[$*president České republiky*$]_0\}_1$. 'It was announced by Václav Havel, president of the Czech Republic.'

Following PDT, apposition is treated in the same way as coordination as the members of an apposition are considered to share (multiple) syntactic position in a sentence (like in the case of coordination).

Contrary to PDT, **parenthesis** without explicit/unambiguous formal mark, as e.g. brackets, is annotated as segment(s) on the same level as its/their neighboring segments.
$\{[$*Před smrtí*$]_0$, $\{[$*neznámo proč*$]_0\}_1$, $[$*si koupil tramvajenku*$]_0\}_2$. 'Before dying, nobody knows why, he bought a tram pass.'

Again, parenthetical expressions are captured as separate clauses even if they consist of fragmental expression.

Semi-direct speech, i.e., direct speech without quotation marks (or other formal label(s)) is annotated as segment(s) on the same level as the segment containing a governing verb. The reason is quite evident – there is no formally expressed indication of subordination in the segment(s) creating a semi-direct speech.
$\{[$*Přijde později*$]_0\}_1$, $\{[$*ohlásil doma Pavel*$]_0\}_2$. 'I will be late, said Pavel.'

4 Data Format and Editor for Segment Annotation

4.1 PML Data Format

The *Prague Markup Language*[6] (PML), see Pajas and Štěpánek (2006) is an XML-based domain language which has been developed and is used as primary data format for PDT (version 2.0).

The PDT 2.0 data consist of one non-annotated word layer (w-layer) and three layers of annotation: morphological (m-layer), analytical (a-layer) and tectogrammatical (t-layer). In PML, individual layers of annotation can be stacked one over another in a stand-off fashion and linked together as well as with other data resources in a consistent way.

We use two layers in our annotation editor, namely the m-layer and the a-layer. The m-layer provides the word form, lemma and tag for every token. The a-layer represents syntactic relations between tokens, resulting in an analytical tree. For the segment annotation, only information on analytical functions of tokens is used – it helps the annotators in their decisions on the appropriate level of embedding and in disambiguation if more readings of a particular sentence are possible.

[5] The account for this decision lies in the verb-centric character of dependency syntax traditionally used for Czech.

[6] `http://ufal.mff.cuni.cz/pdt2.0/doc/pdt-guide/en/html/ch03.html#a-data-formats`

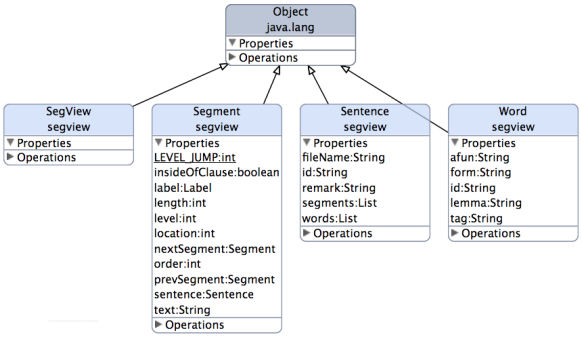

Figure 2: Class hierarchy of SegView annotation editor.

The output of the segment annotation is stored as a new layer of annotation, the seg-layer.

4.2 SegView Annotation Editor

The SegView annotation editor is implemented completely in Java because of its cross-platformity and availability of rich libraries. The presentation layer is implemented in the class *MainWindow* using the standard Swing library. As for the data layer, the editor works with files in the PML format (see Section 4.1). The model representing the core of the implementation comprises three classes: Sentence, Word and Segment, Figure 2.

After launching the editor, the user has the possibility to select multiple files to annotate. After the selection, the program directly reads the files and creates an internal representation with the instances of the three aforementioned classes. The manual annotation is saved in files with the extension `.seg`.

The screenshot of SegView editor is shown in Figure 3.

5 Basic Statistics and Conclusion

We have described the pilot phase of the segment annotation, during which 2,699 sentences from PDT were annotated with respect to their sentence structure.[7] Table 1 summarizes the amount of annotated data and gives statistics on number of processed segments and clauses.

The most frequent annotation patterns are presented in Table 2 showing the most common types of sentences and relation among their clauses (only patterns with more than 100 sentence instances are listed).

[7]We have focused on the sentences from data/full/amw/train2 portion of the PDT data.

# sentences	2,699
# segments	7,975
# clauses	5,003
max segments in clause	27
max clauses in sentence	11
max levels of embedding	4

Table 1: Basic statistics of the annotated texts.

sentences	segments	clauses	max level
783	1	1	0
298	2	1	0
195	2	2	1
148	3	2	1
123	3	1	0
111	2	2	0

Table 2: Distribution of segments and clauses.

The most frequent type of annotated sentence consists of one segment only (and thus one clause), then comes the case where two segments form a single clause. The third position is for sentences with two segments, each forming an individual clause, where one of them depends on the other). The fourth case represents sentences formed by two clauses, one either depending on the other or forming a parenthesis. The fifth and sixth line represent sentences with segments on the same level, e.i., with sentence members in coordination or apposition and with coordinated clauses, respectively. (The most common cases listed in the table represent 61.5% of the annotated sentences; the rest has more complicated structures.)

Future work

We focus on the inter-annotator agreement on a reasonable large set of data now to check the consistency between the human annotators. Then the annotation will continue – the goal is to cover 10% of sentences from PDT with assigned sentence structure.

We expect the use of the manually annotated data for testing tools and hypotheses on possible sentence structures. The proposed amount of data is comparable with the standard PDT testing data. We do not foreseen the use of this set of segmentation charts for training statistically-based tool(s) for an automatic identification of sentence structures.

The set of precisely annotated data allows us to solidly compare and evaluate the already existing automatic segmentation tools processing either the raw texts or syntactically annotated trees, see Krůza and Kuboň (2009) and Lopatková and

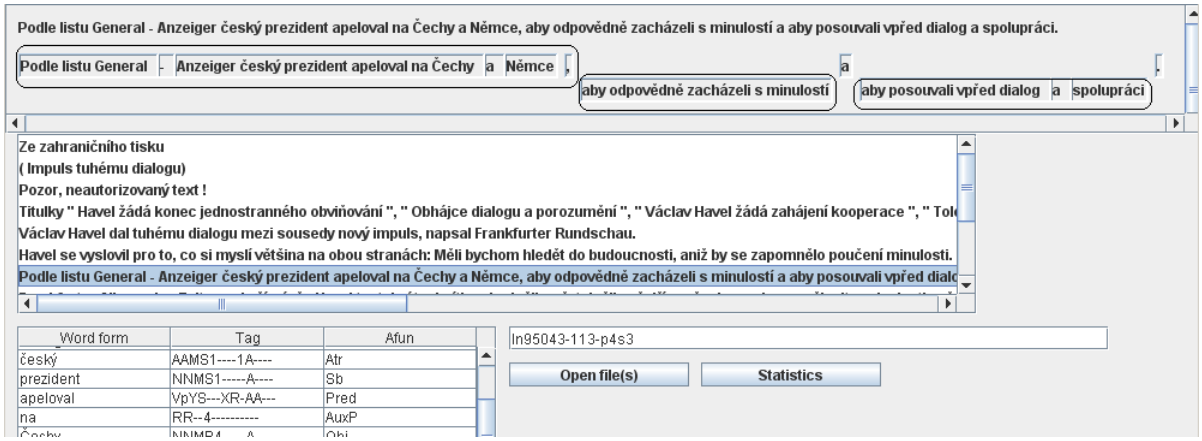

Figure 3: SegView editor: The segmentation chart for sentence 'According to the General-Anzeiger, Czech president appealed to Czechs and Germans that they should treat their history responsibly and improve their mutual dialogue and cooperation.' (clauses marked by ellipses).

Holan (2009). These data also allow us to search for systemic differences between the manual and automatic sentence structure annotation. Then the possibility of further improving the tools will be opened.

The use of data with automatically annotated sentence structure in machine translation system among related languages, as in Homola and Kuboň (2008), is also foreseen.

Acknowledgements

This paper presents the results of the grant of the Grant Agency of Czech Republic No. 405/08/0681. The research is carried out within the project of the Ministry of Education, Youth and Sports of Czech Republic No. MSM0021620838.

References

Steven P. Abney. 1991. Parsing By Chunks. In R. Berwick, S. Abney, and C. Tenny, editors, *Principle-Based Parsing*, pages 257–278. Kluwer Academic Publishers.

Steven P. Abney. 1995. Partial Parsing via Finite-State Cascades. *Journal of Natural Language Engineering*, 2(4):337–344.

Fabio Ciravegna and Alberto Lavelli. 1999. Full Text Parsing using Cascades of Rules: An Information Extraction Procedure. In *Proceedings of EACL'99*, pages 102–109. University of Bergen.

Jan Hajič, Eva Hajičová, Jarmila Panevová, Petr Sgall, Petr Pajas, Jan Štěpánek, Jiří Havelka, and Marie Mikulová. 2006. *Prague Dependency Treebank 2.0*. LDC.

Jan Hajič. 2004. *Disambiguation of Rich Inflection (Computational Morphology of Czech)*. Karolinum Press.

Petr Homola and Vladislav Kuboň. 2008. A hybrid machine translation system for typologically related languages. In David Wilson and Chad Lane, editors, *Proceedings of FLAIRS 2008)*, pages 227–228, Coconut Grove, Florida, USA. AAAI Press.

Bernard E. M. Jones. 1994. Exploiting the Role of Punctuation in Parsing Natural Text. In *Proceedings of the COLING'94*, pages 421–425.

Oldřich Krůza and Vladislav Kuboň. 2009. Automatic Extraction of Clause Relationships from a Treebank. In *Computational Linguistics and Intelligent Text Processing - Proceedings of CICLing 2009*, volume 5449 of *LNCS*, pages 195–206. Springer-Verlag.

Vladislav Kuboň, Markéta Lopatková, Martin Plátek, and Patrice Pognan. 2007. A Linguistically-Based Segmentation of Complex Sentences. In D.C. Wilson and G.C.J. Sutcliffe, editors, *Proceedings of FLAIRS Conference*, pages 368–374. AAAI Press.

Vladislav Kuboň. 2001. *Problems of Robust Parsing of Czech*. Ph.D. thesis, Faculty of Mathematics and Physics, Charles University in Prague.

Markéta Lopatková and Tomáš Holan. 2009. Segmentation Charts for Czech – Relations among Segments in Complex Sentences. In A. H. Dediu, A. M. Ionescu, and C. Martín-Vide, editors, *Proceedings of LATA 2009*, volume 5457 of *LNCS*, pages 542–553. Springer-Verlag.

Tomohiro Ohno, Shigeki Matsubara, Hideki Kashioka, Takehiko Maruyama, and Yasuyoshi Inagaki. 2006. Dependency Parsing of Japanese Spoken Monologue Based on Clause Boundaries. In *Proceedings of COLING and ACL*, pages 169–176. ACL.

Petr Pajas and Jan Štěpánek. 2006. XML-Based Representation of Multi-Layered Annotation in the PDT 2.0. In *Proceedings of LREC 2006 Workshop on Merging and Layering Linguistic Information*, pages 40–47. ELRA.

Daniel Zeman. 2004. *Parsing with a Statistical Dependency Model*. Ph.D. thesis, Charles University in Prague, Prague.