Extracting Phrases from PDT 2.0

Vašek Němčík

NLP Centre
Faculty of Informatics, Masaryk University
Brno, Czech Republic
xnemcik@fi.muni.cz

Abstract. The Prague Dependency Treebank (henceforth PDT) is a large collection of texts in Czech. It is renown for its respectable size and rich multi-layer annotation covering a wide range of complex phenomena. One the other hand, it can be argued that the complexity of the dataset may be a notable hindrance to using certain aspects of the data in a straightforward way. To overcome these problems, we present an export filter converting PDT into a more transparent data format, containing information about the most common phrase types. We believe that availability of the PDT data in this form will help encourage people unfamiliar with the underlying theory to use the corpus.

Key words: PDT, corpus, treebank, export, format, complex annotation, phrase, clause

1 Introduction

The Prague Dependency Treebank 2.0 (henceforth just PDT) is a large collection of Czech texts compiled at the Institute of Formal and Applied Linguistics at the Charles University in Prague. It was created within an open-ended project for manual annotation of substantial amount of Czech-language data with linguistically rich information ranging from morphology through syntax and semantics/pragmatics and beyond. [1]

PDT is a notable linguistic resource and is renown for its fair size and the underlying sophisticated linguistic theory, the FGD (Functional Generative Description), with a respectable tradition in the field of general linguistics. The corpus is organized in three layers, two of them tree-based, and covers a wide range of linguistic phenomena. The annotation principles are derived mainly from the concept of syntactic dependence.

The concept of syntactic dependence offers a straightforward theoretical base for building tree structures. However, in order to create fully connected syntactic trees for all sentences, it is necessary to spoil its theoretical purity by adding exceptions necessary to handle the irregularities and obscurities of everyday language. In such a situation, it is necessary to reach a certain trade-off between theoretical purity on one hand, and clarity, transparence and ease of use on the other.

Aleš Horák, Pavel Rychlý (Eds.): Proceedings of Recent Advances in Slavonic Natural Language Processing, RASLAN 2011, pp. 51–57, 2011.

In my opinion, the PDT is strongly inclined towards theoretical purity. The scope of the attributes and phenomena encompassed by the PDT is very broad, from part-of-speech and morphological tags to complex verb groups, coordinations, reconstructions of elliptical phrases, and semantic features not directly present, but inferrable from the data. In order to annotate the corpus correctly, the human annotators were provided with rather bulky annotation guidelines,[1] describing many important structural differences based on making minute distinctions. The resulting representation is theoretically sound, but for a person not thoroughly familiar with the underlying theory rather difficult to grasp. This is striking especially in sentences exhibiting a combination of several complex linguistic phenomena. Abundance of such sentences in the corpus lead us to create an export tool projecting the data to a more straightforward data format. We hope it helps computational linguists to use the data in practically oriented systems in a more efficient way.

Next section gives an overview of selected annotation principles that may cause confusion with new users of the corpus. Further, in Section 3, we present the output format of the proposed *pdt2vert* export tool and sketch on the conversion technicalities. Finally, we summarize the paper and comment on our future work.

2 Complex Annotation Drawbacks

This section gives an overiew of selected features of the PDT data, which are rather irregular and may be a source of misunderstandings or confusion.

The core of the annotation, the analytical and tectogrammatical layer, is based on the concept of dependence. This concept has a long tradition, and the underlying relation can be straightforwardly defined and determined,[2] and exhibits theoretically appealing properties. To cover commonly occurring phenomena, the dependency concept was supplemented by the following constructions:

Coordination introduces a different type of edges into the PDT trees. These edges connect the coordination root (eg. a conjunction or a comma) to the coordinated elements. The dependency relation can be virtually reconstructed as "skipping" the coordination root – ie. the parent node of the coordination root having the coordination members as its dependent nodes. Coordinations are often recursive, which makes this reconstruction of node dependencies nontrivial.

Ellipsis concerns words that have been omitted from the sentence. It is possible to think of them as of nodes "skipped" when creating a branch of

[1] The annotation manual for analytical layer [2] has 301 pages, for tectogrammatical layer [3] 1215 pages.

[2] The syntactic dependence between elements A and B, the element A being dependent on the element B, specifies that B can occur without A, but A cannot occur without B. [4]

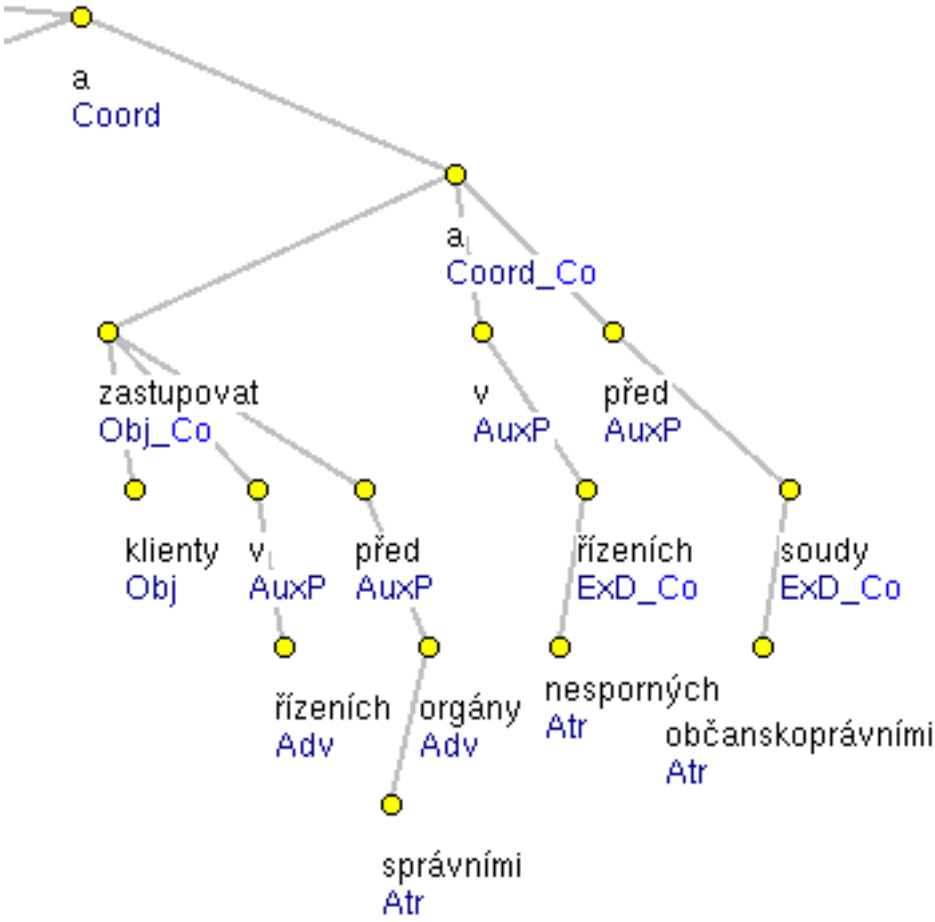

Fig. 1. A fragment of an analytical tree.

the dependency tree. Their descendants are placed below its parent with the analytical functor `ExD` (which unfortunately veils their real functors).

Shared attributes of coordinated nodes are represented by nodes directly dependent on the coordination root, but not marked as coordination members. In my opinion, this convention has the most far-reaching consequences. Most notably, phrases are not necessarily subtrees of the sentence tree, and after reconstructing the dependency relation from this notation, it does not form a tree (the root of such a shared subtree has several parents). When working with the data, this can come rather unexpected and counter-intuitive.

Conjunctions and prepositions are represented by nodes with analytical functors `AuxC` and AuxP respectively. These words are syntactically governors, but their function is mostly auxiliary. As a result, they are ignored (skipped) in many annotation rules.

The tectogrammatical layer is very complex and its annotation involves making many decisions. These are often based on rather fine distinctions and have various consequences – not only different tectogrammatical functors (for example `&Gen` vs. `&Unsp`), but also structural differences (such as ellipsis vs. shared attributes). In my opinion, with regard to the abundance of structural exceptions and irregularities, and the number of node attributes and values, this annotation level may be insightful for a human linguist, but is rather unsuitable for automatic processing.

Especially combinations of these phenomena may lead to a certain confusion based on false assumptions about the data structure. This can be illustrated by the example in Figure 1. It contains a part of an analytical tree, containing a multiple coordination (coordinating an infinite verb form and two prepositional phrases), and ellipsis, where the nodes signalling the coordination membership and ellipsis (with its `ExD_Co` functor) are below preposition nodes.

In order to make the PDT data more transparent, and minimize the interference of annotation of different phenomena, we have proposed an alternative export data format. We hope it will encourage computational linguists who would otherwise be discouraged by the complexity of the data, to use it in their applications.

The proposed export format is described in the following section.

3 The PDT2vert format

The *pdt2vert* is proposed to offer an alternative, more straightforward way of presenting the PDT data. We also believe this format is more convenient for automatic processing within typical NLP applications.

The output structure is linear, and is based on the so-called vertical format. The main principle of the vertical format is that each line contains either information about one token, or a structural tag. A token line typically contains the word surface form, lemma, morphological tag, etc. Structure tags are used to express the global data structure, mainly boundaries of sentences, paragraphs, and documents.

Table 1. Example of a sentence in the *pdt2vert* format.

```
<sentence id="ln94202-55-p5s2">
<clause id="t-ln94202-55-p5s2w3">
<markable id="ln94202-55-p5s2w1" type="np" mtag="NNFS7---A--">
Podmínkou podmínka             NNFS7---A--   Pnom
</markable>
však      však                 J^-------     AuxY
je        být                  VB-S--3P-AA-- Pred
</clause>
,         ,                    Z:-------     AuxX
<clause id="t-ln94202-55-p5s2w9">
aby       aby                  J,-------     AuxC
<markable id="ln94202-55-p5s2w8" type="np" mtag="NNFP1---A--">
tyto      tento                PDFP1-----    Atr
činnosti  činnost_^(*3ý)       NNFP1---A--   Sb
</markable>
s         s-1                  RR-7-----     AuxP
<markable id="ln94202-55-p5s2w12" type="np" mtag="NNIS7---A--">
právním   právní               AAIS7--1A--   Atr
úkonem    úkon                 NNIS7---A--   Obj
obsaženým obsažený_^(*5áhnout) AAIS7--1A--   Atr
v         v-1                  RR-6-----     AuxP
<markable id="ln94202-55-p5s2w16" type="np" mtag="NNIS6---A--">
notářském notářský             AAIS6--1A--   Atr
zápise    zápis                NNIS6---A--   Adv
</markable>
</markable>
nebo      nebo                 J^-------     Coord
s         s-1                  RR-7-----     AuxP
<markable id="ln94202-55-p5s2w19" type="np" mtag="NNFS7---A--">
přípravou příprava             NNFS7---A--   Obj
<markable id="ln94202-55-p5s2w21" type="np" mtag="NNIS2---A--">
tohoto    tento                PDZS2-----    Atr
úkonu     úkon                 NNIS2---A--   Atr
</markable>
</markable>
</clause>
.         .                    Z:-------     AuxK
</sentence>
```

As demonstrated by Table 1, the *pdt2vert* format extends the scope of structures accounted for. It covers most notably noun phrases and clauses, optionally, singleton tags are inserted to represent zero subjects. The token attributes can contain the analytical functor (last column in Table 1), the functor of the corresponding tectogrammatical node (when applicable), the token identifier, or the identifier of the parent node. The choice and order of attributes can be customized.

The conversion of the data is not straightforward, the most difficult part being clause boundary detection, necessary for instance for detecting zero subjects and pruning noun phrases. The convertor is written in Perl, using the BTrEd interface supplied with the corpus.

4 Conclusions and Further Work

We have presented an export filter for converting PDT into a more transparent data format, which is perhaps more appealing to users not interested in the advanced features of the corpus.

As a next step, the PDT data in this format will be used for various evaluation tasks. As mentioned in [5] and [6], PDT in its original form has a certain bias when used to evaluate parsers. The PDT dependency trees are too detailed (eg. accounting for punctuation and non-word tokens) and a more compact representation would be needed to serve as a plausible evaluation standard. Further, this format will be used to evaluate algorithms for anaphora resolution and topic-focus articulation.

The generality of the output format makes it possible to create files that can be conveniently used to train or evaluate a wide range of linguistic models.

Acknowledgments This work has been partly supported by the Ministry of Education of CR within the Center of basic research LC536 by the Czech Science Foundation under the project 407/07/0679.

References

1. Hajič, J., et al.: The Prague Dependency Treebank 2.0. Developed at the Institute of Formal and Applied Linguistics, Charles University in Prague. (2005) `http://ufal.mff.cuni.cz/pdt2.0/`.
2. Hajič, J., Panevová, J., Buráňová, E., Urešová, Z., Bémová, A.: Anotace Pražského závislostního korpusu na analytické rovině: pokyny pro anotátory. Technical Report 28, ÚFAL MFF UK, Prague, Czech Republic (1999)
3. Mikulová, M., Bémová, A., Hajič, J., Hajičová, E., Havelka, J., Kolářová-Řezníčková, V., Kučová, L., Lopatková, M., Pajas, P., Panevová, J., Razímová, M., Sgall, P., Štěpánek, J., Urešová, Z., Veselá, K., Žabokrtský, Z.: Anotace Pražského závislostního korpusu na tektogramatické rovině: pokyny pro anotátory. Technical report, ÚFAL MFF UK, Prague, Czech Republic (2005)

4. Bußmann, H.: Lexikon der Sprachwissenschaft. Alfred Kröner Verlag, Stuttgart (2002)
5. Horák, A., Holan, T., Kadlec, V., Kovář, V.: Dependency and phrasal parsers of the czech language: A comparison. In: Proceedings of 10th International Conference on Text, Speech, and Dialogue (TSD 2007), Berlin, Heidelberg, Springer (2007) 76–84
6. Kovář, V., Jakubíček, M.: Prague dependency treebank annotation errors: A preliminary analysis. In: RASLAN 2009 : Recent Advances in Slavonic Natural Language Processing, Brno, Masaryk University (2009) 101–108