Chapter 1

THE PRAGUE DEPENDENCY TREEBANK: A THREE-LEVEL ANNOTATION SCENARIO

Alena Böhmová, Jan Hajič, Eva Hajičová, Barbora Hladká
Institute of Formal and Applied Linguistics
Faculty of Mathematics and Physics
Charles University
Malostranské nám. 25
CZ—118 00 Prague 1
{bohmova,hajic,hajicova,hladka}@ufal.mff.cuni.cz

Abstract The availability of annotated data (with as rich and "deep" annotation as possible) is desirable in any new developments. Textual data are being used for so-called training phase of various empirical methods solving various problems in the field of computational linguistics. While there are many methods that use texts in their plain (or raw) form (in most cases for so-called unsupervised training), more accurate results may be obtained if annotated corpora are available. The data annotation itself is a complex task. While morphologically annotated corpora (pioneered by Henry Kučera in the 60's) are now available for English and other languages, syntactically annotated corpora are rare. Inspired by the Penn Treebank, the most widely used syntactically annotated corpus of English, we decided to develop a similarly sized corpus of Czech with a rich annotation scheme.

Keywords: corpora, treebanks, annotation schema, morphology, syntax, tectogrammatical tree structures, Czech language

1. THE PRAGUE DEPENDENCY TREEBANK

The Prague Dependency Treebank (PDT) has a three-level structure. Full *morphological* annotation is done on the lowest level ([5], [6]). The middle level deals with superficial (surface) syntactic annotation ([1]) using dependency syntax; it is called the *analytical* level, and it is conceptually close to the level of syntactic annotation used in the Penn Treebank ([16]). The highest level of annotation is the *tectogrammatical* level, or the level of linguistic meaning (based on the framework of Functional Generative Description, [21]).

The textual data used for the PDT task contain general newspaper articles (60%; including but not limited to politics, sports, culture, hobbies, etc.), economic news and analyses (20%), popular science magazines (20%), all selected from the Czech National Corpus (CNC, [4]). We annotate the same texts on all three levels, but the amount of annotated material decreases with the complexity of the levels[1], from about 1.8 mil. tokens on the morphological level to about 1 mil. tokens on the tectogrammatical level. SGML markup is used throughout as the data interchange format[2].

The PDT is a long-term project (1996-2000, now extended to 2004) with two major phases. In the first phase, now completed, the first two levels of annotation have been completed and made available; also the specification of the tectogrammatical level has been finished[3]. During the second phase, the tectogrammatical annotation will be completed and released.

2. MORPHOLOGICAL LEVEL

The morphological analysis of an isolated[4] word form produces a *lemma*[5] (or more in the case of a morphological ambiguity) and a combination of values of individual morphological categories. The combination of those values is called a morphological tag (*MTag*); in other words, the list of possible MTags together with corresponding lemmas represents the output of the morphological analysis of the input word form. In a given context, just one pair (MTag, lemma) "fits in"; the context-sensitive process of selecting the fitting pair is called morphological annotation (if it is done manually) or morphological tagging (if it is automatically). In order to use the tags effectively in applications, and for uniformity, we also follow the usual practice and assign "lemmas" and appropriate "morphological" tags to punctuation.

[1]The decrease in the volume of the annotated material is dictated mostly by the technical considerations related to intended applications and current evaluation metrics.
[2]Some annotation and processing tools do use a different format internally.
[3]To the extent possible without the feedback of large scale annotation.
[4]I.e., regardless of context.
[5]A lemma is an identifier of the underlying lexical unit, and it is usually represented by a word (string of characters) corresponding to a usual dictionary headword for readability (possibly complemented by a distinguishing number for homonymous or polysemous words).

Thus on the morphological level of the PDT, a MTag and a lemma are assigned to each token in the input data. The morphological annotation of the PDT has been done semi-automatically. It is a two-step process: first, the input text (Fig. 1.1) is processed automatically by the morphological analyzer ([6]), resulting in a list of possible (lemma, MTag) pairs for each input token (Fig 1.2, SGML markup `<MMl>, <MMt>`). Then, manual disambiguation yields the desired unique pair (Fig 1.3, SGML markup `<l>, <t>`). Currently, Czech MTags are defined as a concatenation of 15 morphological categories and each morphological category corresponds to precisely one position. For instance, the part of speech "sits" in the 1st position, gender in the 3rd, case in the 5th, etc. For example, the MTag `NNFS6-----A----` represents a singular (`S`) feminine (`F`) general (`N`) noun (`N`) in the locative case (`6`), without (negative) prefix (`A`)[6]. A detailed description of the positional MTag system is presented also in [6].

`<s id="s/inf/j/1994/cmpr9410:001-p24s3">`	unique sentence ID
`<f cap>Šance`	word form token
`<f>je`	word form token
`<f>přesto`	word form token
`<f>minimální`	word form token
`<d>.`	punctuation token

Figure 1.1 The SGML format of CNC illustrated by the Czech sentence Šance je přesto minimální [lit. The chance is nevertheless minimal].

A morphologically annotated corpus can then be used to train a tagger based on a probabilistic model (or another type of automatic tagger) which in turn can be used to automatically annotate large amounts of previously unseen (new) texts. Similar SGML markup is used there (Fig. 1.4, SGML markup `<MDl>`, `<MDt>`).

2.1 ANNOTATION

The morphological level has been annotated by a separate team of annotators. The group (seven undergraduate students with either a computer science or linguistics background) proceeded in two separate phases. During the first phase - for each text to be annotated - two annotators independently chose the (lemma, MTag) pair from the list suggested by the morphological analyzer. The two versions of the same text were compared to each other, and then in the second phase another

[6]All other categories, such as person or tense, are irrelevant and denoted by a hyphen (`-`).

```
<s id="s/inf/j/1994/cmpr9410:001-p24s3">
<f cap>Šance<MMl>šance<MMt>NNFP1-----A----<MMt>NNFP4-----A----
   <MMt>NNFP5-----A----<MMt>NNFS1-----A----<MMt>NNFS2-----A----
   <MMt>NNFS5-----A----
<f>je<MMl>být<MMt>VB-S---3P-AA---<MMl>on<MMt>PPNS4--3-------
   <MMt>PPXP4--3-------
<f>přesto<MMl>přesto<MMt>Dg-------1A----
<f>minimální<MMl>minimální<MMt>AAFP1----1A----
   <MMt>AAFP4----1A----<MMt>AAFP5----1A----<MMt>AAFS1----1A----
   <MMt>AAFS2----1A----<MMt>AAFS3----1A----<MMt>AAFS4----1A----
   <MMt>AAFS5----1A----<MMt>AAFS6----1A----<MMt>AAFS7----1A----
   <MMt>AAIP1----1A----<MMt>AAIP4----1A----<MMt>AAIP5----1A----
   <MMt>AAIS1----1A----<MMt>AAIS4----1A----<MMt>AAIS5----1A----
   <MMt>AAMP1----1A----<MMt>AAMP4----1A----<MMt>AAMP5----1A----
   <MMt>AAMS1----1A----<MMt>AAMS5----1A----<MMt>AANP1----1A----
   <MMt>AANP4----1A----<MMt>AANP5----1A----<MMt>AANS1----1A----
   <MMt>AANS4----1A----<MMt>AANS5----1A----
<d>.<MMl>.<MMt>Z:-------------
```

Figure 1.2 Input sentence (from Fig. 1.1) after automatic morphological analysis

```
<s id="s/inf/j/1994/cmpr9410:001-p24s3">
<f cap>Šance<l>šance<t>NNFS1-----A----
<f>je<l>být<t>VB-S---3P-AA---
<f>přesto<l>přesto<t>Dg-------1A----
<f>minimální<l>minimální<t>AAFS1----1A----
<d>.<l>.<t>Z:-------------
```

Figure 1.3 Annotated input sentence (from Fig. 1.2), after manual disambiguation

```
<s id="s/inf/j/1994/cmpr9410:001-p24s3">
<f cap>Šance<MDl>šance<MDt>NNFS1-----A----
<f>je<MDl>být<MDt>VB-S---3P-AA---
<f>přesto<MDl>přesto<MDt>Dg-------1A----
<f>minimální<MDl>minimální<MDt>AANS1----1A----
<d>.<MDl>.<MDt>Z:-------------
```

Figure 1.4 Tagged input sentence (from Fig. 1.1), with “automatic” tags

annotator resolved the differences between them. Six of the seven students were the “first phase” annotators and only one was the “second phase” were annotator-arbiter, with the hope of consistent tag assignment throughout the corpus.

In order to make the annotation of texts more human-friendly (and less error-prone), a special purpose tool has been developed. The tool was first implemented under Linux and then reimplemented for the MS Windows platform.

The input text contained about 56% ambiguous tokens at the time of the annotation[7]. In line with the observations of others (most notably, [15]), we have found that about 4.9% of the input tokens were annotated differently by the two "first phase" annotators. Almost all the differences are caused by performance errors.

3. ANALYTICAL LEVEL

The analytical (syntactic) level of annotation([1]) is the second (middle) level of the overall annotation scheme. We have chosen the dependency structure to represent the (surface) syntactic relations within a sentence; no "phrase labels" are used. The dependency structure is based on a dependency relation (often referred to as the relation of determination, or mother/daughter, or head/modifier relation) between a governor and its dependent node (labeled by words) in a dependency tree.

The basic design principles of the analytical level (or ATS, for Analytical Tree Structures) are:

(1) each word and each punctuation mark is represented by exactly one node,

(2) no nodes are added (with the exception of a special "technical" auxiliary root node of the tree),

(3) non-projectivity (i.e. crossing of edges) is allowed,

(4) the result is a dependency tree, in which the edges (links) are explicitly labeled analytical (syntactic) tags (*STag*s),

(5) each node of the resulting analytical tree consists of three parts:

  (a) the original word form,

  (b) the morphological tag and lemma (which come from the morphological level, unchanged),

  (c) the syntactic tag (STag as a label of the dependency link).

All possible values (STags) of the analytical function attribute (*afun*) are described in Table 1.A.1 in Appendix.

[7]The ambiguity level is largely determined by the morphological analyzer. As new words are being added to its dictionary, more ambiguity is introduced.

The annotation rules ([1]) follow the traditional grammar books whenever possible, but are both extended (where no guidance has been found in such books) and modified (where the current grammars are inconsistent, for example).

Fig. 1.5 gives an example of the analytical-level annotation[8] of the sentence

(1) Do 15. května budou cestující platit dosud platným způsobem.
Until May 15, passengers will pay using the current scheme.

The original word forms as well as the STags assigned to the analytical function node attributs are displayed. This example illustrates:

- the extra root node of the tree, with the number of the sentence within the file;
- the handling of an analytical verb form (`AuxV` *budou* + infinitive *platit*);
- the fact that the verb is the governing node of the whole sentence (or of every clause in compound sentences), as opposed to the complex subject–complex predicate distinction made even in the otherwise dependency-oriented traditional grammars of Czech, such as ([22]);
- an attachment of a manner-type adverbial to an analytical verb form;
- the handling of a date;
- prepositional phrase structure (preposition as the governor);

and, of course, all the analytical functions at these nodes.

Figure 1.6 illustrates the SGML format of the PDT with both the morphological and analytical levels marked. The morphological level markup `<l>, <t>` is described above in Sect. 2.; the analytical level markup `<A>, <r>` and `<g>` denotes the Stag (analytical function), the token numerical ID (based on the surface word order within the sentence) and the numerical ID of its governing token, respectively.

The annotation process was viewed as (an interative) process where the rules for annotation were constructed based on the evidence found in

[8]Please note that for technical reasons, analytical functions are part of the dependent node label, even though they refer to the dependency link as a whole.

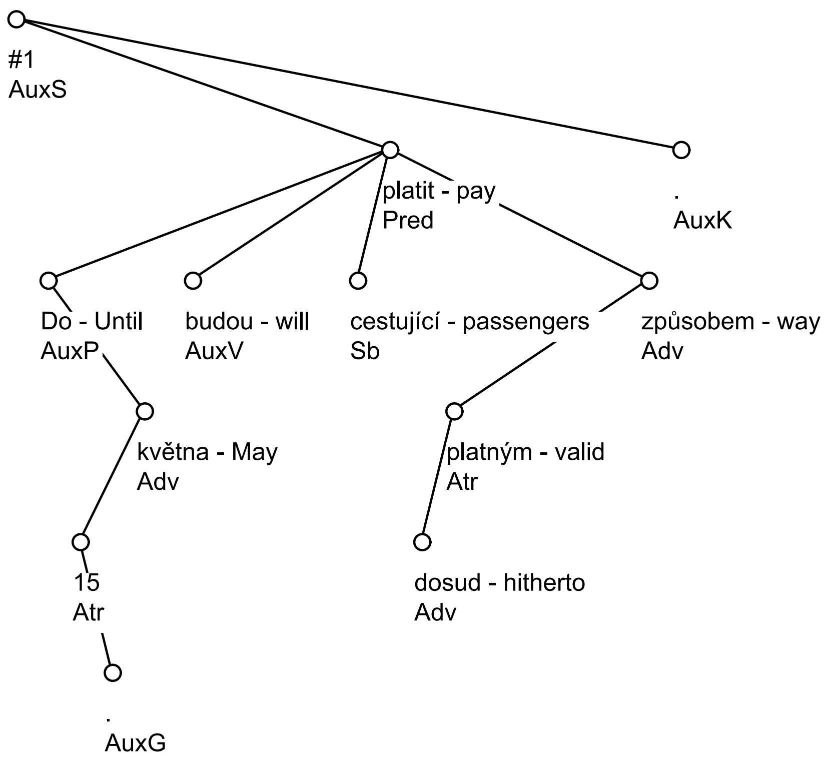

Figure 1.5 Analytical level annotation of sentence (1)

```
<f cap>Do<l>do<t>RR--2----------<A>AuxP<r>1<g>7
<f num>15<l>15<t>C=-------------<A>Atr <r>2<g>4
<d>.<l>.<t>Z:-------------<A>AuxG<r>3<g>2
<f>května<l>květen<t>NNIS2-----A----<A>Adv<r>4<g>1
<f>budou<l>být<t>VB-P---3F-AA---<A>AuxV<r>5<g>7
<f>cestující<l>cestující<t>NNMP1-----A----<A>Sb <r>6<g>7
<f>platit<l>platit<t>Vf--------A----<A>Pred<r>7<g>0
<f>dosud<l>dosud<t>Db-------------<A>Adv<r>8<g>9
<f>platným<l>platný<t>AAIS7----1A----<A>Atr<r>9<g>10
<f>způsobem<l>způsob<t>NNIS7-----A----<A>Adv<r>10<g>7
<d>.<l>.<t>Z:-------------<A>AuxK<r>11<g>0
```

Figure 1.6 The SGML format of the morphologically and analytically annotated input sentence the data. Thus before the manual annotation began[9], we explained the basic principles of annotation to the annotators, and asked them to use existing grammar books, most notably [22], an old, but still the most suitable and authoritative description of Czech syntax. It is based on a dependency framework, although there are some (easily identifiable and replaceable) deviations. We were aware of the fact that there are many gaps in such a traditional grammar from the point of view of an explicit annotation: mainly, the requirement to have each input word represented by a node in the tree (a demand quite natural from the computational point of view) is largely not reflected in any human-oriented grammar description.

3.1 TREATMENT OF SPECIAL PHENOMENA

We believe that the dependency structure we use is a very good formal representation for the central notion of a dependency theory, i.e. the determination relation. However, it is not simple to represent syntactic relations of another type (non-determinative, such as coordination and apposition) in a plain two-dimensional graph; therefore we have introduced some special conventions for that purpose. Fig. 1.7 illustrates a coordination of two members of a sentence (attributes in this case):

(2) V roce 1994 dochází k mírnému oživení světové ekonomiky a světového obchodu.

A weak rebound of the world's economy and world trade occurred in 1994.

The analytical function at the coordinated nodes now contains an Stag bearing a suffix `_Co`, which denotes coordinated nodes while still keeping the information about the true original Stag. As an additional rule, we have stated that all coordinated nodes must contain the same Stag (with the exception of elliptical constructions).

When a syntactically ambiguous sentence is being analyzed and the annotator is not able to decide on the preferred type of dependency (either an adverbial dependency, where a verb is a head, or an adnominal dependency with a noun as the head), and *both readings are in fact*

[9]The manual annotation proper was started in November 1996 and finished in December 1999.

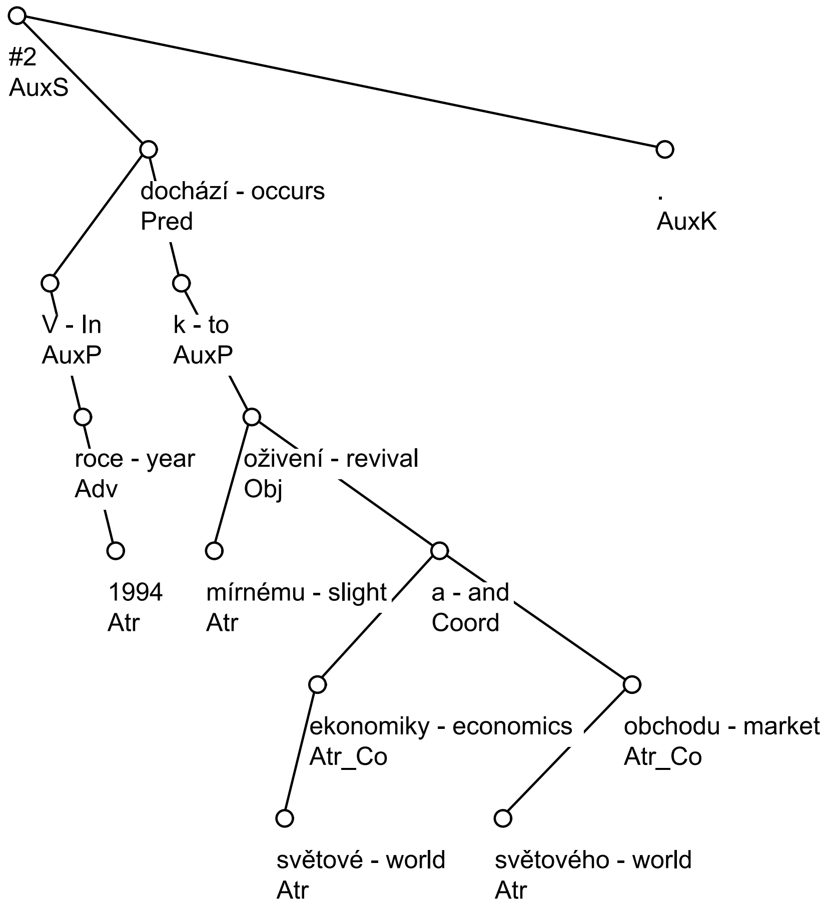

Figure 1.7 Example of coordination - sentence (2); lit. word translation used

plausible[10], "double" analytical functions (`ObjAtr, AdvAtr`) are used, see Examples (3) and (4):

(3) Moderní teorie o bytí říká, že bytí...

Modern theory of human beings says that human beings ...

(4) Napsal referát do vědeckého časopisu.

[10] As opposed to the case when simply there is not enough context to decide, or the annotator's "world knowledge" is insufficient to separate the two incompatible readings; in such a case, someone eventually has to decide just one of them.

He has written a paper for a scientific journal.

In (3) either the dependency pair *říká* 'says' (as a mother node) and *o bytí* 'of human being' (daughter node with `Obj`(ect) function), or the pair *teorie* 'theory' (mother) and *o bytí* 'of human being' (daughter with `Attr`(ibute) function) is present): here the "double" function `ObjAtr`[11] is applied. In Ex. (4) the same phenomenon occurs with the prepositional case *do časopisu* 'for a scientific journal', which will be labeled by the `AdvAtr` function.

The sentence

(5) Maďarsko a Slovensko očekává stagnaci HDP, Polsko pokračující růst.

Hungary and Slovakia expect the GDP to stay flat while Poland expects its continued growth.

and its analytical tree structure in Fig. 1.8 illustrate a situation when no token can be found in the sentence to be the governor of 'Poland' and 'growth' (i.e. the governor is deleted in the surface structure); in such a case we use the label `ExD` (extra dependency, denoting that the governor is actually missing) and point the dependency link towards a node that would normally govern the missing one. Even though in many sentences the missing node could easily be identified, we have decided not to add it at this level of annotation (thus sticking with the first and second basic principles of analytical level annotation) and deal with the problem properly at the third, tectogrammatical level.

A detailed description of the conventions used can be found in [1].

3.2 TWO MODES OF ANNOTATION

At the beginning of the annotation effort the annotators were constructing the syntactic structure and deciding on the analytical functions at the tree nodes purely manually with only the help of "user-friendly" software with a graphical interface ([14]; see also Sect. 5.5). The only information they had at this stage was the textual form of the sentence itself, and a morphological analysis of each token (not disambiguated, however).

Later, after macro programming capabilities had been added to the annotation software, a set of (manually written) rules was used to preassign the analytical functions (STags) to a completed tree. The annotators first had to manually build the tree structure, then - at a press

[11] The order of the simple STags within the "double" Stag is not important.

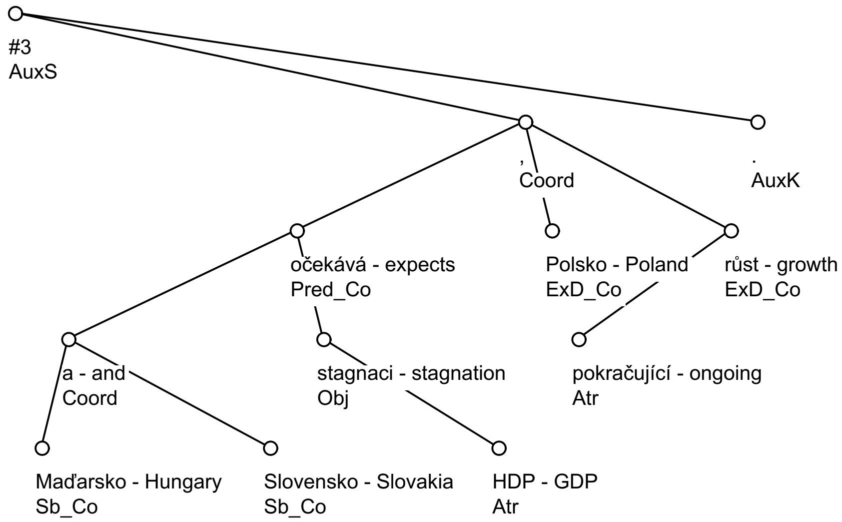

Figure 1.8 Example of ellipsis handling - sentence (5); lit. word translation used

of a button - the rules applied and populated the tree with possible STags achieving about 80% precision[12]. The annotators then reviewed, inserted, selected and/or corrected the resulting STags.

Once enough data had been annotated, even more preprocessing was introduced. Collins' lexicalized stochastic parser ([2]) was adapted to suit the data structure and format and trained on 19126 analytically annotated Czech sentences, assigning 80% of dependencies correctly. Our adaptation ([7], [3]) consists mainly of determining the phrase structure from the dependency structure[13] and the phrase labels from morphology[14], since Collins' parser works with labeled parse trees.

Using the parsing results in the later stages of the annotation effort, the annotators' task had changed: instead of building the tree structure from scratch, they were instructed to review the structure created by the parser, and correct it if necessary. The rest of the task remained unchanged; the analytical functions[15] were assigned by the manually written rules and checked by the annotators.

[12]About 10% of nodes have been left without any Stag, and about 5% of nodes got more than one Stag.
[13]Which is a non-trivial task since the dependency-to-bracketing mapping is one-to-many.
[14]This is also a non-trivial task given the rich inflectional morphology Czech enjoys, leading to data sparseness problems.
[15]Analytical functions are not used nor produced by the adapted Collins' parser.

Fig. 1.9 presents the analytic tree structure for the sentence

(6) Ve svobodných celních zónách dnes pracují po celém světě čtyři miliony lidí.

Today, four million people work off-shore all over the world.

which has been parsed correctly by the parser. At this sentence size, the parser's output is almost error-free. However, complicated, incomplete, coordinated, and/or partially ungrammatical sentences are unfortunately quite common in the treebank and are rarely parsed without an error. We have analyzed parsing results obtained on a randomly chosen set of 50 sentences, and found that 19 sentences were correctly assigned all the dependency links, 7 sentences contained one wrong dependency, and the rest required more than one intervention by the annotator.

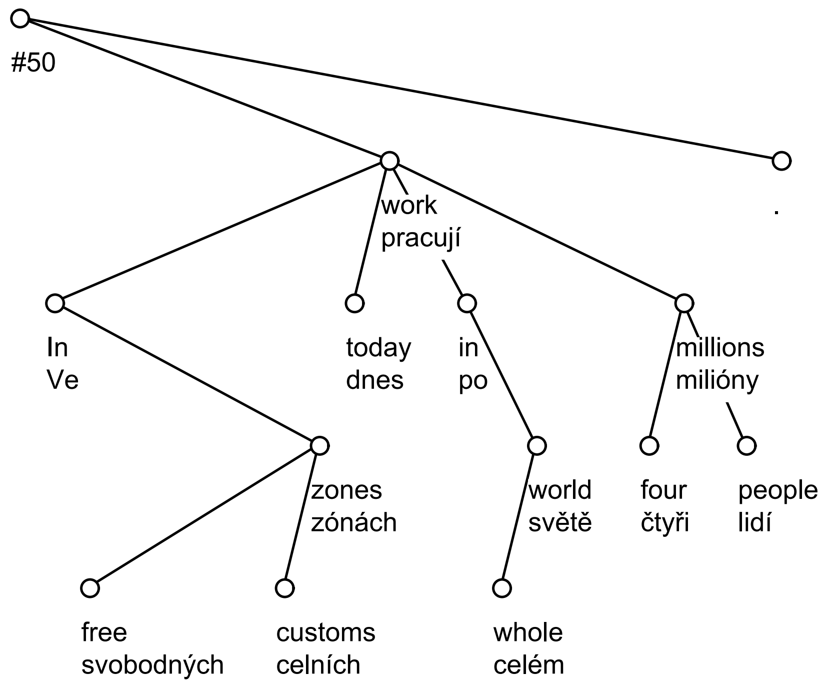

Figure 1.9 Sentence (6) correctly parsed by Collins' statistical parser

It is interesting to note than the Collins' statistical parser makes "human-like" errors[16] in most cases, as exemplified in the sentence (7):

(7) Uvádí se v analýze Komerční banky, kterou jsme dostali minulý týden.

It is presented in the Komerční bank's analysis, which we obtained last week.

The relative clause was (semantically) incorrectly determined to be a modifier of the noun *banka* 'bank' (instead of *analýza* 'analysis') because the morphological tags of both *banka* 'bank' and *analýza* 'analysis' are possible antecedents for the relative pronoun *kterou* 'which' (based on number and gender agreement).

We can also examine the nature of the errors the parser makes outside the project to obtain very important material for the study of syntactic irregularities and idiosyncrasies.

4. MERGING THE MORPHOLOGICAL AND THE ANALYTICAL SYNTACTIC LEVEL

As has already been mentioned, the texts used for PDT come from the Czech National Corpus. They are reasonably but not 100% clean[17] and already have an SGML markup (Fig. 1.1). These texts are then morphologically analyzed, and after that, they are annotated simultaneously on both the morphological as well as the analytical levels (see Fig. 1.10 for a general scheme)[18]. The separately produced annotations must be then merged into a single resource. Due to the less than perfect state of the input texts, some manual corrections (such as adding or deleting a wrongly introduced sentence boundary) are done to the original markup, making the merging task nontrivial. The merging procedure is a semi-automatic process; any discrepancies must be resolved manually.

5. TECTOGRAMMATICAL LEVEL

The third level of annotation, the tectogrammatical level, aims to describe the linguistic meaning of a sentence. It also uses the dependency framework; in fact, it is at this level where the Functional Generative Perspective ([21]) is followed most closely (and for the first time applied to real data).

[16] Of a semantically slightly challenged human, that is; purely syntactic errors are truly rare.
[17] Duplicates are removed, numbers normalized and marked, sentence breaks identified, etc.
[18] Obviously, we would have more (pre)processing options if we could do the morphological annotation first in full and the analytical level afterwards; unfortunately, various organizational, human resource, and funding constraints prevented that from happening.

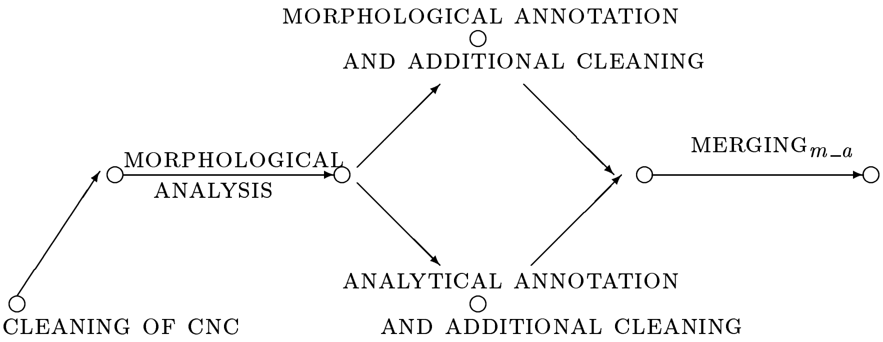

Figure 1.10 The scheme of merging the morphological and the analytical levels

5.1 FROM THE ANALYTICAL TO THE TECTOGRAMMATICAL LEVEL OF REPRESENTATION

In comparison to the analytic tree structures (ATS), the tectogrammatical tree structures (TGTS) have the following characteristics:

(a) only autosemantic (lexical) words have a node of their own, while the correlates of function words (auxiliaries, prepositions etc.) are attached as indices to the autosemantic words to which they "belong" (auxiliaries and conjunctions to lexical verbs, prepositions to nouns, etc.);

(b) nodes are added in case of clearly specified deletions on the surface level;

(c) non-projectivity (i.e., crossing of edges with each other or with perpendiculars incident to nodes) is not allowed; the relevant nodes are rearranged so that the condition of projectivity is met in the TGTS; the rearrangament is not arbitrary, though: it follows from the information structure of the sentence (see also (e) below);

(d) analytic functions are substituted by tectogrammatical functions (such as Actor/Bearer, Patient, Addressee, Origin, Effect, different kinds of Circumstantials);

(e) basic features of the information structure of the sentences (Topic-Focus Articulation) are added.

5.2 TECTOGRAMMATICAL LEVEL OF REPRESENTATION

A detailed, though still tentative description of the attributes and their values assigned to nodes of the TGTS is given in [12]. For the purpose of perspicuity we give here a brief overview of the character of the labels; Table 1.A.2 in Appendix contains a list of all the attributes as prepared for the annotators (at this stage of the project).

Each label consists basically of three sets of attributes: the lexical value of the word (in the present phase, this value consists of the lemma attribute: *trlemma*), the so-called (morphosyntactic) grammatemes, reflecting the meaning of morphological categories, and the so-called functors, corresponding to syntactic functions (for the difference between grammatemes and functors, cf. the writings of the Functional Generative Description, esp. [20], [18], [21]). In addition to these three parts, there is an attribute *tfa* attached to each node, capturing the basic features of the topic-focus articulation of the sentence; the three values of this attribute are T(opic), F(ocus) and C(ontrast).

Among the (morphosyntactic) grammatemes there are attributes for three kinds of modality (sentential, verbal, and that expressed by modal verbs: *sentmod, verbmod, deontmod*), *tense, aspect, iterativeness, number, gender*, and degrees of comparison (*degcmp*).

About 40 functors (*func*) are distinguished, such as actor/bearer, addressee, patient, origin, effect, cause, regard, concession, aim, manner, extent, substitution, accompaniment, locative, means, temporal, attitude, cause, regard, directional, benefactive, comparison; there are also specific functors for dependents on nouns, as e.g. material, appurtenance, restrictive and descriptive adjunct, the relation of identity (see Table 1.A.2 in Appendix; compare also a similar scenario based on an assignment of 'cases' as presented in [13], or the assignment of predicate-argument structure as suggested by [15] and performed by [17]). In addition to the above functors, we work with a more subtle differentiation of syntactic relations by means of the so-called syntactic grammatemes (in the present scheme, there are 12 syntactic grammatemes, attribute *gram*). The temporal relation can be supplemented by one of the grammatemes 'before', 'after', and 'on', accompaniment, regard and benefactive are accompanied by a positive or a negative grammateme (with/without, for/against). The possibility to express an uncertainty of the annotator is ensured (either by means of an assignment of a second functor or by adding a special mark denoting that the assignment of the value of the functor is doubted). There are no remaining "double" functors anymore (cf. Sect. 3.1). The node attributes listed in

Table 1.A.2 in Appendix are reserved for future annotation efforts, e.g. marking the coreference relation, or some technical purpose (marking of deleted/inserted nodes.)

In order to preserve the form of a tree for the TGTS, we work with a special node of coordination, with values such as conjunction, disjunction, gradation, adversative relation, consequence, reason and apposition.

5.3 AUTOMATIC AND MANUAL PROCESSING

The transduction of the analytic trees to the tectogrammatical ones is conceived of in two phases:

(a) an automatic tree 'pruning' and transformation,

(b) a manual procedure (with the help of specifically designed software tools).

In the following sections we focus on the first of the two procedures, namely on the automatic procedure translating analytic tree structures to tectogrammatical tree structures. More detailed rules for building the final TGTS manually can be found in [11] and [12].

5.4 AUTOMATIC PREPROCESSING

The input to the automatic procedure are the analytic trees. The main task of the procedure is to reduce the number of nodes in the tree by creating complex labels of the nodes that represent more than one word (as a result of the attachment of synsemantic words to the autosemantic ones, see point 5.1 (a)) and to translate the values of synsemantic words into the attributes of autosemantic words. The nodes in the pre-processed tree keep the values of all the analytical attributes and newly created attributes are added for the description of the properties of the TGTS briefly characterized in Section 5.2. The nodes deleted in the automatic procedure are in fact only marked as hidden (and are not considered to be part of the TGTS nor are they displayed on the screen), but they remain in the tree structure and can be examined if needed. Thus no information gets lost for later research and analysis.

5.5 TOOLS

The same macro programming language on the analytical level has been taken advantage of within the tree structure editor, since it is a powerful tool for handling tree structures. The following operations are available:

(i) get a value of a given attribute of a given node,

(ii) assign an attribute a value,

(iii) find parent of a node,

(iv) find the left or right brother of a node,

(v) cut and paste a subtree.

The macros work either in an interactive mode, so that they can be run in the editor for the displayed tree, or in a batch mode, for preprocessing all the trees in a file.

Automatic Tree Structure Modifications. The following tree modifications are performed fully automatically in a batch mode, before the annotators get the data for manual work.

(i) Merge the verb with its auxiliary nodes (having the category `AuxV` or `AuxT` as their analytical function in ATS) and assign the values of the grammatemes of tense and verb modality to the respective verb on the basis of the lexical values of these auxiliary nodes. The lemmas of the merged nodes are concatenated into the trlema attribute of the verb. The original auxiliary nodes are deleted.

(ii) Merge the modal verb with the autosemantic verb that depends on it in the ATS. This procedure works in three steps: the tree is rearranged so that the modal verb depends on the autosemantic verb, the value for the attribute deontmod of the latter verb is assigned according to the lexical value of the modal verb, and the modal verb node is deleted.

(iii) Merge the nodes of complex prepositions and complex conjunctive expressions into a single node.

(iv) Delete the nodes for prepositions; the lexical values of the prepositions are temporarily 'preserved' as values of the attribute fw (function word) of the respective governing noun and wait for further (manual) treatment. In case of coordination the prepositions are added to all coordinated nodes.

(v) Delete the nodes representing subordinating conjunctions; the lemma of a conjunction is again 'preserved' as a value of the 'fw' attribute of the dependent verb and thus prepared for further (manual) treatment. In case of coordination the conjunction is added to all coordinated nodes.

(vi) Switch the direction of dependency for certain numerals and counted nouns (the numeral governs the noun in some cases on the analytical level, while in the tectogrammatical representation the noun should always be the governing node).

(vii) Change the governor of the predicative complement; the complement in the TGTS depends on the verb (while in the ATS the dependency relation on the verb was only implicit, with the explicit relation to the respective noun or adjective). Assign the number of the original governor to the cornum attribute of the complement (thus marking a trace where the complement was moved from).

(viii) Delete all auxiliary nodes with the analytic function `AuxX`.

(ix) Fill the unresolved attribute values by one of the following special values: `'???'` indicates that the value needs to be processed manually, the value `NIL` is temporarily assigned to the attributes for which this value is expected to apply in most cases (e.g. attribute antec), and the value `NA` is assigned to the nodes where the given attribute is not applicable (e.g. tense for nouns).

Procedures that can be called manually for selected subtrees. In addition to the procedures that are invoked automatically (i.e. in the batch mode before the manual annotation starts), the tree editor offers a possibility for the human annotators to invoke certain procedures in case they need to carry out some rather complex modification of the tree structure. The following tasks can be performed by using a single keystroke:

(i) Merge the current node with its mother into a single complex node.

(ii) Merge the current node as a function word with its mother. This procedure is called for prepositions or conjunctions not annotated in the ATS as such, and therefore not touched automatically).

(iii) Delete (hide) or undelete (restore) a subtree.

(iv) Add a node for an Actor of the given verb. This procedure is called when the subject of the verb is missing in the ATS (due to the prodrop character of Czech). The new node is added as the daughter of the verb that is immediately “before” it. Mark the node by adding the value ELID in the attribute elided. Similar procedures can be used for other cases of adding nodes.

(v) Add a node for a missing verb.

(vi) Add a node as a daughter of the selected node. This procedure is used for adding the missing complements of the governor.

5.6 ILLUSTRATION

In order to illustrate the differences between the ATS and the TGT-S and the transformation described above, we show here the two tree structures for the Czech sentence

(8) Kdo chce investovat dvě stě tisíc korun do nového automobilu, nelekne se, že benzín byl změnou zákona trochu zdražen.

He who wants to invest two hundred thousand crowns into a new car does not get frightened by the fact that gas was made a little bit more expensive due to a law change.

Fig. 1.11 is an ATS of the sentence (8), in which each word (as well as each punctuation mark) has a node of its own; only the lexical tags and the analytic functions are displayed.

In a (dramatically) simplified TGTS of the same sentence (Fig. 1.12) the tags written in lowercase letters denote the lexical values of the nodes; the uppercased tags are values of the dependency relations (attribute *func*) together with some of the grammatemes (*deontmod*, *func* and the *tfa* attribute). The value `DSPP` at the main verb of the sentence denotes that the given sentence is a part of a direct speech. For the purpose of perspicuity of the trees, morphological grammatemes other than those transformed from the function words occurring in the ATS are not displayed in our illustrative example. The tags T and F are the values of the attribute *tfa* capturing the backbone of the topic-focus articulation (information structure) of the sentence.

6. PDT VERSIONS 1.0 AND 2.0

The PDT version 1.0 is the full version of the PDT, containing about three times more tokens and sentences than the PDT version 0.5[19].

The PDT version 1.0 contains complete manual annotation on the morphological and analytical levels. The volume on the morphological level is slightly higher (about 1.8M tokens), because a separate set of training data is needed to train taggers that might be used in the tra-

[19]Version 0.5 ("halfway through", rather optimistically) released in 1998 (http://ufal.ms.mff.cuni.cz/pdt/pdt.html) contains 26610 sentences and 456705 tokens.

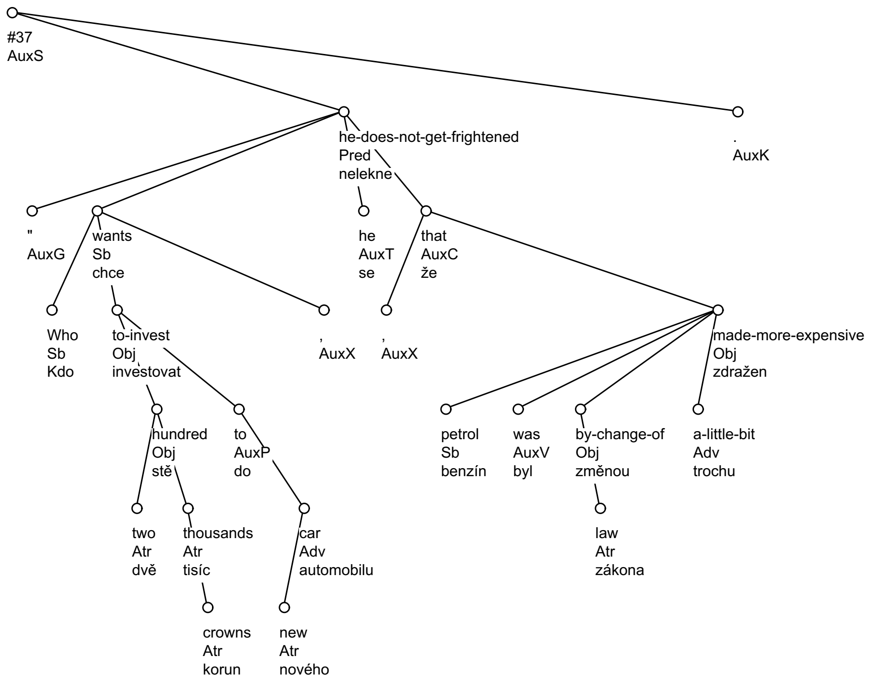

Figure 1.11 A Simplified ATS of sentence (8)

ditional serial[20] approach to statistical parsing. On the analytical level, about 1.3M tokens in about 90k sentences are annotated.

The annotation effort resulting in PDT version 1.0, including a preparation phrase and a checking phase, took five years. A total of 22 people have been involved in one phase of the project or another, with about 17 simultaneously at the peak time. The total cost of the project can be determined only roughly, given the various funding sources and schemes; it is estimated at about $600,000 over the five-year period.

The PDT version 2.0 will add the final third level of annotation (tectogrammatical annotation) to PDT version 1.0. It will be available with a reduced amount of data as preliminary "version 1.5" during 2002, and the final data volume will be reached at the end of 2004. The current plan is to annotate the tectogrammatical structure (deleting and adding

[20]If automatic tagging and parsing of a new text is done in this order, it is better to train the parser on an automatically tagged training corpus using an identical tagger (cf. also Fig. 1.4). The tagger must obviously be trained on different training data to achieve realistic results.

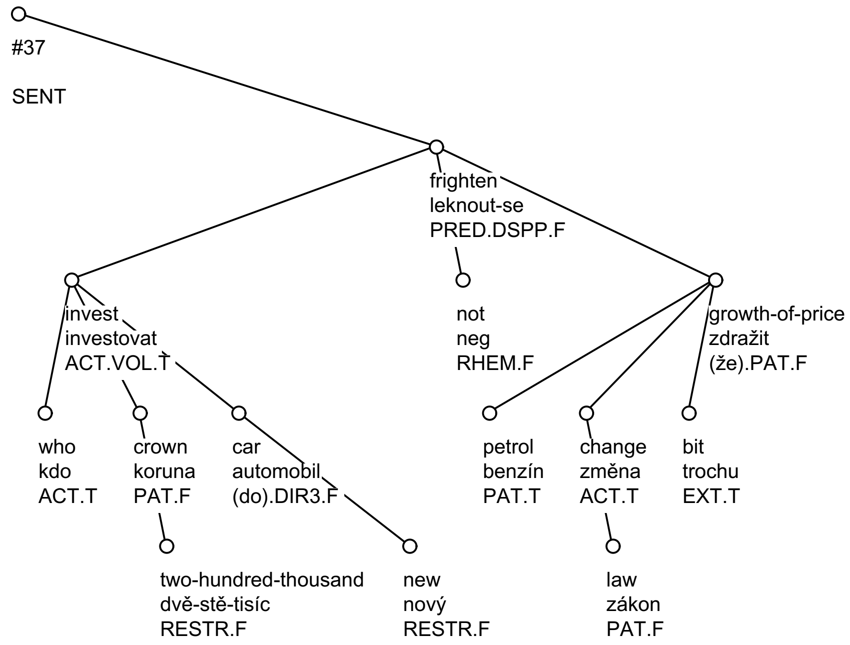

Figure 1.12 A Simplified TGTS of sentence (8)

nodes, changing the "deep" word order etc.) first, and then annotate the rest, adding values to attributes one by one or in small groups. We do not expect all the attributes described in Table 1.A.1 in Appendix to be present[21] in either version, but an example file will be provided with all of them, in as high volume as possible. After 2004, the remaining attributes will continue to be filled in by manual and enhanced semi-automatic procedures.

The total effort resulting eventually in PDT version 2.0 will take roughly the same resources as version 1.0 did (slightly less people but somewhat higher total estimated expenses). Also, errors encountered at the lower two levels of annotation will be continuously corrected during this phase, and re-released in version 2.0.

[21] Currently, a high-volume annotation effort has started for the attributes *func* and *tfa*. Also, many morphosyntactic attributes, such as *number*, *tense* and many others can be and will be filled in automatically.

7. CONCLUSION

Everybody certainly agrees that building a treebank is a difficult task. Our belief is, however, that all the hard work will pay off in that not only we who are building it, but all computational linguists interested in morphology and syntax of natural languages in general, and of Czech and other inflectional and free word order languages in particular, will benefit from its existence. The building of the treebank has been very fruitful even now, two-thirds of the way through the whole treebank annotation.

Acknowledgments

The project was started by the support from the grant GAČR (Formal specification of language structures) No. 405/96/0198, and the annotation process has been made possible by the grant GAČR No. 405/96/K214, by the project of the Ministry of Education of the Czech Republic No. VS96151, and by the NSF grant #IIS-9732388 through Johns Hopkins University, Baltimore. The development of some software tools used in this project has been supported by the grant GAČR No. 405/95/0190 and by an individual grant OSF RSS/HESP 1996/195. This contribution is an updated version of two papers presented at the ATALA treebanks workshop in Paris, June 18-19, 1999. Section 5 was in a preliminary form published in Prague Bulletin of Mathematical Linguistics No. 71, pp. 5-12.

We would like to thank all those involved in the project, specifically Petr Sgall, and all the annotators who manually processed the large volume of text. We would also like to thank those who created the indispensable software tools used throughout the project, most notably Michal Křen and Petr Pajas, and Michael Collins for providing us with his parser adapted for Czech. The authors of the present paper gratefully acknowledge the valuable share of the co-authors of one of the original papers in the research that has led to the results reported here.

Appendix

afun	*Description*
`Pred`	Predicate if it depends on the added root node (main predicate)
`Sb`	Subject
`Obj`	Object
`Adv`	Adverbial (without a detailed type distinction)
`Atv`	Complement; technically depends on its non-verbal governor
`AtvV`	Complement; if only one governor is present (the verb)
`Atr`	Attribute
`Pnom`	Nominal predicate's nominal part, depends on the copula "to be"
`AuxV`	Auxiliary Verb "to be" (být)
`Coord`	Coordination node
`Apos`	Apposition node
`AuxT`	Reflexive particle *se*, lexically bound to its verb
`AuxR`	Reflexive particle *se*, which is neither `Obj` nor `AuxT` (passive)
`AuxP`	Preposition, or a part of compound preposition
`AuxC`	Conjunction (subordinate)
`AuxO`	(Superfluously) referring particle or emotional particle
`AuxZ`	Rhematizer or other mode acting to stress another constituent
`AuxX`	Comma (but not the main coordinating comma)
`AuxG`	Other graphical symbols not classified as AuxK
`AuxY`	Other words, such as particles without specific (syntactic) function, parts of lexical idioms, etc.
`AuxS`	The (artificially created) root of the tree (#)
`AuxK`	Punctuation at the end of sentence or direct speech or citation clause
`ExD`	Ellipsis handling (Ex-Dependency): function for nodes which "pseudo-depend" on a mode on which they would not if there were no ellipsis
`AtrAtr` `AdvAtr` `ObjAtr`	A node (analytical function: an attribute) which could depend also on its governor's governor (and have the appropriate other function). There must be no semantic or situational difference between the two cases (or more, in case of several attributes depending on each other).

Table 1.A.1 Values of the analytical function attribute (STags)

<table>
<tr><th>Attribute</th><th>Values (any value is possible if the list is not given)</th></tr>
<tr><td>trlemma</td><td>string</td></tr>
<tr><td>gender</td><td><code>ANIM|INAN|FEM|NEUT|NA|???</code></td></tr>
<tr><td>number</td><td><code>SG|PL|NA|???</code></td></tr>
<tr><td>degcmp</td><td><code>POS|COMP|SUP|NA|???</code></td></tr>
<tr><td>tense</td><td><code>SIM|ANT|POST|NA|???</code></td></tr>
<tr><td>aspect</td><td><code>PROC|CPL|RES|NA|???</code></td></tr>
<tr><td>iterativeness</td><td><code>IT1|IT0|NA|???</code></td></tr>
<tr><td>verbmod</td><td><code>IND|IMP|CDN|NA|???</code></td></tr>
<tr><td>deontmod</td><td><code>DECL|DEB|HRT|VOL|POSS|PERM</code>
<code>|FAC|NA|???</code></td></tr>
<tr><td>sentmod</td><td><code>ENUNC|EXCL|DESID|IMPER|INTER|NA|???</code></td></tr>
<tr><td>tfa</td><td><code>T|F|C|NA|???</code></td></tr>
<tr><td>func</td><td><code>|ACT|PAT|ADDR|EFF|ORIG|ACMP|ADVS|AIM|APP|APPS|ATT</code>
<code>|BEN|CAUS|CNCS|COND|CONJ|COMPL|CPR|CRIT|CSQ|CTERF</code>
<code>|DENOT|DES|DIFF|DIR1|DIR2|DIR3|DISJ|ETHD|EXT</code>
<code>|FRWH|GRAD|ID|INTF|INTT|HER|LOC|MANN|MAT|MEANS|MOD</code>
<code>|NORM|PAR|PREC|REAS|REG|RESL|RESTR|RHEM|RSTR|SUBS</code>
<code>|TFHL|THL|THO|TOWH|TPAR|TSIN|TTILL|TWHEN|VOC|VOCAT</code>
<code>|NA|SENT|???</code></td></tr>
<tr><td>gram</td><td><code>0|GNEG|DISTR|APPX|GPART|GMULT|VCT|PNREL|ON|BEF|AFT</code>
<code>|JAFT|INTV</code></td></tr>
<tr><td>reltype</td><td><code>CO|PA|NIL|???</code></td></tr>
<tr><td>fw</td><td>string</td></tr>
<tr><td>phraseme</td><td>string</td></tr>
<tr><td>del</td><td><code>ELID|ELEX|EXPN|NIL|???</code></td></tr>
<tr><td>quoted</td><td><code>QUOT|NIL|???</code></td></tr>
<tr><td>dsp</td><td><code>DSP|DSPP|NIL|???</code></td></tr>
<tr><td>coref</td><td>ID number (pointer)</td></tr>
<tr><td>cornum</td><td>ID number (pointer)</td></tr>
<tr><td>corstn</td><td><code>PREV|NIL|???</code></td></tr>
<tr><td>antec</td><td>all 'func' values</td></tr>
</table>

Table 1.A.2 List of the TGTS attributes and their values

References

[1] Bémová Alla, Buráňová Eva, Hajič Jan, Kárník Jiří, Pajas Petr, Panevová Jarmila, Urešová Zdeňka and Jan Štěpánek. (1997). *Anotace na analytické rovině - příručka pro anotátory* [Annotation on the Analytical Level - Annotator's Guidelines], Technical Report #4 (draft), ÚFAL MFF UK, Prague, Czech Republic (in Czech).

[2] Collins, Michael. (1997). Statistical Parser Based on Bigram Lexical Dependencies. In *Proceedings of the 35th Annual Meeting of the ACL/EACL'97*, pp. 16-23, Madrid, Spain.

[3] Collins, Michael, Hajič Jan, Brill Eric, Ramshaw Lance, and Christopher Tillmann. (1999). A Statistical Parser of Czech. In *Proceedings of 37th ACL'99*, pp. 505–512, University of Maryland, College Park, June 22-25.

[4] Czech National Corpus (CNC). `http://ucnk.ff.cuni.cz`.

[5] Hajič, Jan. (1998). Building a Syntactically Annotated Corpus: The Prague Dependency Treebank. In *Issues of Valency and Meaning. Studies in Honor of Jarmila Panevová*, ed. Eva Hajičová, pp. 106-132, Karolinum, Charles University Press, Prague, Czech Republic.

[6] Hajič, Jan. (in press). *Disambiguation of Rich Inflection (Computational Morphology of Czech).* Charles University Press - Karolinum.

[7] Hajič, Jan, Brill, Eric, Collins, Michael, Hladká, Barbora, Jones, Douglas, Kuo, Cynthia, Ramshaw, Lance, Schwartz, Oren, Tillmann, Christopher and Daniel Zeman. (1998). Core Natural Language Processing Technology Applicable to Multiple Languages: Workshop98 Final Report for the 1998 Language Engineering Workshop for Students and Professionals: Integrating Research and Education, Center for Language and Speech Processing, Johns Hopkins University, Baltimore, MD, Research Note 37.

[8] Hajič, Jan, and Eva Hajičová. (1997). Syntactic Tagging in the Prague Tree Bank. In *Proceedings of the Second European Seminar "Language Applications for a Multilingual Europe"* (ed. by R. Marcinkeviciene and N. Volz), pp. 55-68, Kaunas.

[9] Hajič, Jan, and Barbora Hladká. (1997). Probabilistic and Rule-Based Tagger of an Inflective Language - a Comparison. In *Proceedings of the 5th Conference on Applied Natural Language Processing*, pp. 111-118, Washington, USA.

[10] Hajič, Jan, and Barbora Hladká. (1998). Tagging Inflective Languages: Prediction of Morphological Categories for a Rich, Structured Tagset. In *Proceedings of COLING-ACL Conference*, pp. 483-490, Montreal, Canada.

[11] Hajičová, Eva. (2000). Dependency-Based Underlying-Structure Tagging of a Very Large Corpus, to be published in a special issue of T.A.L. no. 1, pp. 47-66.

[12] Hajičová, Eva, Panevová Jarmila, and Petr Sgall. (1998). Language Resources Need Annotations To Make Them Really Reusable: The Prague Dependency Treebank. In *Proceedings of the First International Conference on Language Resources & Evaluation.* Granada, Spain, pp. 713-718, Paris:ELRA.

[13] Chan Keh-Jiann et al. (2000). The CKIP Chinese Treebank, this volume.

[14] Křen, Michal. (1996). *GRAPH editor.* MSc. Thesis, Institute of Formal and Applied Linguistics, Charles University, Prague, Czech Republic

[15] Marcus M. P., Kim G., Marcinkiewicz M. A. et al. (1994). The Penn Treebank: Annotating Predicate Argument Structure. In *Proceedings of the ARPA Human Language Technology Workshop.* San Francisco: Morgan Kaufmann.

[16] Marcus M. P., Santorini, Beatrice and Marcinkiewicz M. A. (1993). Building a large annotated corpus of English: the Penn Treebank. *Computational Linguistics, 19(2)*, 313-330.

[17] Palmer, M., Dang, H.T., and J. Rosenzweig. (2000). Sense Tagging the Penn Treebank. In: *Proceedings of LREC'00*, Athens, Greece.

[18] Panevová, Jarmila. (1980). Formy a funkce ve stavbě české věty [Forms and functions in the structure of the Czech sentence], Prague:Academia.

[19] Prague Dependency Treebank (PDT). `http://ufal.ms.mff.cuni.cz/pdt/pdt.html`.

[20] Sgall, Petr. (1967). *Generativní popis jazyka a česká deklinace.* Academia, Prague, Czech Republic.

[21] Sgall, Petr, Hajičová Eva, and Jarmila Panevová. (1986) *The Meaning of the Sentence and Its Semantic and Pragmatic Aspects.* Reidel Publishing Company, Dordrecht, Netherlands, Academia, Prague, Czech Republic.

[22] Šmilauer, Vladimír. (1969). *Novočeská skladba* [Syntax of Contemporary Czech], 3rd ed., SPN, Prague, Czech Republic.