# Synthesis of Czech Sentences from Tectogrammatical Trees*

Jan Ptáček, Zdeněk Žabokrtský

Institute of Formal and Applied Linguistics
Malostranské náměstí 25, 11800 Prague, Czech Republic
{ptacek,zabokrtsky}@ufal.mff.cuni.cz

**Abstract.** In this paper we deal with a new rule-based approach to the Natural Language Generation problem. The presented system synthesizes Czech sentences from Czech tectogrammatical trees supplied by the Prague Dependency Treebank 2.0 (PDT 2.0). Linguistically relevant phenomena including valency, diathesis, condensation, agreement, word order, punctuation and vocalization have been studied and implemented in Perl using software tools shipped with PDT 2.0. BLEU score metric is used for the evaluation of the generated sentences.

## 1 Introduction

Natural Language Generation (NLG) is a sub-domain of Computational Linguistics; its aim is studying and simulating the production of written (or spoken) discourse. Usually the discourse is generated from a more abstract, semantically oriented data structure. The most prominent application of NLG is probably transfer-based machine translation, which decomposes the translation process into three steps: (1) analysis of the source-language text to the semantic level, maximally unified for all languages, (2) transfer (arrangements of the remaining language specific components of the semantic representation towards the target language), (3) text synthesis on the target-language side (this approach is often visualized as the well-known machine translation pyramid, with hypothetical interlingua on the very top; NLG then corresponds to the right edge of the pyramid). The task of NLG is relevant also for dialog systems, systems for text summarizing, systems for generating technical documentation etc.

In this paper, the NLG task is formulated as follows: given a Czech tectogrammatical tree (as introduced in Functional Generative Description, [1], and recently elaborated in more detail within the PDT 2.0 project[1,2]), generate a Czech sentence the meaning of which corresponds to the content of the input tree. Not surprisingly, the presented research is motivated by the idea of transfer-based machine translation with the usage of tectogrammatics as the highest abstract representation.

* The research has been carried out under projects 1ET101120503 and 1ET201120505.

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

[2] In the context of PDT 2.0, sentence synthesis can be viewed as a process inverse to treebank annotation.

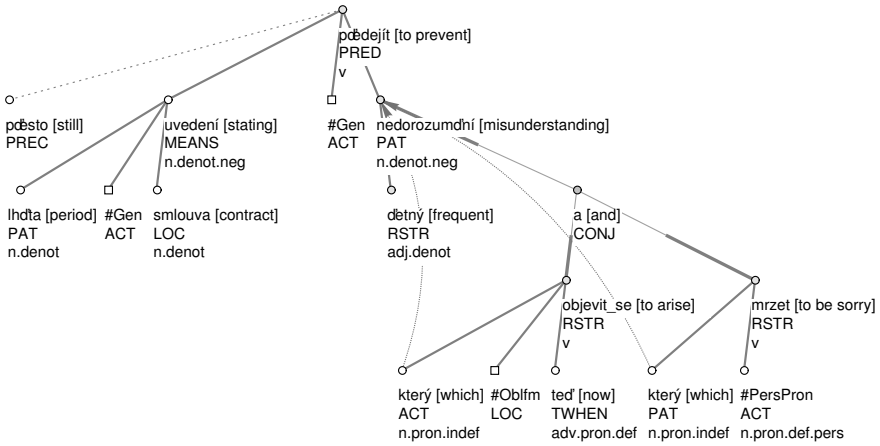


**Fig. 1.** Simplified t-tree fragment corresponding to the sentence *'Přesto uvedením lhůty ve smlouvě by se bylo předešlo četným nedorozuměním, která se nyní objevila a která nás mrzí.'* (But still, stating the period in the contract would prevent frequent misunderstandings which have now arisen and which we are sorry about.)

In the PDT 2.0 annotation scenario, three layers of annotation are added to Czech sentences: (1) *morphological layer* (m-layer), on which each token is lemmatized and POS-tagged, (2) *analytical layer* (a-layer), on which a sentence is represented as a rooted ordered tree with labeled nodes and edges corresponding to the surface-syntactic relations; one a-layer node corresponds to exactly one m-layer token, (3) *tectogrammatical layer* (t-layer), on which the sentence is represented as a deep-syntactic dependency tree structure (t-tree) built of nodes and edges (see Figure 1). T-layer nodes represent auto-semantic words (including pronouns and numerals) while functional words such as prepositions, subordinating conjunctions and auxiliary verbs have no nodes of their own in the tree. Each tectogrammatical node is a complex data structure – it can be viewed as a set of attribute-value pairs, or even as a typed feature structure. Word forms occurring in the original surface expression are substituted with their t-lemmas. Only semantically indispensable morphological categories (called grammatemes) are stored in the nodes (such as number for nouns, or degree of comparison for adjectives), but not the categories imposed by government (such as case for nouns) or agreement (congruent categories such as person for verbs or gender for adjectives). Each edge in the t-tree is labeled with a functor representing the deep-syntactic dependency relation. Coreference and topic-focus articulations are annotated in t-trees as well. See [2] for a detailed description of the t-layer.

The pre-release version of the PDT 2.0 data consists of 7,129 manually annotated textual documents, containing altogether 116,065 sentences with 1,960,657 tokens (word forms and punctuation marks). The t-layer annotation is available for 44 % of the whole data (3,168 documents, 49,442 sentences).

## 2 Task Decomposition

Unlike stochastic 'end-to-end' solutions, rule-based approach, which we adhere to in this paper, requires careful decomposition of the task (due to the very complex nature of the task, a monolithic implementation could hardly be maintainable). The decomposition was not trivial to find, because many linguistic phenomena are to be considered and some of them may interfere with others; the presented solution results from several months of experiments and a few re-implementations.

In our system, the input tectogrammatical tree is gradually changing – in each step, new node attributes and/or new nodes are added. Step by step, the structure becomes (in some aspects) more and more similar to a-layer tree. After the last step, the resulting sentence is obtained simply by concatenating word forms which are already filled in the individual nodes, the ordering of which is also already specified.

A simplified data-flow diagram corresponding to the generating procedure is displayed in Figure 2. All the main phases of the generating procedure will be outlined in the following subsections.

### 2.1 Formeme Selection, Diatheses, Derivations

In this phase, the input tree is traversed in the depth-first fashion, and so called *formeme* is specified for each node. Under this term we understand a set of constraints on how the given node can be expressed on the surface (i.e., what morphosyntactic form is used). Possible values are for instance simple case *gen* (genitive), prepositional case *pod+7* (preposition *pod* and instrumental), *v-inf* (infinitive verb),[3] *že+v-fin* (subordinating clause introduced with subordinating conjunction *že*), *attr* (syntactic adjective), etc.

Several types of information are used when deriving the value of the new *formeme* attribute. At first, the valency lexicon[4] is consulted: if the governing node of the current node has a valency frame, and the valency frame specifies constraints on the surface form for the functor of the current node, then these constraints imply the set of possible formemes. In case of verbs, it is also necessary to specify which diathesis should be used (active, passive, reflexive passive etc.; depending on the type of diathesis, the valency frame from the lexicon undergoes certain transformations). If the governing node does not have a valency frame, then the formeme default for the functor of the current node (and subfunctor, which specifies the type of the dependency relations in more detail) is

[3] It is important to distinguish between infinitive as a formeme and infinitive as a surface-morphological category. The latter one can occur e.g. in compound future tense, the formeme of which is not infinitive.

[4] There is the valency lexicon PDT-VALLEX ([3]) associated with PDT 2.0. On the t-layer of the annotated data, all semantic verbs and some semantic nouns and adjectives are equipped with a reference to a valency frame in PDT-VALLEX, which was used in the given sentence.

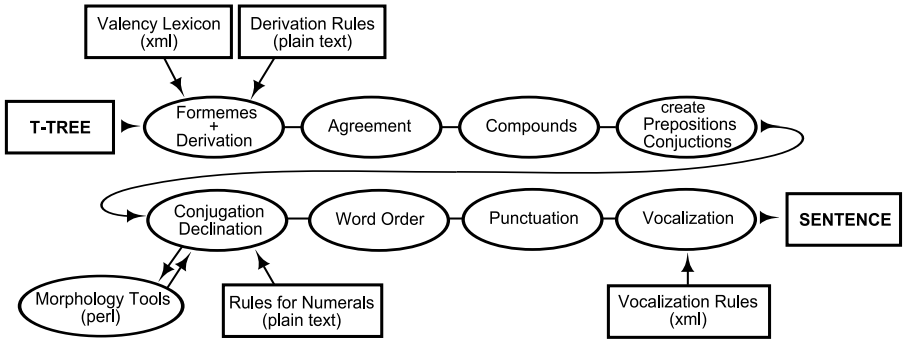


**Fig. 2.** Data-flow diagram representing the process of sentence synthesis.

used. For instance, the default formeme for the functor `ACMP` (accompaniment) and subfunctor `basic` is *s+7* (with), whereas for `ACMP.wout` it is *bez+2* (without).

It should be noted that the formeme constraints depend also on the possible word-forming derivations applicable on the current node. For instance, the functor APP (appurtenance) can be typically expressed by formemes *gen* and *attr*, but in some cases only the former one is possible (some Czech nouns do not form derived possessive adjectives).

### 2.2 Propagating Values of Congruent Categories

In Czech, which is a highly inflectional language, several types of dependencies are manifested by agreement of morphological categories (agreement in gender, number, and case between a noun and its adjectival attribute, agreement in number, gender, and person between a finite verb and its subject, agreement in number and gender between relative pronoun in a relative clause and the governor of the relative clause, etc.). As it was already mentioned, the original tectogrammatical tree contains those morphological categories which are semantically indispensable. After the formeme selection phase, value of case should be also known for all nouns. In this phase, oriented agreement arcs (corresponding to the individual types of agreement) are conceived between nodes within the tree, and the values of morphological categories are iteratively spread along these arcs until the unification process is completed.

### 2.3 Expanding Complex Verb Forms

Only now, when person, number, and gender of finite verbs is known, it is possible to expand complex verb forms where necessary. New nodes corresponding to reflexive particles (e.g. in the case of reflexiva tantum), to auxiliary verbs (e.g. in the case of complex future tense), or to modal verbs (if deontic modality of the verb is specified) are attached below the original autosemantic verb.

### 2.4 Adding Prepositions and Subordinating Conjunctions

In this phase, new nodes corresponding to prepositions and subordinating conjunctions are added into the tree. Their lemmas are already implied by the value of node formemes.

### 2.5 Determining Inflected Word Forms

After the agreement step, all information necessary for choosing the appropriate inflected form of the lemma of the given node should be available in the node. To perform the inflection, we employ morphological tools (generator and analyzer) developed by Hajič ([4]). The generator tool expects a lemma and a positional tag (as specified in [5]) on the input, and returns the inflected word form. Thus the task of this phase is effectively reduced to composing the positional morphological tag; the inflection itself is performed by the morphological generator.

### 2.6 Special Treatment of Definite Numerals

Definite numerals in Czech (and thus also in PDT 2.0 t-trees) show many irregularities (compared to the rest of the language system), that is why it seems advantageous to generate their forms separately. Generation of definite numerals is discussed in [6].

### 2.7 Reconstructing Word Order

Ordering of nodes in the annotated t-tree is used to express information structure of the sentences, and does not directly mirror the ordering in the surface shape of the sentence. The word order of the output sentence is reconstructed using simple syntactic rules (e.g. adjectival attribute goes in front of the governing noun), functors, and topic-focus articulation. Special treatment is required for clitics: they should be located in the 'second' position in the clause (Wackernagel position); if there are more clitics in the same clause, simple rules for specifying their relative ordering are used (for instance, the clitic *by* always precede short reflexive pronouns).

### 2.8 Adding Punctuation Marks

In this phase, missing punctuation marks are added to the tree, especially (i) the terminal punctuation (derived from the `sentmod` grammateme), (ii) punctuations delimiting boundaries of clauses, of parenthetical constructions, and of direct speeches, (iii) and punctuations in multiple coordinations (commas in expressions of the form *A, B, C and D*).

Besides adding punctuation marks, the first letter of the first token in the sentence is also capitalized in this phase.

### 2.9 Vocalizing Prepositions

Vocalization is a phonological phenomenon: the vowel *-e* or *-u* is attached to a preposition if the pronunciation of the prepositional group would be difficult without the vowel (e.g. *ve výklenku* instead of **v výklenku*). We have adopted vocalization rules precisely formulated in [7] (technically, we converted them into the form of an XML file, which is loaded by the vocalization module).

## 3 Implementation and Evaluation

The presented sentence generation system was implemented in ntred[5] environment for processing the PDT data. The system consists of approximately 9,000 lines of code distributed in 28 Perl modules. The sentence synthesis can also be launched in the GUI editor tred providing visual insight into the process.

As illustrated in Figure 2, we took advantage of several already existing resources, especially the valency lexicon PDT-VALLEX ([3]), derivation rules developed for grammateme assignment ([8]), and morphology analyzer and generator ([4]).

We propose a simple method for estimating the quality of a generated sentence: we compare it to the original sentence from which the tectogrammatical tree was created during the PDT 2.0 annotation. The original and generated sentences are compared using the BLUE score developed for machine translation ([9]) – indeed, the annotation-generation process is viewed here as machine translation from Czech to Czech. Obviously, in this case BLEU score does not evaluate directly the quality of the generation procedure, but is influenced also by the annotation procedure, as depicted in Figure 3.

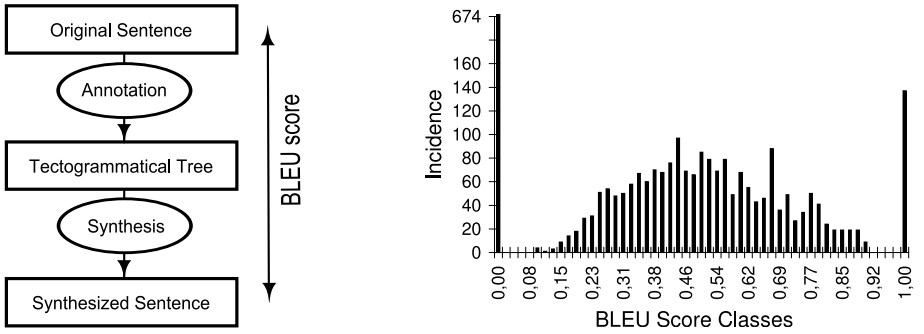


**Fig. 3.** Evaluation scheme and distribution of BLEU score in a development test sample counting 2761 sentences.

[5] http://ufal.mff.cuni.cz/~pajas

It is a well-known fact that BLEU score results have no direct common-sense interpretation. However, a slightly better insight can be gained if the BLEU score result of the developed system is compared to some baseline solution. We decided to use a sequence of t-lemmas (ordered in the same way as the corresponding t-layer nodes) as the baseline.

When evaluating the generation system on 2761 sentences from PDT 2.0 development-test data, the obtained BLEU score is **0.477**.[6] Distribution of the BLEU score values is given in Figure 3. Note that the baseline solution reaches only 0.033 on the same data.

To give the reader a more concrete idea of how the system really performs, we show several sample sentences here. The *O* lines contain the original PDT 2.0 sentence, the *B* lines present the baseline output, and finally, the *G* lines represent the automatically generated sentences.

(1) *O*: Dobře ví, o koho jde.
*B*: vědět dobrý jít kdo
*G*: Dobře ví, o koho jde.

(2) *O*: Trvalo to až do roku 1928, než se tento problém podařilo překonat.
*B*: trvat až rok 1928 podařit_se tento problém překonat
*G*: Trvalo až do roku 1928, že se podařilo tento problém překonat.

(3) *O*: Stejně tak si je i adresát výtky podle ostrosti a výšky tónu okamžitě jist nejen tím, že jde o něj, ale i tím, co skandál vyvolalo.
*B*: stejně tak být i adresát výtka ostrost a výška tón okamžitý jistý nejen jít ale i skandál vyvolat co
*G*: Stejně tak je i adresát výtky podle ostrosti a podle výšky tónu okamžitě jistý, nejen že jde o něj, ale i co skandál vyvolalo.

(4) *O*: Pravda o tom, že žvýkání pro žvýkání bylo odjakživa činností veskrze lidskou – kam paměť lidského rodu sahá.
*B*: pravda žvýkání žvýkání být odjakživa činnost lidský veskrze paměť rod lidský sahat kde
*G*: Pravda, že žvýkání pro žvýkání bylo odjakživa veskrze lidská činnost (kam paměť lidského rodu sahá).

## 4 Final Remarks

The primary goal of the presented work – to create a system generating understandable Czech sentences out of their tectogrammatical representation – has been achieved. This conclusion is confirmed by high BLUE-score values. Now we are incorporating the developed sentence generator into a new English-Czech

[6] This result seems to be very optimistic; moreover, the value would be even higher if there were more alternative reference translations available.

transfer-based machine translation system; the preliminary results of the pilot implementation seem to be promising.

As for the comparison to the related works, we are aware of several experiments with generating Czech sentences, be they based on tectogrammatics (e.g. [10], [11], [12]) or not (e.g. [13]), but in our opinion no objective qualitative comparison of the resulting sentences is possible, since most of these systems are not functional now and moreover there are fundamental differences in the experiment settings.

## References


1. Sgall, P.: Generativní popis jazyka a česká deklinace. Academia (1967)
2. Mikulová, M., Bémová, A., Hajič, J., Hajičová, E., Havelka, J., Kolářová, V., Lopatková, M., Pajas, P., Panevová, J., Razímová, M., Sgall, P., Štěpánek, J., Urešová, Z., Veselá, K., Žabokrtský, Z., Kučová, L.: Anotace na tektogramatické rovině Pražského závislostního korpusu. Anotátorská příručka. Technical Report TR-2005-28, ÚFAL MFF UK (2005)
3. Hajič, J., Panevová, J., Urešová, Z., Bémová, A., Kolářová-Řezníčková, V., Pajas, P.: PDT-VALLEX: Creating a Large-coverage Valency Lexicon for Treebank Annotation. In: Proceedings of The Second Workshop on Treebanks and Linguistic Theories, Vaxjo University Press (2003) 57–68
4. Hajič, J.: Disambiguation of Rich Inflection – Computational Morphology of Czech. Charles University – The Karolinum Press, Prague (2004)
5. Hana, J., Hanová, H., Hajič, J., Vidová-Hladká, B., Jeřábek, E.: Manual for Morphological Annotation. Technical Report TR-2002-14 (2002)
6. Ptáček, J.: Generování vět z tektogramatických stromů Pražského závislostního korpusu. Master's thesis, MFF, Charles University, Prague (2005)
7. Petkevič, V., ed.: Vocalization of Prepositions. In: Linguistic Problems of Czech. (1995) 147–157
8. Razímová, M., Žabokrtský, Z.: Morphological Meanings in the Prague Dependency Treebank 2.0. LNCS/Lecture Notes in Artificial Intelligence/Proceedings of Text, Speech and Dialogue (2005)
9. Papineni, K., Roukos, S., Ward, T., Zhu, W.J.: Bleu: a Method for Automatic Evaluation of Machine Translation. Technical report, IBM (2001)
10. Panevová, J.: Random generation of Czech sentences. In: Proceedings of the 9th conference on Computational linguistics, Czechoslovakia, Academia Praha (1982) 295–300
11. Panevová, J.: Transducing Components of Functional Generative Description 1. Technical Report IV, Matematicko-fyzikální fakulta UK, Charles University, Prague (1979) Series: Explizite Beschreibung der Sprache und automatische Textbearbeitung.
12. Hajič, J., Čmejrek, M., Dorr, B., Ding, Y., Eisner, J., Gildea, D., Koo, T., Parton, K., Penn, G., Radev, D., Rambow, O.: Natural Language Generation in the Context of Manchine Translation. Technical report, Johns Hopkins University, Baltimore, MD (2002)
13. Hana, J.: The AGILE System. Prague Bulletin of Mathematical Linguistics (78) (2001) 147–157