Tectogrammatical Annotation of the Wall Street Journal

Silvie Cinková Josef Toman Jan Hajič

Kristýna Čermáková Václav Klimeš Lucie Mladová

Jana Šindlerová Kristýna Tomšů Zdeněk Žabokrtský

January 19, 2009

1 Introduction

We are presenting the first results of a manual tectogrammatical annotation of the Wall Street Journal - Penn Treebank III. We call the WSJ-PTB texts and the annotation above them the **Prague English Dependency Treebank (PEDT)**. The current release (PEDT 1.0) is about 25% of the WSJ-PTB.

The Wall Street Journal section of the Penn Treebank was one of the first large manually annotated treebanks. It has become established as a standard reference corpus for statistical machine learning experiments. The PTB bracketing style was adopted by corpora of other languages, which strengthened the prominence of the original WSJ-PTB corpus. Although WSJ in practice is a restricted-domain corpus, which may affect its usability for general NLP tasks [1] (cf. e.g. Oepen [2] and Gildea [3], we believe that building an additional syntactico-semantic annotation on WSJ is sensible. After having built and refined the Prague Dependency Treebank, a one-million corpus of Czech 1990s newspaper texts with manual syntactico-semantic annotation [4], we have adapted the PDT annotation scheme to English. We decided to draw on a corpus manually annotated in a widely known format, since the option of comparing both annotation schemes can be particularly useful for some users. In addition, familiar text examples facilitate the understanding of the new annotation scheme by users, and, in turn, we benefit from the constant confrontation with the PTB bracketing style while creating the annotation guidelines [5]. Most importantly, the original manual annotation provided an excellent input for the conversion.

While creating the annotation guidelines, we made a tentative annotation of English spontaneous (but slightly edited) spoken dialogs [6], [7] in order to compensate for the style bias of WSJ-PTB and to make sure that the current annotation scheme would fit a broader range of styles than business press can offer.

[1] From the linguistic point of view the corpus domain restriction is not necessarily a drawback, given the linguistic research is consciously focused on local discourse patterns and local meanings (cf. e.g. [1]).

2 Background

2.1 Functional Generative Description and Tectogrammatical Representation

The **Functional Generative Description** (FGD) is a stratified formal language description based on the structuralist tradition, developed since the 1960s [8]. The unique contribution of FGD is the so-called **tectogrammatical representation (TR)**. It is implemented in a family of syntactico-semantically annotated treebanks. The treebanks are typically annotated at three layers:

- morphological layer (m-layer)
- analytical layer (a-layer)
- tectogrammatical layer (t-layer).

At the m-layer the text is still a sequence of strings with added tokenization, POS tagging, and lemmatization. Each token has its unique ID. The a-layer displays the sentences as dependency trees in which each token is represented by a node. The nodes are labeled with coarse syntactic labels. The topmost layer so far is the tectogrammatical layer (t-layer), which is based on the tectogrammatical representation (TR) proposed by FGD. Conceived as an underlying syntactic representation, the TR captures the **linguistic meaning of the sentence**. By *linguistic meaning* we understand "what has been said and can be perceived without any special knowledge of the situation" but with the common understanding of basic conversational implicatures, as well as with tolerance for redundance and vagueness. E.g. unlike a strictly logical representation, the tectogrammatical representation would not deal with the question whether in the sentence *John heard a cry* there must have been a cry for John to hear, or whether John might have mistakenly interpreted a sound he had heard as a cry. On the other hand, the tectogrammatical representation would indicate that something unexpressed on the surface is likely to be understood from the context or from the situation, or that something has been deliberately left underspecified; e.g., in the sentence *I told you last night* the tectogrammatical representation of the verb *to tell* would indicate that *something* (EFF), possibly about a mentioned matter (PAT) was told to somebody, and it would indicate whether these entities could be retrieved from the verbal context or not. (While the missing argument of *tell* is in this case likely to be retrievable from the context, some ellipses systematically express generalizations; e.g., *Peter can eat* [something, anything] *alone.*)

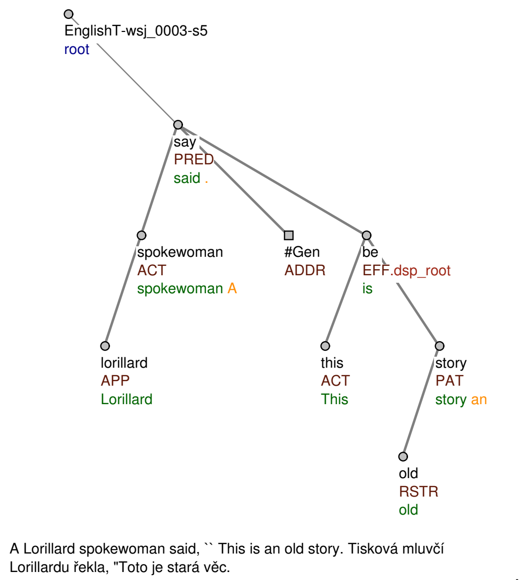
2.2 Tectogrammatical Annotation

Tectogrammatical annotation is to be held apart from the theoretical construct of tectogrammatical representation, as many annotation resolutions have still been introduced for technical and consistency reasons rather than being conditioned by the theory. The dependency treebanks of the PDT family are however being continuously refined, with the ambition of adequately reflecting the FGD as a linguistic description. That is done by a step-by-step uncovering and consistent tectogrammatical representation of lexical and structural patterns.

The basic description unit of the tectogrammatical annotation is the **sentence**. Each sentence is represented as a projective dependency tree with nodes and edges (henceforth **tectogrammatical tree structure** or **TGTS**). Only **content words** are represented by nodes. Each node has a semantic label ("functor"), which renders the semantic relation of the given node to its parent node. Function words are represented as attribute values in the internal structure of the respective nodes. The attribute values contain references to the analytical (surface-syntax) annotation layer instead of the forms of the function words themselves.[2] Tectogrammatical annotation, which draws on TR, captures the following aspects of text:

- syntactic and semantic dependencies
- argument structure (data interlinked with a lexicon)
- information structure (topic-focus articulation)
- grammatical and textual coreference
- ellipsis restoration
- information on lexical and syntactic derivation[3]
- semantically determined grammatical categories **(grammatemes)**[4]

2.2 shows the tectogrammatical tree structure (TGTS) of the sentence *A Lorillard spokewoman said: "This is an old story."*



Each sentence is identified with a unique identifier in the **technical root** of the tree (the topmost node). This node does not reflect any part of the sentence. The topmost linguistically relevant tectogrammatical node **(t-node)** is

[2] A more detailed specification of the annotation conventions is given by [5].
[3] so far Czech only
[4] just a tentative automatic insertion in English at the moment, not in this data

the predicate *said*, whose tectogrammatical lemma is *say*. The internal structure of this node contains references to the analytical (dependency surface-syntax) tree of the same sentence, in which each token is represented by a node. The references point to all analytical nodes **(a-nodes)** that affect the meaning unit rendered by the given t-node. We distinguish two types of references pointing to the analytical layer:

- reference to a content word
- reference to an auxiliary word.

The green strings in 2.2 represent the targets of the content-word references. The orange strings represent the targets of the auxiliary-word references. The predicate *say* has three obligatory participants: the Actor, the Addressee and the Effect (what is being said). The Addressee is underspecified, which is why a generated node with the t-lemma substitute #Gen (generalized) was inserted. In general, each occurrence of a word with argument structure (so far only verbs and verbal nouns in the English annotation) is interlinked with an instance (a **valency frame**) in the **valency lexicon**. When assigned to a lexicon frame, the occurrence of the given word must have the complete pattern of obligatory arguments **(inner participants)** determined by the valency lexicon. Generated nodes with t-lemma substitutes are inserted to complete the valency frame.

2.2 shows a few common semantic labels (functors) used in TGTS. The functors ACT, PAT, ADDR (in the figure), ORIG and EFF are labels for complementations which cannot repeat in the same predicate. About 70 functors in total are used in the annotation. It is partly functors for adjuncts, partly functors for semantic relations between conjuncts in coordinations, and a few functors which help organize cognitively specific syntactic structures such as comparisons. A complete list of functors along with the list of t-lemma substitutes is to be found in [5]. The functors indicate the semantic relation of a given node to its parent node. The node that modifies another node is governed by that node (except in shared modifiers in conjuncts). For instance, *Lorillard* modifies *spokewoman*, and the semantic relation between *Lorillard* and *spokewoman* is labelled as APP ("appurtenance"; i.e. association in a broader sense than ownership). The effective root of a direct speech subtree is marked with the note `dsp_root`.

3 The Original Penn Treebank

The Wall Street Journal section of the Penn Treebank [9] comprises approx. 1.25 million POS-tagged words in 49 208 sentences, which are manually annotated with constituency **bracketing** and **labels**. PTB-WSJ III keeps the PTB II [10] bracketing style [11]. Each bracket is labeled with one of the standard syntactic labels (NP, ADVP, PP, S, etc.). Since PTB II, the brackets are enriched with more detailed labeling. On the clausal level, the labels distinguish 5 types of clauses (subordinate clause, inverted question, inverted declarative sentence, direct wh-question and simple declarative clause). The phrase labels separate structural anomalies (lists, fragments, parentheses, reduced relative clauses, unlike coordinated phrases), heads of certain parts of speech (adjective, adverb, etc.), recurrent semantic units (e.g. quantifier phrases used within noun phrases) and transition phenomena (e.g, multi-word conjunctions like *as well as, not to mention*, etc., which have coordinative as well as subordinative features). On top of phrase and clause labels, non-terminal nodes can get **function tags**. The function tags mark specific linguistic phenomena, such as the nominal function of a gerundial clause *(Baking pies is fun., I do not mind about your leaving early.)*, "dative" alternation in certain verbs (to give), predicate complements *(I consider Kris a fool.)*, topicalization of a phrase by the left shift in the word order *(Of the 500 barbers in Philadelphia only 10 know what they are doing.)*, and several semantic labels of adjuncts (temporal, spatial, extent, etc.). The bracketing manual gives detailed information on linguistic phenomena which were captured systematically, along with several financial-speak-specific annotation templates.

4 Complementary Annotations

Several important annotations have been built above the PTB-WSJ texts since the release of the treebank. Two lexical sources were created and interlinked with the data:

- PropBank [12], the valency lexicon of verbs
- NomBank [13], the valency lexicon of nouns, which in fact also comprises lexicons of predicate nouns (the nominal components of light verb constructions), adjectives and adverbs.

Both lexicons are referenced by data annotations of argument structure.

- Annotation of flat noun phrases [14, 15]
- BBN Pronoun Coreference and Entity Type Corpus [16]

4.1 Flat Noun Phrases Annotation

Complex noun phrases like *an Air Force Contract* are left flat by the original Penn Treebank annotation. Vadas [15], [14] created a manual annotation of the almost 61,000 complex noun phrases in WSJ-PTB, making use of the entity annotation made by [16]. By adopting the basic priciples of the annotation of biomedical texts [17], Vadas et al. inserted labelled brackets around left-branching structures. The newly created constituents with noun heads were assigned the label NML, whereas those with adjectival heads were marked as JJP.

Hence, the phrase *Air Force contract*, in the original PTB bracketing represented as

```
(NP (NNP Air) (NNP Force) (NN contract))
```

is supplemented with an NML constituent that indicates that *Air Force* is a sub-NP structure within the entire phrase:

```
(NP
(NML (NNP Air) (NNP Force))
(NN contract))
```

4.2 BBN Corpus

Weischedel and Brunstein [16] created a stand-off annotation of pronoun coreference along with an annotation of a variety of entity and numeric types above WSJ-PTB. The entity annotation has been designed for question-answering tasks. It distinguishes 29 categories with subtypes. The most relevant for our annotation (see 6) are the following categories:

- Person Name
- Person Descriptor
- Facility Name
- Facility Descriptor
- Organization Name
- Organization Descriptor
- GPE: country, city, state/province
- Work of Art.

5 Conversion

Since we launched the routine tectogrammatical annotation of PEDT, we worked with automatically pre-generated tectogrammatical trees, which were obtained by a conversion of the original constituency trees into the FGD-based analytical trees and subsequently from the analytical trees into tectogrammatical trees. The conversion tools were recently refined and integrated into a complex English-to-Czech machine-translation system called **TectoMT** [18]. The system consists of a long sequence of processing modules **(blocks)**, which perform small partial tasks. First, English tectogrammatical trees are generated from the English text input. Then the English tectogrammatical trees are transferred into Czech tectogrammatical trees. Czech analytical trees are created from the Czech tectogrammatical trees. Finally, the Czech text is created from the analytical trees.

For the automatic pre-generation of English tectogrammatical trees we used the manually created constituency trees of WSJ-PTB converted into a PML format as input for the first sequence of blocks, by which we obtained automatically generated analytical trees.[5] These blocks:

- lemmatize the word forms
- mark the head node (using a set of heuristic rules)
- build temporary m-trees containing morphological information (to be merged with a-trees later)
- convert constituency trees into a-trees

[5]Some of the blocks used in the MT tasks were left out when building tectogrammatical trees for manual annotation.

- apply some heuristic rules to fix apposition constructions
- apply other heuristic rules for reattaching incorrectly positioned nodes
- unify the way in which multiword prepositions (such as *because of*) and subordinating conjunctions (such as *provided that*) are treated.
- assign analytical functions (labels) if necessary for a correct treatment of paratactic constructions.

Next (much bigger) chain of blocks build tectogrammatical trees upon the analytical trees. The procedure is the following:

- Mark a-nodes which represent auxiliary words.
- Build t-trees. Each a-node cluster formed by an autosemantic node and possibly several associated auxiliary nodes is 'collapsed' into a single t-node. T-tree dependency edges are derived from a-tree edges connecting the a-node clusters.
- Distinguish coordination members from shared modifiers.
- Modify t-lemmas when necessary, insert t-lemma substitutes for selected nodes.
- Assign functors necessary for proper treatment of coordination and apposition constructions and fix the "coordination member" attributes.
- Distribute shared auxiliary words in coordination constructions.
- Mark t-nodes which are roots of t-subtrees corresponding to finite verb clauses.
- Mark passive verb clauses.
- Assign functors in selected cases (rule based).
- Assign functors by a statistically based procedure consisting of several blocks.
- Mark t-nodes corresponding to infinitive verbs.
- Mark t-nodes which are roots of t-subtrees corresponding to relative clauses or direct speech.
- Mark t-nodes which are roots of parenthetic t-subtrees.
- Fill in or correct several internal attributes of the nodes.
- Insert a reference Czech (manual) translation of the sentence.
- Assign valency frames.
- Recompute deep ordering of the nodes.
- Strip some attributes which are no longer useful when the procedure is finished.

Apart from the original TectoMT blocks, a statistical functor assigner (a recent component of a tectogrammatical parser, [19]) was employed to increase the accuracy of the automatic functor pre-assignment (it is already mentioned in the above list of blocks). A preliminary measurement (the trees pre-generated with and without the assigner compared respectively with the same trees which had been manually annotated before) proved a significant improvement on the WSJ-PTB data. The trees generated without the assigner achieved a 57.6 % functor agreement with the reference manual annotation. The introduction of the assigner raised the agreement to 77.3 %. That is quite good because the best interannotator agreement ever achieved was 85.7 %.

6 Rule-based pre-annotation

A significant improvement of the pregenerated tectogrammatical trees was brought by the flat NP annotation [15], which we integrated into the WSJ-PTB data fed to TectoMT. To increase the consistency and to speed up the annotation even more, we decided to improve the trees obtained from TectoMT by hand-written rules. These rules were designed to apply to selected recurrent structures, which were often impossible to detect by morpho-syntactic criteria, being conditioned rather lexically or even stylistically. When creating the rules for automatical pre-annotation, the constituency trees of WSJ-PTB were first browsed with Netgraph [20] and informally described along with the tectogrammatical subtrees desired as output. These informal descriptions were rewritten into perl scripts.

All our hand-written rules for automatic pre-annotation of WSJ-PTB are designed as "Find a specified constituency structure, locate the corresponding tectogrammatical structure and correct it". To create these rules, we used the following features:

- WSJ-PTB terminal, nonterminal and function tags
- WSJ-PTB structure
- lemmatization
- text strings (lists of words)
- BBN entity tags

We are including a few examples of the rules here.

Phrases of the type "$600 a share"

We are looking for an NP phrase (node A) with the function tag ADV and an NP or QP phrase (node B) to the left. Node A has exactly two childnodes (both terminal), the left one having the wordform "a" and the tag "DT". In case of a match we identify the t-subtrees created from the constituency structures rooted at the nodes A and B (let's call them TSA and TSB). Then we hang TSA under TSB and assign the functor REG to the root node of TSA.

This rule has 1701 hits in the corpus. See figures 1 and 2 for the constituency and for the resulting tectogrammatical structures.

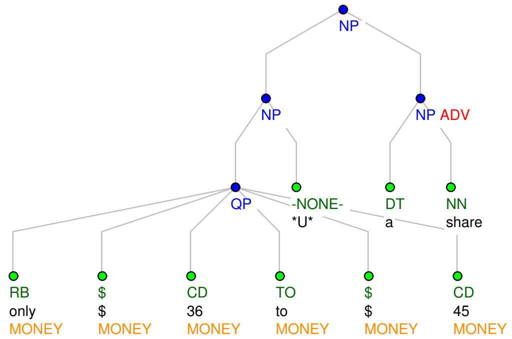


Figure 1: Example of a constituency structure of a phrase of the type "$600 a share"

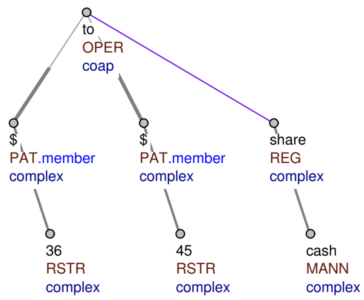


Figure 2: Example of a tectogrammatical structure of a phrase of the type "$600 a share"

Mixed Numbers

Whenever we found a mixed number (something like 3 2/7) in the form of two terminal nodes with the tag CD, we transformed it into a tectogrammatical structure shown in Figure 3. There are 1351 mixed numbers in the corpus.

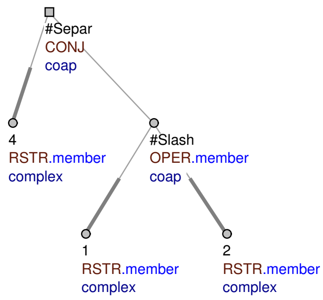


Figure 3: Example of a tectogrammatical structure of a mixed number

Phrases of the type "Boston, Massachusetts"

We are looking for an NP or an NML nonterminal with the phrase attribute value NAC and with the function LOC as its child (let's call it Node A). There has to be either an NP or an NML nonterminal or a noun (a terminal with a tag whose first two letters are NN) among the right siblings of the Node A – let's call it Node B. Node A has three or four childnodes. The second one is comma or left round bracket (a terminal node). If there is the fourth childnode, it has to be a comma or a right round bracket (again a terminal node). If the fourth childnode is not present and the leftmost node of the Node B subtree satisfies the requirements, we can consider it to be the fourth child. The third childnode has to satisfy one of these three demands:

- It is an NP or an NML nonterminal and all the terminals in its subtree have the BBN-tag GPE:STATE_PROVINCE .
- It is a noun with the BBN-tag GPE:STATE_PROVINCE.
- It is a roman number (terminal node) with no BBN-tag.

The tectogrammatical counterpart of this structure is as follows. At first we identify the t-nodes which are roots of structures created from the subtrees rooted in the first and the third childnode of Node A (let's call them TR1 and TR3). Now we hang TR3 under TR1 and assign functors. TR1 should be LOC and TR3 gets the functor PAR. We also set the attribute is_parenthesis to 1 for each descendant of TR3 including the node TR3 itself. The second (and possibly the fourth) child of Node A is auxilliary and the corresponding a-node has to be properly referenced from the TR3 node. We also have to ensure that those auxilliaries do not exist as independent t-nodes and that they are not referenced from any other t-node.

There are 239 occurences of the described constituency structure in the corpus. See figures 4 and 5 for examples of the described structures. This script can with minor modifications be applied for structures consisting of person nouns and their political affiliations (e.g., *Leon Panetta, democrat*).

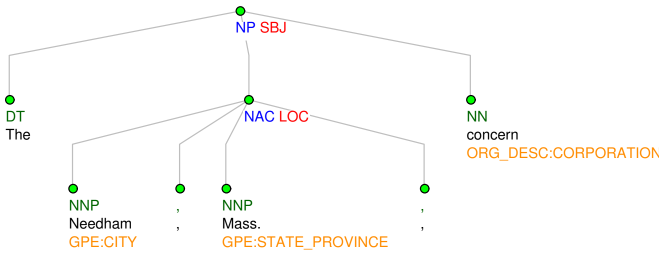


Figure 4: Example of a constituency structure of a phrase of the type "Boston, Massachusetts"

Figure 5: Example of a tectogrammatical structure of a phrase of the type "Boston, Massachusetts"

From August 2008 to November 2008 we created more than 60 rules (some of them became obsolete). The complete set of scripts was tested on one reference section (296 sentences, 7694 words). As a result we registered 1237 changes. We were measuring the agreement with manually annotated data, and we achieved an approx. 4 % improvement in functors and 6 % in referencing auxilliaries, which is not a really substantial improvement. The agreement on other attributes was more or less identical. However, in this case the quantity is not the only goal. Better consistency of the data is important as well. Besides applying annotation templates to structures relatively uninteresting from the linguistic point of view, such as mixed numbers, our rules annotated a number of complex and less frequent linguistically relevant phenomena throughout the corpus. Sometimes the given structures could not be processed completely, but the applied rules saved the annotators at least a part of manual work. The overall effect of these measures on the annotation procedure would be too difficult to quantify, though. The outcomes of some rules were left for manual processing within the expert annotation (10), which has positive effect on the annotation consistency as well.

7 Manual Annotation

The initial tectogrammatical annotations of English data (WSJ-PTB) date back to 2002 [21]. The tectogrammatical trees were built above analytical WSJ-PTB trees obtained by an automatic conversion from the original PTB bracketing into the format used by PDT 1.0 [22]. The automatically converted and generated data as well as this tentative manual tectogrammatical annotation were published along with parsed Czech parallel translations of WSJ-PTB as the **Prague Czech-English Dependency Treebank 1.0** (PCEDT 1.0, [23]). The PCEDT 1.0 with its 500 manually annotated tectogrammatical trees constituted the starting point for the efforts taken up 2004.

Due to substantial format changes of the "mother treebank", the Prague Dependency Treebank, before its second LDC release [4] in 2006, the massive annotation of English data was postponed until the language-independent features of the new annotation scheme [24] would stabilize. In the meantime we concentrated on the conversion of PropBank [12] into an FGD-compliant valency lexicon. In early 2006 we were able to convert the constituency trees into tectogrammatical trees with some of the modules which later became part of TectoMT. We were refining the initial version of the annotation manual.

Four annotators started the manual annotation in late 2006. During 2007, several more annotators were trained. At the moment we have four annotators working regularly, the rest being mostly in training, some having left the project, and some being on maternal leave. The interannotator agreement was measured approx. once a month in 2006 and early 2007. It has not been measured since March 2008, mainly because of the slow annotation pace in 2007, annotator fluctuation, and, since mid-2008, due to the intensive work on consistency controls, which all skilled annotators have been kept busy with.

The annotators work mostly off-line but send and retrieve the data via an SVN system. The data index as well as the work-progress stats are provided with a user-friendly web interface. The annotators currently correct the data produced in 2006 and 2007 by running the consistency-checking scripts upon each file and correcting the detected errors. The annotators are also asked to run the checks and correct the errors before submitting new files. A log of changes in the data is generated every month. It calculates uncorrected detected errors and the ratio of the amount of data vs. the amount of changes. Deviations from the average are examined and random samples are manually re-checked.

8 Consistency Controls

After the annotated data exceeded 12,000 trees (almost 25% of WSJ-PTB), we introduced consistency controls. Most of the scripts we use have been adopted from the Czech PDT-team [25] and modified whenever necessary. We added a few new, English-specific control scripts, and we reuse some of our pre-annotation scripts. The controls check mainly:

- **Paratactic structures**
  - Only a node of the appropriate type and with an acceptable functor is the root of a paratactic construction.
  - Each root of a paratactic construction has at least two descendants which are coordination members.
  - Only permitted combinations of functors occur in coordinated nodes.

- **References from t-nodes to a-nodes (content-word and auxiliary-word references)**
  - All a-nodes which represent alphanumerical tokens are referred to from the t-layer (except punctuation).
  - No a-node is referred to as a content-word from two non-generated t-nodes.
  - All t-nodes except nodes with t-lemma substitutes refer to a content word node at the a-layer.
  - A t-node, whose corresponding content-word reference at the a-layer is a noun in plural, may not refer to an a-node that represents the indefinite article.
  - T-nodes representing punctuation regarded as a content word (e.g., punctuation in asyndetic paratactic constructions) must not be represented as generated nodes.
- Tree structure
  - The effective root of the tree is either the main predicate (which might be an artificially inserted one) or the governing node of a noun group.
  - Nodes representing foreign words comply with all rules.
  - Nodes representing phrasemes comply with all rules.
  - T-nodes with t-lemma substitutes which are used for specific syntactic constructions (e.g. `#AsMuch|#Equal|#Total`) are never terminal nodes (leaves).
  - The technical root has only one descendant.
  - Each t-node has been assigned a functor.
- Valency
  - Each occurrence of a verb except *to be* and *to have* is assigned a valency frame from the lexicon.
  - The valency frame is complete according to the valency lexicon.
  - The valency frame assigned to a verb occurrence must exist in the lexicon (frames can be altered during the lexicon edits).
  - A copied verb has the same valency frame as the original.
  - All controls are dismissed when the verb node contains an annotator's comment regarding the lexicon.

This list presents only selected controls. There are approx. 80 control scripts at the moment. Their amount is slowly but constantly growing. The annotator's comments serve as issues for new pre-annotation scripts, TectoMT improvements or control scripts. The comments regarding the valency lexicon are collected monthly in form of a log file with the examples and sentence identification, and they are e-mailed to the editor-in-chief of the lexicon. Besides, we are experimenting with a string-based consistency control of the tree structure and functor assignment. The data is searched for subtrees consisting of matching textual strings. Differences in the respective annotation resolutions for textual sequences are reported. See a sample of the first tentative inconsistency survey below:

```
previous month
  [month]([previous,RSTR]) 3
  [month]([previous,TWHEN]) 1
rate increase
  [increase]([rate,PAT]) 1
  [increase]([rate,ACT]) 1
size of the increase
  [size]([increase,ACT|of,the]) 1
  [size]([increase,APP|of,the]) 1
so far
  [far]([so,EXT]) 5
  [far]([so,MANN]) 1
```

Some of these reports help us uncover inconsistencies systematically made by the automatic pre-annotation and can be fixed. Many of them have to be manually checked by the annotators.

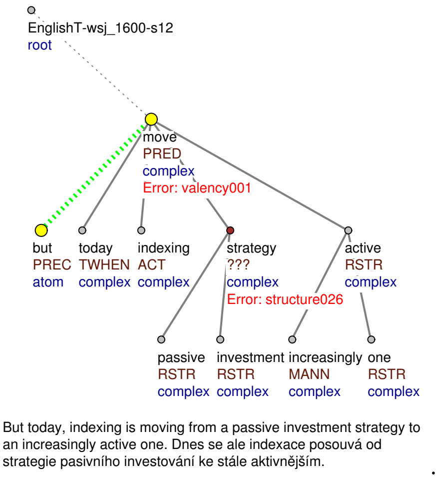
9 PEDT 1.0

The first 12,440 manually annotated and checked trees were released under the title PEDT 1.0. The CD includes the documentation along with relevant publications, the current version of the valency lexicon Engvallex (which is yet still being subject to revisions), and the ready-to-install package of TREd, the tree editor.

10 Discussion

The current annotation practice yields trees quite consistent in tree structure, some financial-speak specific fixed phrases, structured text like addressess and lists, and verbal valency. However, the annotation still remains inconsistent in functor assignment in adjectival and nominal phrases. We decreased this inconsistency by resigning on semantic labeling within named entities (all nodes in the subtree get the new functor NE - *Named Entity*), but we do not find this solution satisfactory, and we are going to introduce a systematic solution of noun valency in later versions of PEDT. We have tentatively merged the NomBank [13] annotation with the PEDT data and are going to explore its benefits for an FGD-based annotation. While PropBank was driven by theoretical approaches quite similar to FGD, the NomBank approach might prove difficult to adopt. No conclusions can be drawn yet as we are just at the very start of the process.

In the next future we are going to continue improving the automatic pre-annotation by detecting problematic phrases and linguistic phenomena while the annotators are supposed to finish the consistency controls within a few months. As soon as the data has been annotated with the complete annotation, we will focus on the so-called **expert annotation**. This is annotation of selected structures across all corpus sections by one or a few 'expert' annotators. This procedure is meant for the annotation of particularly difficult or interesting phenomena. It is mainly supposed to further increase the consistency of the annotation. Besides, it is meant to provide material for linguistic research. 10 shows a TREd window with a highlighted expert-annotation task.



11 Conclusion

PEDT has been built to present the Prague Dependency Treebank-like annotation scheme to the global expert audience. The documents were chosen because of their original manual annotation and due to being a sort of a reference corpus in the NLP community, despite all linguistic objections that could be raised on how much the English used in American business press reflects the patterns of English in general. The annotation procedure has been improved, and so have the control mechanisms. Approximately 1/4 of WSJ-PTB has been annotated at the moment.

12 Acknowledgements

This paper was supported by the Czech Science Foundation (GA CR) project Nr. GA CR 405/06/0589.

References

[1] U. Römer, A neo-firthian approach to academic writing: Uncovering local patterns and local meanings in the discourse of linguistics, oral presentation at the Conference of the American Association for (Applied) Corpus Lingusitics, Brigham Young University, Provo, Utah, USA, 2008.

[2] S. Oepen, Beyond the science of the wall street journal, presentation slides at the Unified Linguistic Annotation (ULA) Workshop, TLT 2007, Bergen, 2007, URL: http://tlt07.uib.no/ulaslides/stephan-ula.pdf, quoted 2008-12-29.

[3] D. Gildea, Corpus variation and parser performance, in *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 167–202, 2001.

[4] J. Hajič *et al.*, *Prague Dependency Treebank 2.0* (Linguistic Data Consortium, 2006).

[5] S. Cinková *et al.*, UFAL MFF UK Report No. 35, 2006 (unpublished).

[6] J. Hajič *et al.*, PDTSL: An annotated resource for speech reconstruction, in *Proceedings of the 2008 IEEE Workshop on Spoken Language Technology*, 2008.

[7] J. Bradley, O. Mival, and D. Benyon, A novel architecture for designing by wizard of oz, in *Proceedings of CREATE08*, 2008.

[8] P. Sgall, E. Hajičová, and J. Panevová, *The Meaning of the Sentence in Its Semantic and Pragmatic Aspects* (Dordrecht:Reidel Publishing Company and Prague:Academia, 1986).

[9] M. P. Marcus, B. Santorini, M. A. Marcinkiewicz, and A. Taylor, *Treebank III* (Linguistic Data Consortium, 1999).

[10] M. P. Marcus, B. Santorini, and M. A. Marcinkiewicz, *Treebank II* (Linguistic Data Consortium).

[11] A. Bies, M. Ferguson, K. Katz, , and R. MacIntyre, Bracketing guidelines for treebank ii style penn treebank project, 1995.

[12] M. Palmer, P. Kingsbury, O. Babko-Malaya, S. Cotton, and B. Snyder, *Proposition Bank I* (Linguistic Data Consortium, 2004).

[13] A. Meyers, R. Reeves, and C. Macleod, *NomBank v 1.0* (Linguistic Data Consortium, 2008).

[14] D. Vadas and J. R. Curran, Adding noun phrase structure to the penn treebank, in *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2007.

[15] School of Information Technologies, University of Sydney Report No., , 2007 (unpublished).

[16] R. Weischedel and A. Brunstein, *BBN Pronoun Coreference and Entity Type Corpus* (Linguistic Data Consortium, 2005).

[17] S. Kulick *et al.*, Integrated annotation for biomedical information extraction, in *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, 2004.

[18] Z. Žabokrtský, J. Ptáček, and P. Pajas, Tectomt: Highly modular mt system with tectogrammatics used as transfer layer, in *Proceedings of WMT'08*, 2008.

[19] V. Klimeš, Transformation-based tectogrammatical dependency analysis of english, in *Proceedings of the 10th International Conference on Text, Speech and Dialogue*, 2007.

[20] J. Mírovský, PDT 2.0 requirements on a query language, in *ACL 2008 Proceedings of the 46th Annual Meeting of the Association of Computational Linguistics: Human Language Technologies*, pp. 37–45, Association for Computational Linguistics, 2008.

[21] I. Kučerová and Z. Žabokrtský, p. 77 (2002).

[22] J. Hajič, E. Hajičová, P. Pajas, J. Panevová, and P. Sgall, *Prague Dependency Treebank 1.0* (Linguistic Data Consortium, 2001).

[23] J. Cuřín *et al.Prague Czech-English Dependency Treebank Version 1.0* No. LDC2004T25 (Linguistic Data Consortium (LDC), University of Pennsylvania, 2004).

[24] P. Pajas and J. Štěpánek, XML-based representation of multi-layered annotation in the PDT 2.0, in *Proceedings of the LREC Workshop on Merging and Layering Linguistic Information (LREC 2006)*, edited by R. E. Hinrichs, N. Ide, M. Palmer, and J. Pustejovsky, pp. 40–47, Paris, France, 2006.

[25] J. Štěpánek, Prague Bulletin of Mathematical Linguistics , 23 (2006).