Treebank Annotator for multiple formats and conventions

Cătălina Mărănduc, Florinel Hociung, Victoria Bobicev

Abstract

The UAIC-RoDepTreebank becomes an important balanced corpus, with increasing dimensions, which is intended to be used for multiple applications. For this purpose, it will be available in several formats, classical syntactic, semantic, Universal Dependencies and PROIEL. Dependency trees are in XML format, and for viewing it, we use frameworks that allow manual annotation or automatic annotation correction. As the interface used so far only allows working with the classic syntactic format, we present here a new, multifunctional interface that allows the input of any tree format.

Keywords: natural language processing, dependency treebank, manual annotation, automate supervised annotation.

1 Introduction

Corpora are very important resources for Natural Language Processing, in the absence of which there is no possibility of training the programs and of making searches. The degree of computerization of a language depends on the existence of large corpora with many types of consistently annotated information.

UAIC-RoDepTreebank **[4, 5]** has complete morphological analysis **[9]** and syntactic classic annotation, in Dependency Grammar **[6]**, entirely supervised. The classical syntactic format has now 19,825 sentences and 389,357 tokens, punctuation included. The specificity of our corpus is the tendency to cover all the styles of the language, being more interested in the non-standard language. We already have 2,575 phrases on chat communication and 6,882 phrases in old Romanian, written from the

seventeenth to the nineteenth century, although it is difficult to create and train tools to process these types of texts.

In order to make the corpus accessible for multiple applications and as many users as possible, we decided to transpose it into an international format, Universal Dependencies, (UD), maintained by a group that brings together over 30 treebanks **[8]**. Currently, the UD format of our treebank has 4,600 sentences and 101,568 tokens, and other sub-corpora are in the course of transformation.

Another purpose of our work is to add new complex annotations to the corpus, so we started creating another format, in which the syntactic trees are transformed into semantic dependency trees. This format has now 4,405 sentences with 72,607 tokens, and other sub-corpora are in course of transformation.

As the trees are in XML format, (or CONLLU for the UD format) **[1]**, annotators and users have the necessity to visualize, and eventually modify them using a framework. The one used so far has been created according to the classic format of the treebank and only allows uploading sub-corpora in this format.

We present here a new multifunctional and flexible framework used currently for dealing with the new (UD and semantic) formats of our treebank. The tool has been created as a dissertation project **[2]**, being able to perform all the new tasks of the treebank that the old annotator could not fulfill. Each function has been experienced by the user up to its perfect operation, and the drop-down lists have been aligned with all the new changes of the tag lists.

2 Related Work

We present shortly four other similar programs. The old *Treeannotator* **[7]** is an application built on the Java platform. It allows loading only a particular format. If there are inconsistencies in the corpus loaded, the tool brings the corpus to its standard form. It restores the numbering of words in sentences, puts the items of XML in the same order, checks the validity of the XML, and shows where is the mistake that makes the XML invalid, marks the graphs that are complete trees in the list. It can draw the sentence graph in a linear manner or in form of trees.

The proposed interface on the UD site is *Dependency Grammar Annotator* (DGA) **[10]**. The format that can be loaded is UD, and the graph is linear.

Easy Tree is an application running in the browser, created within the CQL summer program, edition 2015 **[3]**. It allows the editing of small sentences. The display is in tree form.

Grammar Scope is an application on the Stanford Parser \ Stanford CoreNLP platform **[11]**. It offers very advanced functions of editing, creating and parsing linguistic resources, syntactically and semantically annotated, in a different format than ours.

3 Treebank Annotator Settings

Although *Grammar Scope* is a very good tool, it cannot be used by us because it was made for different format. Although these tools are language-independent, some specificities of a language make the authors to choose specific annotation conventions. We can see that each corpus creates specific automatic and manual annotation tools that are presented on its site.

The new annotator of the UAIC-RoDepTb corpus performs all functions of the Treeannotator described above, and adds many more. It works with the folder called ”Configurations” that contains working files with formatting features with the lists of all possible values. The interface has a drop down list for each feature. We can add a new file in the folder ”Configurations”, to introduce another format, or a new feature in a configuration, or a new value allowed in the list of a feature, or a new list of allowed values.

In order to open a sub-corpus (which has around 1,000 sentences) it is necessary to choose first a configuration, i.e. to specify in which format the corpus to be opened is.

Concerning the XML and CONLLU format, the application allows uploading of one of them and saving in the other, i.e. it can function as convertor. The application also allows uploading simultaneously more sentences in different formats and comparing two of them. (see Fig.1).

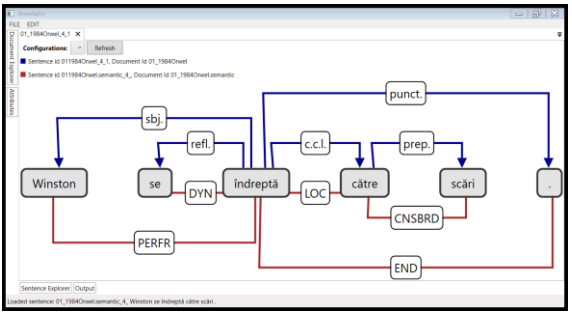

Figure 1. Comparison between the syntactic and semantic format. (En: Winston made for the stairs.)

4 Other facilities offered by Treebank Annotator

There are three possibilities of graphs visualization: linear, tree or oblique. The linear order is preferred because it is the only one that allows the tracing of dependencies simultaneously with the order of words in the text. It would be desirable for the tree view to allow the order of the words to be traced, as is the case in the old Treeannotator.

The new application allows uploading of a treebank or creating a new dependency tree, introducing a text to be manually annotated. The tool creates an XML for the inserted text, with all the features of the format chosen by default, and the user selects a value for each feature, from its drop-down list.

The tool allows not only changing the edges and the tags of all features, but also adding or removing words. It is an important function, because some multi word expressions (MWEs) are not correctly interpreted by the tokenizer. A group of words may be misinterpreted as MWE, so the words need to be separated, or multiple words have unitary meaning, forming a MWE, and need to be united.

When we are adding or removing a graph node, the application automatically changes both word and head ID-s, so that the rest of the graph does not change. The tool also allows changing the word order in the sentence, without changing the dependency structure. When a word is added, it is placed at the end of the sentence and then, by changing the order of words, it is brought to the desired location.

When a word is selected, all its features existent in XML appear in the ”Attributes” field, each with its dropdown list. The existence of dropdown lists allows selecting a tag by typing only the first letters and eliminates the possibility of mistyping. The program does not allow writing a tag that is not in the chosen Configuration list.

5 Conclusion and future work

Specific tools have to be built for each treebank corpus, respecting the distinctive rules and conventions, even if they are intended to be language independent. In the paper, we have presented a new tool created for the needs of UAIC-RoDepTb.

Following the model of *Grammar Scope* cited above, a syntactic or/and semantic parser will be integrated in the Treebank Annotator. Currently, the tool is used with success for the supervision of automatic transformation of the syntactic classic format in the UD or in the semantic one. In the future, tools for the automatic transposition of the classical syntactic format into UD and into semantic ones will be integrated as convertors in the Treebank Annotator.

Another project of our NLP group is to create a Pattern Dictionary of Romanian Verbs (PDRoV), which will be accessible online. It will include contemporary, archaic, regional, familiar verbs. The Patterns are structures of mandatory or facultative syntactic dependencies, with their semantic possible realizations.

For each verb described, examples of all the patterns will be included in the dependency treebank, in all of its three formats. The corpus in the three formats will form a database linked to the PDRoV site.

The Treebank Annotator will be a very important component of this project, permitting the user to visualize the tree form of each example for each pattern described, in the format chosen, classical, UD, or semantic, displayed in linear, tree or diagonal view.

This interface is important because it respects all the conventions of annotation of our treebank and also permits us to make changes in its Configurations, to meet new requirements, format changes, or new formats introduced to synchronize with similar international projects.

References

[1] J. Hall. J. Nilsson. *CoNLL-X Shared Task: Multi-lingual Dependency Parsing*. MSI report 06060. Växjö University, School of Mathematics and Systems Engineering. (2007).

[2] F. Hociung. *Treebank Annotator*. J dissertation, Faculty of Computer Science, Alexandru Ioan Cuza University, Iaşi (2016).

[3] A. Little, S. Tratz. *EasyTree. A Graphical Tool for Dependency Tree Annotation*., In Proceedings of LREC (2016), pp. 2343-2347.

[4] C. Mărănduc, C. A. Perez. *A Romanian Dependency Treebank*. International Journal of Computational Linguistics and Applications, Vol. 6, No. 2 - July–December. (2015) pp. 25–40.

[5] C. Mărănduc, C. A. Perez. *A Resource for the Written Romanian: the UAIC Dependency Treebank.* Proc. of ConsILR, Mălini. (2016), pp. 79-90.

[6] I. A. Mel'Čuk. *Dependency Syntax: Theory and Practice*, Buffalo, Suny Press. (1987).

[7] Al. Moruz. *Development of an FDG (Functional Dependency Grammar) annotator for Romanian*, dissertation, Faculty of Computer Science, Alexandru Ioan Cuza University, Iaşi. (2008).

[8] R. Rosa, J. Mašek, D. Mareček, M. Popel, D. Zeman, Z. Žabokrtský. *HamleDT 2.0: Thirty Dependency Treebanks Stanfordized*., in Proc. of LREC. (2014).

[9] R. Simionescu,: *Hybrid POS tagger*. The Workshop on Language Resources and Tools in Industrial Applications, Eurolan 2011 summer school (2011).

[10] http://phobos.ro/roric/DGA/dga.html

[11] http://grammarscope.sourceforge.net/

Cătălina Mărănduc[1,2], Florinel Hociung[2], Victoria Bobicev[3]

[1]Academic institute of Linguistics Iorgu Iordan – Al. Rosetty Bucharest, Romania
E-mail: catalinamaranduc@gmail.com

[2]Faculty of Computer Science, Al. I. Cuza University, Iaşi, Romania
E-mail: florinel.hociung@info.uaic.ro

[3]Technical University of Moldova, Chișinău, Republic of Moldova
E-mail: victoria.bobicev@ia.utm.md