

VerbNet Workbench

Indrek Jentson

University of Tartu

Tartu, Estonia

indrek.jentson@ut.ee

Abstract

In this paper a tool to manage a dataset for a VerbNet-like verb lexicon is presented. It was designed to allow users to create a verb lexicon for another language than English and at the same time use the same data structure as the English VerbNet. We take a look at the most relevant requirements of the software and will give an overview of the functionality achieved so far.

1 Introduction

In 2000 the verb lexicon for English was created by scientist from the University of Pennsylvania and named as VerbNet (Kipper et al., 2000). The following work has extended the content of the verb lexicon with many new verbs and verb classes (Kipper et al., 2006). Today in the English VerbNet version 3.2 there are 273 total main classes and 214 total subclasses with 6340 total verbs covered (Unified Verb Index, 2013).

Several works have shown that the VerbNet is very useful for NLP but till now a resource of this size and coverage exists only for English. There is no questions that similar verb lexicons for others languages are needed.

In recent work the question was asked - is it feasible to convert the English Verbnet into a similar verb lexicon for some other language and the following analysis for Estonian showed that in principle the class hierarchy, thematic roles with restrictions and semantic descriptions are reusable for such work (Jentson, 2013).

In order to start building a new verb lexicon for Estonian side-by-side with the English VerbNet the appropriate tool - VerbNet Workbench - was designed and implemented.

2 Requirements for the VerbNet Workbench

In order to understand exactly what kind of software is required to manage VerbNet data the most essential functional requirements (FR) were specified.

FR1. The system shall allow each user to choose a target language for the following work session.

FR1.1. The system shall allow an authenticated user to add a new language to the list of available languages.

FR2. The system must be completely compatible with the data structure of the English VerbNet.

FR2.1. The system shall allow to import the VerbNet data files for the selected language in XML format correspondent to XML schema `vn_schema-3.xsd` (VerbNet, 2013).

FR2.2. The system shall allow to export the VerbNet data files for the selected language in XML format with the XML scheme file consistent to the exported data.

FR3. The system shall allow an authenticated user to enter the following information in the context of the selected language:

- 1) the general data for a verb class together with a reference to the corresponding verb class in the English VerbNet;
- 2) the members of the verb class together with the references to the other language resources (for example the WordNet, the FrameNet etc);
- 3) the thematic roles with the selection restrictions for the arguments of the verb class;
- 4) the syntactic frames of the verb class, each containing an example, a syntactic template with the syntactic restrictions and a semantic description;

5) the subclasses of the verb class, they are with the same structure as the main class.

FR3.1. The system shall allow an authenticated user to insert new selection restrictions with the descriptions for the thematic roles.

FR3.2. The system shall allow an authenticated user to insert new syntactic restrictions with the descriptions.

FR3.3. The system shall allow an authenticated user to insert new predicates with the descriptions for the semantic descriptions.

FR4. The system shall allow an authenticated user to reserve a verb class for his/her work and publish the changed data only after the work is marked completed.

FR4.1. The system shall show to an authenticated user the list of verb classes reserved by that user.

FR4.2. The system shall prevent a user from reserving some verb class that is already reserved.

FR5. The system shall maintain all versions of the records for every verb class.

FR6. The system shall allow each user to search the verbs from the lexicon and list all the references to those verb classes where the verb is in the list of members.

In the process of designing and implementing the VerbNet Workbench software all those requirements were taken into account. From the non-functional requirements we highlight only one - the targeted system must be web-based in order to ensure its availability to all interested parties and to allow many linguists to work together on the VerbNet data.

3 Results: overview of functionality

In order to build the VerbNet Workbench we used the programming language Python 2.7 (2013) and the web application framework Django (Django Project, 2013). For data storage the database management system PostgreSQL 9.2 (2013) is used, but it is possible to use any relational database system supported by Django.

The UML class diagram of the necessary data model is presented on Figure 1. The main data object on the diagram is class *VNClass* the purpose of which is to hold data for the verb classes in the context of the chosen language. The list of verbs (class *Member*), thematic roles (class *ThematicRole*) and syntactic frames (class *Frame*) belongs to each verb class. For each syntactic frame, there is a data structure to describe the

template (class *Syntax* etc) and the meaning (class *Semantics* etc) of the sentence.

In Table 1 there is a short overview of the functionality which is available to the users. Access to that functionality is divided between three user roles. The role ‘User’ belongs to any unauthenticated user who wants to use the prepared data. An authenticated user gets the role ‘Contributor’ and can additionally do everything that ‘User’ can do and the user with the role ‘Administrator’ has rights to do everything.

Actor	Use Case
User	Choose a language
User	Browse a class hierarchy
User	View attributes of the verb class
User	Search for a verb class by given verb
User	Export the VerbNet dataset for chosen language (XML-files)
Contributor	Authenticate the user
Contributor	Define a new language
Contributor	Create a new verb class or subclass
Contributor	Enter information for attributes of the verb class
Contributor	Add the members to the verb class
Contributor	Change data of the verb class
Administrator	Import the VerbNet dataset for chosen language (XML-files)
Administrator	Manage the users

Table 1. The overview of realized Use Cases

The usual scenario for entering new information to the database includes activities like choosing some verb class from the English VerbNet, entering an appropriate name for this verb class in the chosen working language, finding the class members (this can be started by translating the verbs in the same verb class of the

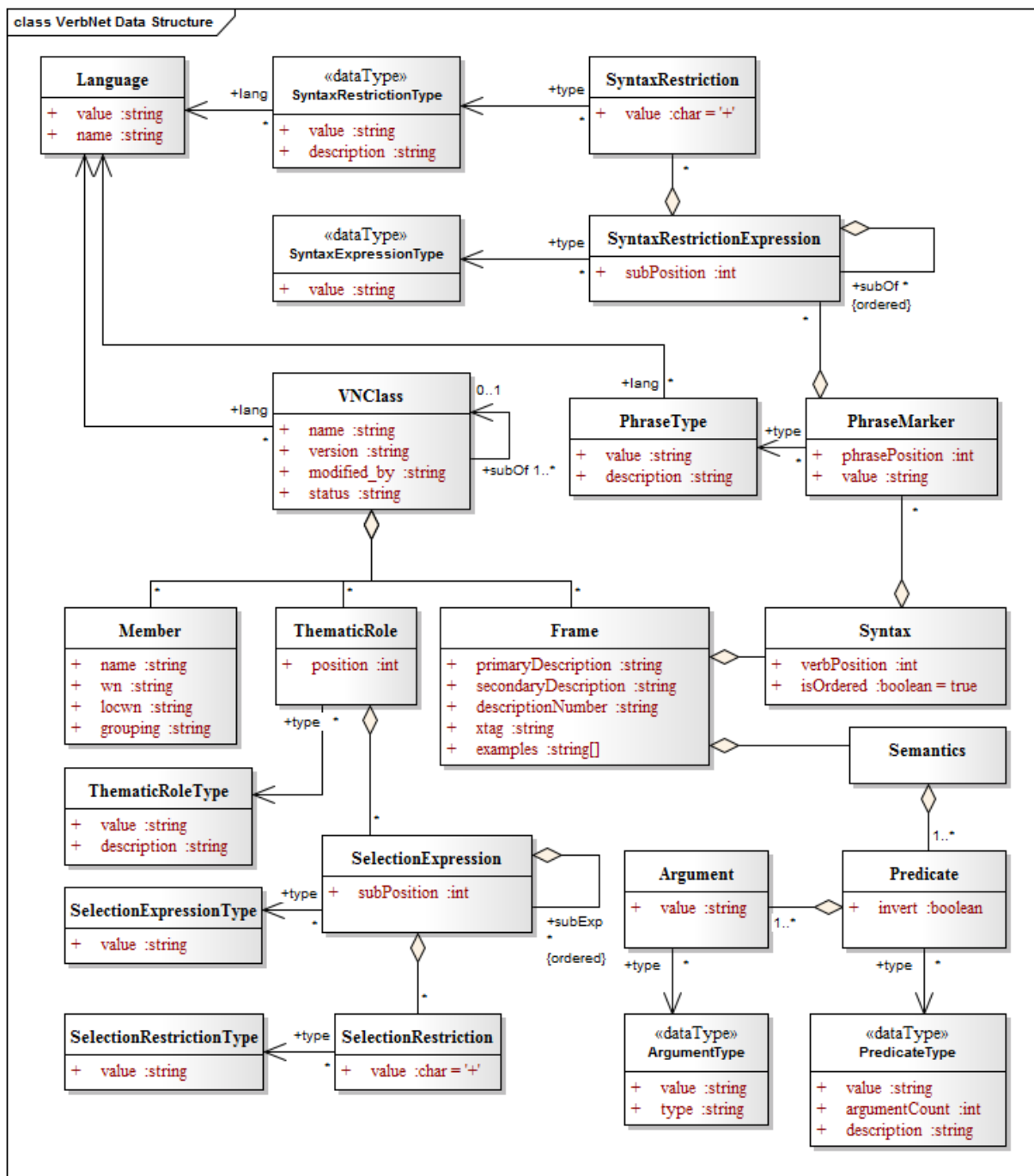


Figure 1. The data model for the VerbNet Workbench

English VerbNet) and defining the syntactic frames. Defining all suitable syntactic templates with the syntactic restrictions for the selected verb class is indeed the most challenging work in this process because there lay the main differences when we are looking from the point of view of another language.

This basic functionality allows linguists to start collecting information about the verbs for many different languages so that the data structure of the gathered information is compatible

with the English VerbNet and the verb classes from one language are comparable to the verb classes from some other language.

4 Discussion: present and future challenges

The first user experience has shown that the tool allows data to be managed in such a way that all necessary information can be entered by the contributors and the users can browse, search and

download data already collected. However, it is also observed that some advanced features would be helpful for data entry, enabling the necessary data type values to be selected and the amount of manual input reduced.

Referencing from the submitted data to other resources for the same language is currently implemented only on the description level. Functionality, which allows opening and viewing referenced resources such as Wordnet or Framenet, is depending on availability and access methods of each specific resource and the general approach is therefore complicated to implement.

5 Conclusion

It can be concluded that the main use cases with basic functionality are indeed realized, but more work is necessary in order to increase usability and user comfort. It is also planned to enable a localization of the application in order to provide the users with the possibility to use a preferred language for the user interface. A separate issue is drafting the user manual to give substantive guidelines for categorizing verbs and to explain the basic principles and the rules about compiling a verb class dataset.

We hope that the availability of the VerbNet Workbench will propitiate work on verb semantics and give the possibility to create a useful language resource for natural language processing in many languages.

Reference

- Indrek Jentson. 2013. *The Feasibility of Estonian VerbNet*. Estonian Papers in Applied Linguistics 9 (2013): 75-83.
- Karin Kipper, Hoa Trang Dang and Martha Palmer. 2000. *Class-based construction of a verb lexicon*. Proceedings of the National Conference on Artificial Intelligence. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999, 2000, 691–696.
- Karin Kipper, Anna Korhonen, Neville Ryant and Martha Palmer. 2006. *Extending VerbNet with novel verb classes*. Proceedings of 5th international conference on Language Resources and Evaluation (2006), No. 2.2.
- Django Project,
<https://www.djangoproject.com/>
(15.11.2013).
- PostgreSQL, <http://www.postgresql.org/>
(15.11.2013).

Python Programming Language,
<http://www.python.org/> (15.11.2013).

Unified Verb Index,
<http://verbs.colorado.edu/verb-index/> (15.11.2013).

VerbNet v3.2,
<http://verbs.colorado.edu/verb-index/vn/verbnet-3.2.tar.gz>
(15.11.2013)