A Spinning Wheel for YARN: User Interface for a Crowdsourced Thesaurus

Pavel Braslavski
Ural Federal University
Kontur Labs
pbras@yandex.ru

Dmitry Ustalov
Ural Federal University
IMM UB RAS
dau@imm.uran.ru

Mikhail Mukhin
Ural Federal University
mfly@sky.ru

Abstract

YARN (Yet Another RussNet) project started in 2013 aims at creating a large open thesaurus for Russian using crowdsourcing. This paper describes synset assembly interface developed within the project — motivation behind it, design, usage scenarios, implementation details, and first experimental results.

1 Introduction

Creation of linguistic resources and annotations using crowdsourcing and gamification is becoming a common practice. Untrained workers contribute to development of thesauri, dictionaries, translation memories, and corpora annotations. Crowdsourcing can employ both paid workers, e.g. on Amazon Mechanical Turk (AMT) platform[1] and volunteers as in case of Wiktionary[2] — a large wiki-style online multilingual dictionary.

The goal of the YARN (Yet Another RussNet) project[3] launched in 2013 is to create a large open thesaurus for Russian language using crowdsourcing (Braslavski et al., 2013). Despite the fact that there were several attempts to create a Russian Wordnet (Azarova et al., 2002; Balkova et al., 2004; Gelfenbein et al., 2003; Loukachevitch and Dobrov, 2002), there is no open resource of acceptable quality and coverage currently available. The choice of crowdsourcing is also advocated by successful projects that are being evolved by volunteers: Russian Wiktionary[4], corpus annotation project OpenCorpora[5] and a wiki for linguistic resources related to Russian NLPub[6].

Wordnets had been traditionally developed within small research teams. This approach maintains conceptual consistency and project manageability, facilitates informal exchange of ideas in a small group of contributors. However, this practice is hardly scalable and can potentially lead to a biased description of linguistic phenomena caused by the preferences of a close group of researchers. Crowdsourcing can possibly reduce costs, increase development pace, and make the results more robust, but puts additional demands on project management and tools, including user interface. Requirements for a crowdsourcing thesaurus development interface are as follows: 1) a low entry threshold for new users and a gradual learning curve; 2) no need for users to install additional software; 3) central data storage, collaborative work for several users in a competitive mode, and permission management; 4) change history tracking to protect data against vandalism.

Princeton WordNet editors had worked directly with lexicographer files stored in a version control system (Fellbaum, 1998). In later thesauri creation projects specialized tools were developed that featured more user-friendly interface, graphical representation of thesaurus relationships, centralized data storage, possibility of collaborative work, and data consistency checks. Examples of thesauri development tools are DEBVisDic (Horák et al., 2006), GernEdiT (Henrich and Hinrichs, 2010), as well as WordNetLoom (Piasecki et al., 2012) (see (Piasecki et al., 2012) for a brief overview of thesauri editing tools). Wiktionary and OmegaWiki[7] use MediaWiki engine and wiki markup to encode dictionary information.

In the preparatory stage of the project we considered adoption of the above mentioned tools. However, we estimated that the amount of work needed for adaptation of existing tools to YARN

[1] http://www.mturk.com/
[2] http://www.wiktionary.org/
[3] http://russianword.net/
[4] http://ru.wiktionary.org/
[5] http://opencorpora.org/
[6] http://nlpub.ru/
[7] http://www.omegawiki.org/

Proceedings of the Demonstrations at the 14th Conference of the European Chapter of the Association for Computational Linguistics, pages 101–104,
Gothenburg, Sweden, April 26-30 2014.

data formats and usage scenarios is quite costly and decided to develop a series of specialized tools.

The paper briefly describes YARN project and its noun synsets assembly interface in particular — motivation behind it, current state and appearance, usage scenarios, as well as results of a preliminary user study and future plans.

2 Project Outline

YARN is conceptually similar to Princeton Wordnet (Fellbaum, 1998) and its followers: it consists of synsets — groups of quasi-synonyms corresponding to a concept. Concepts are linked to each other, primarily via hierarchical hyponymic/hypernymic relationships. According to the project's outline, YARN contains nouns, verbs, and adjectives. We aim at splitting the process of thesaurus creation into smaller tasks and developing custom interfaces for each of them. The first step is an online tool for building noun synsets based on content of existing dictionaries. The goal of this stage is to establish YARN core content, test and validate crowdsourcing approach, prepare annotated data for automatic methods, and create a basis for the work with the other parts of speech.

As mentioned above, important characteristics of the project are its openness and recruitment of volunteers. Our crowdsourcing approach is different, for example, from the one described in (Biemann and Nygaard, 2010), where AMT turkers form synsets using the criterion of contextual substitutability directly. In our case, editors assemble synsets using word lists and definitions from dictionaries as "raw material". Obviously, such a task implies minimal lexicographical skills and is more complicated than an average task offered to AMT workers. Our target editors are college or university students, preferably from linguistics departments, who are native Russian speakers. It is desirable that students are instructed by a university teacher and may seek their advice in complex cases. As in the case of Wikipedia and Wiktionary, we foresee two levels of contributors: line editors and administrators with the corresponding privileges. According to our expectations, the total number of line editors can reach two hundreds throughout a year.

3 Raw Materials for YARN

We used two sources of "raw materials" for YARN: 1) Russian Wiktionary and 2) Small Academic Dictionary (SAD). Russian Wiktionary dump as of March 2012 was parsed and converted to database format using Wikokit software (Krizhanovsky and Smirnov, 2013). Wiktionary dump contains 51,028 nouns, including 45,646 single-word nouns; 30,031 entries have at least one definition. Besides the words and definitions Wiktionary dump contains occasionally synonym references and word usage examples. SAD data contain 33,220 word entries and 51,676 definitions. All single-word nouns were provided with frequencies based on the Russian National Corpus[8].

4 User Interface

The current synset editing interface can be accessed online [9]; its main window is presented in Figure 1.

"Raw data" are placed on the left-hand side of the interface: definitions of the initial word and examples, possible synonyms for each of the meanings in turn with definitions and examples. The right-hand part represents resulted synsets including words, definitions, and examples. In principle, an editor can assemble a "minimal" synset from the dictionary "raw material" simply with several mouse clicks, without any typing.

Synset assembly begins with a word, or "synset starter". The editor selects an item from the list of words ranked by decreasing frequency; already processed words are shaded. The editor can go through the words one after another or choose an arbitrary word using search box.

The top left-hand pane displays definitions of the initial word and usage examples if any. To simplify the view, editor can turn out examples or to blind individual definitions. Possible synonyms of the initial word are listed at the bottom-left pane, in turn with definitions and examples. The top-right pane displays a list of synsets containing the initial word. The editor can copy definitions and usage examples of the initial word from the top left of the interface to the current synset with mouse clicks. From the synonyms pane one can transfer bare words or words along with definitions and examples. The editor can add a new word to the list

[8] http://ruscorpora.ru/en/

[9] http://russianword.net/editor

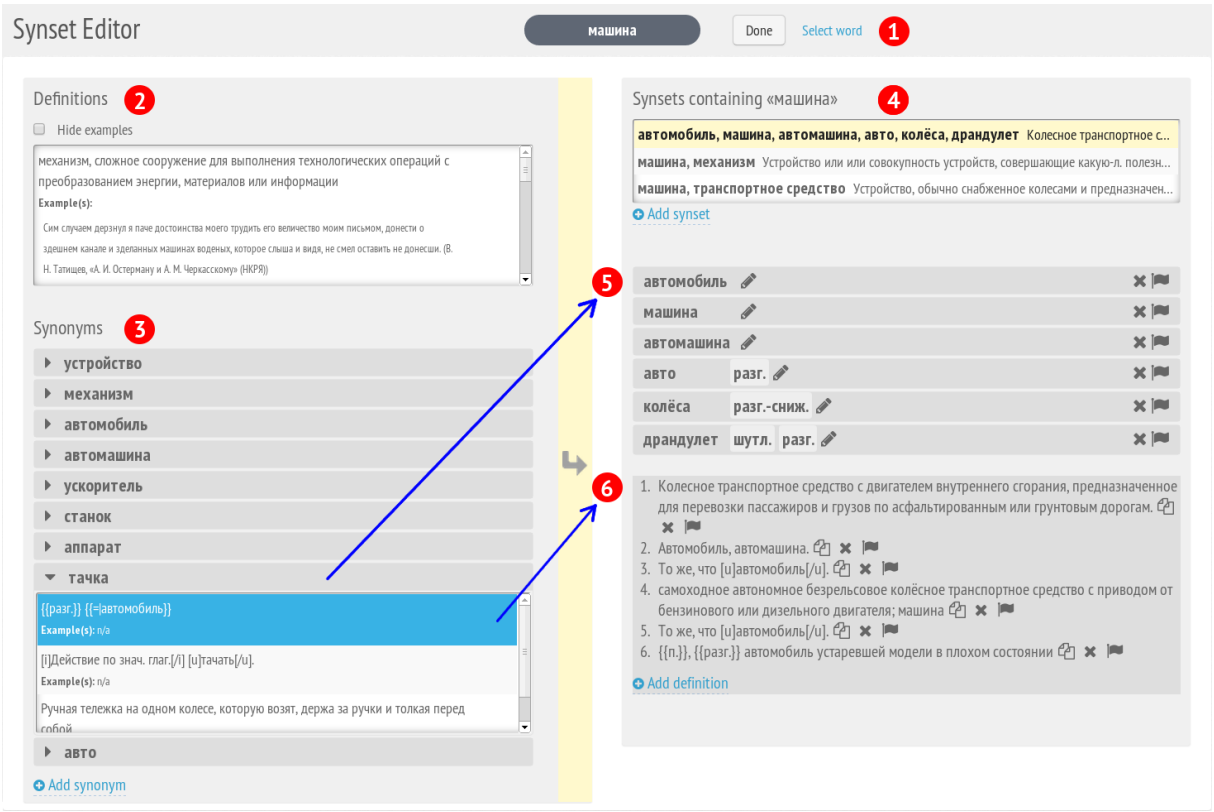

Figure 1: Main window of YARN synset assembly interface (interface captions are translated for convenience of readers into English; originally all interface elements are in Russian): 1) initial word; 2) definitions and examples of the initial word; 3) possible synonyms of the initial word with definitions and examples; 4) a list of synsets containing the initial word (active synset is highlighted); 5) words constituting the current synset; 6) definitions of the current synset. The arrows show how the information items from the left-hand side form synsets in the right-hand side.

of synonyms; it will appear with dictionary definitions and examples if presented in the parsed data. If the editor is not satisfied with the collected definitions, they can create a new one — either from scratch or based on one of the existing descriptions. Additionally, a word or a definition within a synset can be flagged as "main"; and be provided with labels. All synset edits are tracked and stored in the database along with timestamps and editor ID.

YARN software is implemented using Ruby on Rails framework. All data are stored in a PostgreSQL database. User authentication is performed through an OAuth endpoint provided by Facebook. The user interface is implemented as a browser JavaScript application. The application interacts with the server application via JSON API. The entire source code of the project is available in an open repository[10].

[10] https://github.com/russianwordnet

5 Preliminary Results

In the fall 2013 we conducted a pilot user study with 45 students of the linguistics department at the Ural Federal University. The experiment resulted in 1390 synsets; 970 of them are 'non-trivial', i.e. contain more than a single word (253 contain 2 words, 228 — 3 words, 207 — 4, 282 — 5+). Editors spent about two minutes on building a 'non-trivial' synset on average, which we find a very good result. Figure 2 shows the distribution of edit times for 2+ word synsets. Distribution of completed synsets by students is also skewed, e.g. top-5 contributors account for more than a third of all non-trivial synsets (329).

Figure 3 shows a linear trend of time spent by five top contributors on constructing consecutive non-trivial synsets. Four out of five demonstrate a learning effect: average time per synset tends to decrease while the editor proceeds through tasks.

In general, students were very positive about

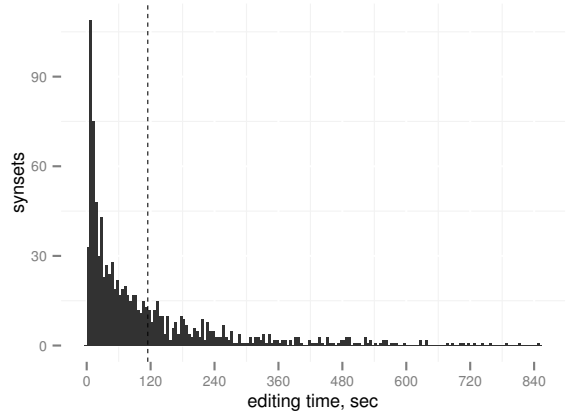

Figure 2: Distribution of times spent on non-trivial synset editing.

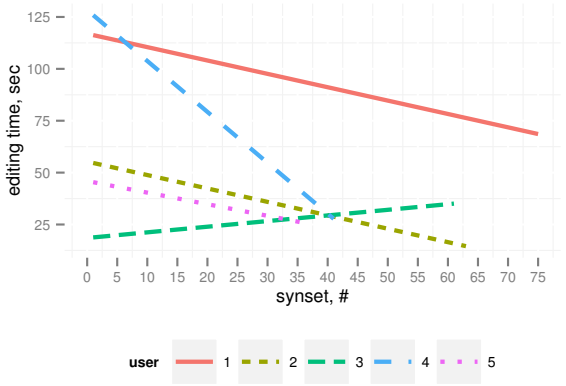

Figure 3: Linear trend of time spent on sequential edits of nontrivial synsets by top-5 contributors.

their participation in the experiment and the YARN interface. Participants mentioned flaws in parsed data, inability to delete an existing synset (we disabled this option during the experiment), and the inconvenience of label assignments as main disadvantages.

6 Conclusions

YARN synset assembly tool passed an initial testing and proved to be a usable tool for creation of thesaurus building blocks. Upon reading simple instructions, volunteers were able to quickly learn an intuitive interface and accomplish the synset assembly task without problems.

During the experiment we were able to diagnose some flaws related to interface design, editor guidelines, and internal data representation. In the future we will elaborate instructions and learning materials, clean existing and add more dictionary data, and perform a thorough evaluation of the interface. Then, we will work on an interface for linking synsets and expand YARN with verbs and adjectives.

Acknowledgments. This work is supported by the Russian Foundation for the Humanities, project #13-04-12020 “New Open Electronic Thesaurus for Russian”.

References

Irina Azarova et al. 2002. RussNet: Building a Lexical Database for the Russian Language. In *Proc. of Workshop on WordNet Structures and Standardisation, and How These Affect WordNet Applications and Evaluation, Gran Canaria, Spain*, pages 60–64.

Valentina Balkova et al. 2004. Russian wordnet. In *Proc. of the Second Global WordNet Conference*, pages 31–38. Citeseer.

Chris Biemann and Valerie Nygaard. 2010. Crowdsourcing Wordnet. In *Proc. of the 5th Global WordNet conference, Mumbai, India.*

Pavel Braslavski et al. 2013. YARN Begins. In *Proc. of Dialog-2013 (in Russian).*

Christiane Fellbaum. 1998. WordNet: An Electronic Database.

Ilya Gelfenbein et al. 2003. Avtomaticheskij perevod semanticheskoj seti WORDNET na russkij yazyk. In *Proc. of Dialog’2003 (in Russian).*

Verena Henrich and Erhard Hinrichs. 2010. GernEdiT-The GermaNet Editing Tool. In *ACL (System Demonstrations)*, pages 19–24.

Aleš Horák et al. 2006. DEBVisDic–First Version of New Client-Server Wordnet Browsing and Editing Tool. In *Proc. of the Third International Wordnet Conference*.

Andrew Krizhanovsky and Alexander Smirnov. 2013. An Approach to Automated Construction of a General Purpose Lexical Ontology Based on Wiktionary. *Journal of Computer and Systems Sciences International*, 52(2):215–225.

Natalia Loukachevitch and Boris Dobrov. 2002. Development and Use of Thesaurus of Russian Language RuThes. In *Proc. of Workshop on WordNet Structures and Standardisation, and How These Affect WordNet Applications and Evaluation, Gran Canaria, Spain*, pages 65–70.

Maciej Piasecki et al. 2012. WordnetLoom: a Wordnet Development System Integrating Form-based and Graph-based Perspectives. *International Journal of Data Mining, Modelling and Management*.