A tool for effective extraction of synsets and semantic relations from BabelNet

Conference Paper · April 2017

DOI: 10.1109/SSDSE.2017.8071954

2 authors, including:

Alexander Panchenko
University of Hamburg
115 PUBLICATIONS **887** CITATIONS

SEE PROFILE

A Tool for Effective Extraction of Synsets and Semantic Relations from BabelNet

Dmitry Ustalov*
*Institute of Natural Sciences and Mathematics
Ural Federal University
Yekaterinburg, Russia
Email: dmitry.ustalov@urfu.ru

Alexander Panchenko†
†Language Technology Group
Computer Science Department
Universität Hamburg
Hamburg, Germany
Email: panchenko@informatik.uni-hamburg.de

***Abstract*—Evaluation experiments in natural language processing often involve construction of samples from large lexical semantic resources, such as WordNet, Wiktionary, and OmegaWiki for evaluation and training purposes. The two most recurrent tasks are extraction of synsets and semantic relations between words. BabelNet is a resource which combines and interlinks all main lexical resources providing a unified assess to them. In this paper, we present BabelNet Extract, an open source tool which helps in addressing these two recurrent extraction tasks effectively in a parallelized manner from the large-scale multilingual BabelNet semantic network. The tool extracts individual word senses and the synsets they form as well as the semantic relations established between the synsets. We show its architecture, describe the output format, and discuss the use cases of the tool.**

I. Introduction

A high-quality lexical ontology is a crucial resource for addressing such problems as word sense disambiguation, search query extension and many other problems in the fields of natural language processing and artificial intelligence. BabelNet [1] is a large-scale multilingual semantic network available for more than 200 natural languages. Such a resource can be highly useful for under-resourced languages like Russian. Although it is possible to obtain BabelNet for both academic and commercial use, it is distributed in the binary form, which urges the development of a simple-to-use tool that extracts the structured linguistic data from this resource.

This paper, dedicated to the development of such a tool, is organized as follows. We review the existing tools in Section II. Then, we present BabelNet Extract, an open source tool for extracting structured lexical data from the BabelNet lexical ontology, in Section III. The tool uses the lexical entry output format described in Section IV. Finally, we discuss the use cases of BabelNet Extract in Section V and provide concluding remarks in Section VII. The software we demonstrate is open source and is available under a libré license.

II. Related Work

WordNet, a very well-known lexical database for English [2], is distributed as a set of plain text files representing structured linguistic data. Although these files are easy to parse, the WordNet developers offer convenient software for manipulating the database.

Wiktionary, a large-scale multilingual lexical ontology [3], is created using crowdsourcing. The volunteer editors collaboratively contribute to the resource via wiki markup. The content is organized in Web pages, one page per word, and the expressed linguistic knowledge is available in an semi-structured form. The end users can use the software like JWKTL [4] and Wikokit [5] to parse the markup and extract structured data from Wiktionary for further use.

In contrast to the above-mentioned lexical resources, BabelNet offers HTTP and Java API [6] as well as the RDF and SPARQL endpoints [7], conforming to the formats of the Linguistic Linked Open Data Cloud framework [8]. Due to the large scale of the BabelNet, extraction of samples of synsets and semantic relations from it requires writing a specialized optimized and parallelized code. Especially computationally heavy are extraction of transitive semantic relations, e.g., transitive hypernyms, as such operations require traversals of the huge BabelNet graph. These computational complexities complicate large-scale inter-resource matching and linking. As the result, the end users need to either invest additional time to write the required software by themselves or even ignore BabelNet at all. The tool presented in this paper solves this issue, providing an effective means for extraction of massive samples of synsets and semantic relations from BabelNet.

III. BabelNet Extract

BabelNet Extract is a Java tool for extracting two kinds of data from BabelNet: the synsets (concepts) and the relation between them. It is built on top of the BabelNet Java API that encapsulates two approaches for accessing the BabelNet contents: the offline Lucene-based index and the online HTTP API via http://babelnet.io/ (Fig. 1).

BabelNet Extract provides both a command-line interface and an application programming interface for seamless integration into various natural language processing pipelines. Its command-line interface has four different execution modes called the *actions* for extracting certain types of linguistic data from BabelNet. These actions wrap the general-purpose BabelNet API and implement specific data extraction routines that have not been explicitly provided by the underlying API. It is possible to specify both the desired language and the part-of-speech tag, for instance, to exclude the English language

978-1-5386-1593-5/17/$31.00 ©2017 IEEE

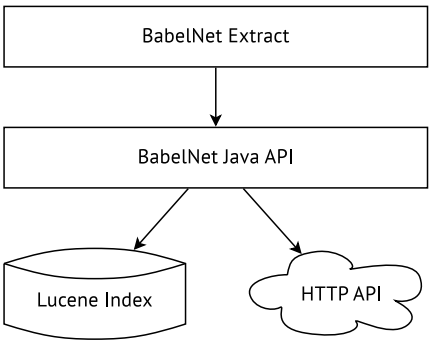

Fig. 1. Architecture of BabelNet Extract.

data in the studies dedicated to the Russian language. In case the command-line interface is used, the output is written to tab-delimited plain text files in the UTF-8 character set.

A. Synset Extraction

The synset extraction action iterates over BabelNet synsets and outputs the tab-delimited text file, each line of which contains the identifier of the synset, the size of the synset, and the comma-separated list of lemmas included into the synset.

B. Sense Extraction

Similarly to the previous one, the lexical sense extraction action inputs the set of synsets and extracts the senses contained by the input synsets. Unlike the synset extraction action, this action makes it possible to define the list of the synsets to be processed. Additionally, this action extracts the sense frequencies according to BabelNet.

C. Cluster Extraction

The cluster extraction action is designed to simplify the alignment of BabelNet with other semantic lexical resources, including unsupervised ones. For that, it iterates over the provided sets of words, i.e., the clusters, and writes two files: the words file and the synsets file. Each line of the former file contains an identifier of the cluster, a word included into this cluster, and a comma-separated set of BabelNet synsets containing the word. The latter file contains the list of all the synsets appeared in the former file.

D. Relation Extraction

The semantic relation extraction action takes the set of synsets S as an input and extracts its neighbors with the maximum depth of d from the BabelNet semantic network (S, E). This action currently supports only hypernymy and hyponymy relations: each neighbor has a distance provided with the plus sign if the neighbor is reachable through the hypernym, otherwise, the minus sign is written. For that, a variation of the depth-first search algorithm is used (Fig. 2).

The output is written into the tab-delimited text file, each line of which contains a synset identifier and a comma-separated list of the identifiers of other synsets the former establishes semantic relations with. The distance to each related synset is represented by the number following after the colon after each written identifier.

Input: a synset $s \in S$, edges E, and the maximum depth d.
Output: a map N of s neighbors to the distances to them.

$N[s] \leftarrow 0$
$q \leftarrow [s]$
while $q \neq \emptyset$ **do**
  $s' \leftarrow \text{shift}(q)$
  for all $(s', n) \in E$ **do**
    $\text{step} \leftarrow N[s']$
    if $n \notin N \wedge |\text{step}| < d$ **then**
      if $\text{step} = 0$ **then**
        $N[n] \leftarrow \begin{cases} +1, & \text{if } n \text{ is a hypernym} \\ -1, & \text{otherwise} \end{cases}$
      else
        $N[n] \leftarrow \text{sign}(\text{step}) \times (|\text{step}| + 1)$
      end if
      $\text{push}(q, n)$
    end if
  end for
end while

Fig. 2. Semantic relation extraction algorithm.

IV. Lexical Entry Output Format

In order to describe a common textual format for lexical entries, we use the context-free grammar encoded using the version 4 of ANother Tool for Language Recognition [9] (ANTLR). This grammar, shown in Fig. 3, represents the following properties of the lexical entry:

- *lexeme* (also known as *word entry*) is a non-empty set of *spans* separated by the underscore symbol;
- *span* is a pair of *lemma* and the corresponding *pos* tag;
- *lemma* is a canonical form of a word;
- *pos* is a part-of-speech tag of the given *lemma*;
- *sense* is a pair of the sense identifier (*id*) of the *lexeme* and the set of sense *labels* separated by the underscore symbol;
- *frequency* is the frequency of the described lexical entry.

In fact, only the *lemma* field is mandatory, while the rest of fields are optional. An example, depicted in Fig. 4, represents the parse tree of the lexical entry composed of the noun phrase "data mining", both lemmas of which are nouns (`NN`). The entry corresponds to the sense number one, holds the uncountable (`uncnt`) label and which relative frequency is 2.28.

V. Use Cases

We believe that BabelNet Extract can be useful for many applications involving the processing of large-scale semantic networks, including language resources evaluation, lexical resource linking, sense inventory applications, and community analysis.

```
sememe     : lexeme (HASH sense)? (COLON frequency)? EOF ;
lexeme     : span (UNDERSCORE span)* ;
span       : lemma (HAT pos)? ;
lemma      : STRING ;
pos        : STRING ;
sense      : id labels? ;
id         : INTEGER ;
labels     : (UNDERSCORE label)+ ;
label      : STRING ;
frequency  : INTEGER | DECIMAL ;
INTEGER    : [0-9]+ ;
DECIMAL    : [0-9]* DOT [0-9]+ ;
STRING     : CHAR+ ;
CHAR       : ~[^#:_\t\n\r ] ;
HAT        : '^' ;
HASH       : '#' ;
COLON      : ':' ;
UNDERSCORE : '_' ;
DOT        : '.' ;
```

Fig. 3. Context-free grammar in ANTLR 4

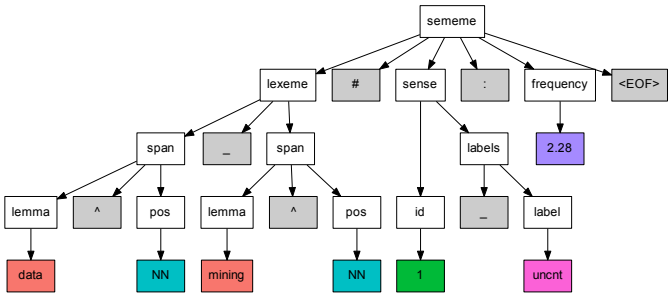

Fig. 4. A parse tree for `data^NN_mining^NN#1_uncnt:2.28`.

A. Language Resources Evaluation

Evaluation experiments in natural language processing often involve construction of samples from large lexical semantic resources, such as BabelNet, for the evaluation purposes. This is especially useful for under-resourced languages like Russian, because BabelNet can be considered as a gold standard in studying the word sense induction methods.

B. Lexical Resources Linking

Similarly to the previous case, BabelNet can be useful in linking different lexical semantic resources for under-resourced language in such procedures as ontologisation [10] that require a gold standard to be available.

C. Sense Inventory Acquisition

For word sense disambiguation tasks, BabelNet Extract can provide a sense inventory that represents the words and the senses associated with them in the target language. It is especially useful in addressing problems involving word sense disambiguation.

D. Machine Translation

Since BabelNet is a multilingual semantic network, it is possible to apply the semantic knowledge extracted using BabelNet Extract in various machine translation and multilingual natural language processing problems, including multilingual information retrieval, etc.

E. Community Analysis

In community analysis and high-performance graph analytics, BabelNet Extract can be the source of the real world network suitable for both development, tuning and evaluating not just the language resources, but the graph traversal algorithms for community detection and high-performance computing benchmarking.

VI. Distribution

The source code of BabelNet Extract is available on GitHub under the terms of Apache License 2.0: https://github.com/nlpub/babelnet-extract.

We also provide a Docker-based container image [11] with properly configured BabelNet API and BabelNet Extract: https://hub.docker.com/r/nlpub/babelnet/. The image is designed for using the offline version of BabelNet, but it does not bundle it due to the licensing issues. Instead, the user has to download the personal version of BabelNet from its website and then provide the corresponding dictionary into the container as the volume mount.

VII. Conclusion

In this paper, we presented and described BabelNet Extract, an open source tool for effective extraction of synsets and semantic relations from the BabelNet semantic network. The tool is available for general public and supports all the languages supported by BabelNet.

Acknowledgment

The reported study was funded by RFBR according to the research project No. 16-37-00354 mol_a. This work is supported by the Russian Foundation for the Humanities project no. 16-04-12019 “RussNet and YARN thesauri integration”. We acknowledge the support of the Deutscher Akademischer Austauschdienst (DAAD) and by the Deutsche Forschungsgemeinschaft (DFG) foundation under the “JOIN-T” project.

REFERENCES

[1] R. Navigli and S. P. Ponzetto, "BabelNet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network," *Artificial Intelligence*, vol. 193, pp. 217–250, 2012.

[2] C. Fellbaum, *WordNet: An Electronic Database*. MIT Press, 1998.

[3] Wiktionary. [Online]. Available: https://www.wiktionary.org/

[4] T. Zesch, C. Müller, and I. Gurevych, "Extracting Lexical Semantic Knowledge from Wikipedia and Wiktionary," in *Proceedings of the 6th International Conference on Language Resources and Evaluation*. Marrakech, Morocco: European Language Resources Association, 2008, pp. 1646–1652.

[5] A. A. Krizhanovsky and A. V. Smirnov, "An approach to automated construction of a general-purpose lexical ontology based on Wiktionary," *Journal of Computer and Systems Sciences International*, vol. 52, no. 2, pp. 215–225, 2013.

[6] BabelNet™ — Download. [Online]. Available: http://babelnet.org/download

[7] BabelNet™ — SPARQL Endpoint. [Online]. Available: http://babelnet.org/sparql/

[8] C. Chiarcos, S. Hellmann, and S. Nordhoff, "Towards a Linguistic Linked Open Data cloud: The Open Linguistics Working Group," *TAL*, vol. 52, no. 3, pp. 245–275, 2011.

[9] T. Parr, *The Definitive ANTLR 4 Reference*. The Pragmatic Programmers, LLC, 2013.

[10] M. Pennacchiotti and P. Pantel, "Ontologizing Semantic Relations," in *Proceedings of the 21st International Conference on Computational Linguistics and the 44th Annual Meeting of the Association for Computational Linguistics*, ser. ACL-44. Stroudsburg, PA, USA: Association for Computational Linguistics, 2006, pp. 793–800.

[11] D. Merkel, "Docker: Lightweight Linux Containers for Consistent Development and Deployment," *Linux Journal*, vol. 2014, no. 239, 2014.