ONTOLOGY-DRIVEN COMPUTATIONAL PROCESSING FOR UNSTRUCTURED TEXT

Olga Nevzorova[1], Vladimir Nevzorov[2]

[1]Kazan Federal University, Tatarstan Academy of Sciences, Russia
[2]Kazan National Research Technical University named after A.N. Tupolev, Russia

{onevzoro, nevzorovvn}@gmail.com

At the present time there are various widely used frameworks for Natural Language Processing (Core NLP Suite, Natural Language Toolkit, Apache OpenNLP, GATE and Apache UIMA, etc.).

Stanford CoreNLP [1] provides a set of natural language analysis tools. It can give the base forms of words, their parts of speech, normalize dates, times, and numeric quantities, mark up the structure of sentences in terms of phrases and word dependencies, indicate which noun phrases refer to the same entities, indicate sentiment, extract particular or open-class relations between entity mentions, etc. Natural Language Toolkit [2] (NLTK) is a leading platform for building Python programs to work with language data. It provides easy-to-use interfaces to over 50 corpora and lexical resources such as WordNet, along with a suite of text processing libraries for classification, tokenization, stemming, tagging, parsing, and semantic reasoning, wrappers for industrial-strength NLP libraries.The Apache OpenNLP [3] library is a machine learning based toolkit for the processing of natural language text. The GATE framework [4] comprises a core library and a set of reusable Language Engineering modules. The framework implements the architecture and provides facilities for processing and visualising resources, including representation, import and export of data.

Unstructured Information Management applications (UIMA) [5] are software systems that analyze large volumes of unstructured information in order to discover knowledge that is relevant to an end user. An example UIMA application might ingest plain text and identify entities, such as persons, places, organizations; or relations, such as works-for or located-at. UIMA enables applications to be decomposed into components. Each component implements interfaces defined by the framework and provides self-describing metadata via XML descriptor files. The framework manages these components and the data flow between them. UIMA additionally provides capabilities to wrap components as network services.

For the Russian language, the general classes of computational linguistic tools have been developed, including those based on semantic technologies. Let us mention some systems. OntosMiner system [9] uses semantic ontologies to analyze natural language text. The outcome is a set of searchable and conceptually structured data, which can be categorized, browsed and visually presented in semantic networks. Tamita parser [10] is the linguistic tool for extracting structured data (facts) from text. The extraction of facts is based on context-free grammars and dictionaries of keywords.

Compreno technology [9] is a universal linguistic platform for applications that solve a variety of applied tasks for NLP. In Compreno project, the ultimate goal is to achieve the syntactic and semantic disambiguation. Semantic and syntactic representations are viewed rather as two facets of the same structure. Another (interrelated) feature of the Compreno parsing technology is that syntactic and semantic disambiguation are processed in parallel from the very start (in contrast to the architecture more usual for the NLP systems — the semantic analysis follows the syntactic one).

However, many of NLP systems are commercial and do not provide a clear enough clarification of the details of the main processes.

This article discusses another framework ("OntoIntegrator" system) developed for ontology-driven computational processing for NLP [6].

An important features of the "OntoIntegrator" system are supporting all the processes necessary to build a solution to the NLP task, including the development of ontologies, linguistic resources and specialized databases. We also developed the original ontology-based method of building solution for NLP tasks. The system is available for various NLP applications for Russian language and can be accessed by contacting developers.

The "OntoIntegrator" system includes the following functional subsystems, such as:

- the "OntoEditor+" subsystem for ontology development;
- the "Text analysis" subsystem;
- the subsystem of external linguistic recourses;
- the ontology subsystem;
- the "Integrator" subsystem.

The "OntoEditor+" subsystem supports the main table functions for development of ontology (addition, modification, deletion, automatic correction; keeping of more than one or compound ontologies, in other words with the general lists of relations, classes, text equivalents and others; an import of the ontologies with the different formats of data; a filtration of ontology; keeping of statistics automatically, searching for chains of relations and others). The functions of the visualization unit support different graphic modes of system, including the graphic mode of the ontology modeling.

The “Text analysis” subsystem contains a base linguistic tools for processing Russian, include tools for tokenization (splitting of text into words), part of speech tagging, grammar parsing (identifying things like noun and verb phrases), word sense disambiguation, named entity recognition, and more.

The subsystem of the external linguistic recourses supports storage of the basic linguistic recourses include the grammatical dictionaries and a set of special linguistic data bases.

The ontology subsystem is used in building solutions to applied NLP problems.

The “Integrator” subsystem implements a building the applied linguistic problem solution using all available system resources. The solution is being built under control of the ontology system that includes domain ontologies, the model ontology and the task ontology.

The ontology system is a connected three-component system. The components of this system are the ontologies mentioned above.

To construct a solution to applied linguistic problems, it is necessary to create a new concept of the task ontology (a task-concept) and to perform the structural decomposition of the new concept into the concepts (logical parts) included in the task ontology. The task ontology contains special classes, objects and object properties (relations) to implement the structural decomposition.

To execute the structural decomposition, a special software module has been developed. Thus, the structural scheme of the solution of the problem is determined. Then the structural elements of solution of the linguistic problem are mapped to the set of the concepts of model ontology. The models are used for computational processing of the task-concept. There are the following classes of computational models such as a model for property assigning, a model for relation defining, and computational processing of basic NLP problems. Many models are open and replenished dynamically. The ontology of models contains classes and instances of objects, and their properties (relations).

For easier interpretation all model-concepts are split to following groups supported by complex visualization mechanisms:

- Basic models providing the minimal functionality of ontology system;
- Syntactical models for extracting syntactical structures from a text (text models);
- Semantic models that create both an adequate interpretation of the decision results for applied task, and defining the connections between structural elements of model-concepts and the sequences of syntactic structures extracted from the input text;
- user-defined models dynamically created.

Thus, a task-concept defines the structural sequence of subtasks implemented at the level of model-concepts. Data for model-concepts are extracted from a input text and are interpreted in structural components (classes, instances, properties) of domain ontology (the third component of the ontological system).

The results obtained are displayed in the view window of the extracted textual models of the processed text and are saved as annotated output text.

At present time we developed a main library of basic and applied linguistic tasks based on ontology-driven technology of "OntoIntegrator" system. The library includes the solutions for such tasks as resolution of various types of polysemy, named entity recognition, annotation of text for special purposes and others. All solutions built are the components that may be integrated into new applications.

The software solutions of our system were used for processing mathematical texts, namely for extracting mathematical terminology from texts, annotating mathematical articles [7], and designing ontology of professional mathematics.

Acknowledgment

This work was funded by the subsidy allocated to Kazan Federal University for the state assignment in the sphere of scientific activities, grant agreement no. 1.2368.2017

References

1. Manning, Christopher D., Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky. 2014. The Stanford CoreNLP Natural Language Processing Toolkit In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations, pp. 55-60.
2. Natural Language Toolkit, http://www.nltk.org/
3. Apache OpenNLP, http://opennlp.apache.org/
4. H. Cunningham, V. Tablan, A. Roberts, K. Bontcheva (2013) Getting More Out of Biomedical Documents with GATE's Full Lifecycle Open Source Text Analytics. PLoS Comput Biol 9(2): e1002854.
5. Ferrucci, D. et al. (2009) Unstructured Information Management Architecture (UIMA) Version 1.0. OASIS Standard, March 2009.
6. Nevzorova O., Nevzorov V. Terminological annotation of the document in a retrieval context on the basis of technologies of system “OntoIntegrator”. International Journal «Information Technologies & Knowledge", 2011, vol. 5, n. 2, pp. 110-118.
7. Olga Nevzorova, Nikita Zhiltsov, Danila Zaikin, Olga Zhibrik, Alexander Kirillovich, Vladimir Nevzorov, Evgeniy Birialtsev Bringing Math to LOD: A Semantic Publishing Platform Prototype for Scientific Collections in Mathematics. The Semantic Web - ISWC 2013: 12th International Semantic Web Conference, Sydney, NSW, Australia, October 21-25, 2013, Proceedings, Part I (Lecture Notes in Computer Science) 2013th Edition, pp. 379-394.
8. Anisimovich K. V., Druzhkin K. Ju., Minlos F. R., Petrova M. A., Selegey V. P., Zuev K. A. (2012) Syntactic and semantic parser based on ABBYY Compreno linguistic technologies. Computational Linguistics and Intellectual Technologies Papers from the Annual International Conference “Dialogue”, 2012, Issue 11, Volume 2 of 2. Papers from special sessions, pp. 91-103.
9. Khoroshevsky V.F. (2009) Ontology Driven Multilingual Information Extraction and Intelligent Analytics. In: Proc. of NATO Advanced Research Workshop on Web Intelligence and Security, November 18-20, 2009 in Ein-Bokek, Israel, 2009.
10. Tamita parser https://tech.yandex.ru/tomita/doc/tutorial/concept/about-docpage/