An Approach to Semantic Natural Language Processing of Russian Texts

Valery Solovyev[1], Vladimir Polyakov[2], Vladimir Ivanov[1],
Ivan Anisimov[3], Andrey Ponomarev[3]

[1] Kazan Federal University, Kazan, Russia
[2] Institute of Linguistics, Russian Academy of Sciences, Moscow, Russia
[3] National Research Technologic University "MISiS", Moscow, Russia

{maki.solovyev, nomemm}@mail.ru, pvn-65@mail.ru,
{valeter, genhoo}@mail.ru

Abstract. The article contains results of the first stage of a research and development project aimed at creating a new generation of intellectual systems for semantic text analysis. Described are the main principles, system architecture, and task list. The features cloud and cluster architecture realization are regarded as well.

Keywords: Paradigmatics, syntagmatics, parallel computing, cloud computing, natural language processing.

1 Introduction

The history of the development of natural-language text processing began over half a century ago; it has its ups and downs, however, as an industrial technology within the framework of informational technologies, this direction has been forming during the past twenty years. Nowadays, there are several program libraries of Natural Language Text Processing (NLTP) in Russia and abroad, and they can be regarded as positive examples of the complex solution of the problem (AOT [26], RCO [27], DICTUM [28], GATE [29], UIMA [30], OpenNLP [31]). However, the overall progress in the sphere of IT collected in the field of NLTP systems creation, particularly, and intellectual systems in general, allows looking at the problem of NLTP from another angle.

The topicality of parallel and distributed computing use in the sphere of NLTP is defined by the following factors:

There is need. Today there are millions of terabytes of texts in the Internet and in the corporative medium that are of great interest from the point of view of machine learning, creating of more powerful search systems, and new cutting-edge applications in the sphere of NLTP.

There is possibility. The task of NLTP suits for both parallel and distributed computing, due to the well-known autonomy of text units on each of its levels

(tokens, morphoforms, chunks, sentences). This autonomy allows organizing mass parallel computation of numerous NLTP tasks.

Technical preconditions in the sphere of IT. There has been achieved a technological limit of increasing the processing power of computers by increasing the clock rate. The world leading processors producer, Intel company, stopped producing mononuclear processors. The growth of processing power is more and more conditioned by the architecture of the computation system. Computation resources have become much cheaper and more available. The carrying capacity of the channels of digital communication has reached the level when it has stopped being an obstacle for the organization of distributed computing in the net.

2 Principles for Developing of the New Generation Technologies for Natural Language Processing

Our group of researchers has formulated the principles, which must underlie the creation of program libraries and new generation technologies for natural language processing.

Division of algorithms and data. As a rule, in modern systems of NLTP, algorithms and data are so deeply intertwined that it is almost impossible to use them separately.

We suggest to follow the principle of division, thus, it is possible to use ready sets of data and change the algorithms to a more powerful one, and vice versa, by only changing the algorithms, which makes the researcher's work considerably easier.

Similarly, third-party developers can specialize in ready sets of data for linguistic support of standard algorithms and program libraries.

This principle is widely used by the developers of foreign systems of natural language processing [29-31].

Open algorithm standards and data formats. A consequence of the first principle, which allows comparing achievements in NLP sphere on a healthy competitive basis.

This does not mean open program code or other objects of intellectual property.

Pipeline architecture. At present the market has a positive experience of creating program platforms on the base of Java language, which serve for the aim of integration of packets for NLP. Nowadays, the most popular projects are GATE, UIMA and OpenNLP. But there is a clear lack of libraries for Russian language processing compatible with the mentioned platforms. Developing business oriented applications for Russian language based on the stated platforms is laborious both in terms of time and technology.

The solution for overcoming this obstacle is to simplify interaction with the platform on the user level, where software can be used as a service within the framework of strictly regulated scenarios and for solution of user's certain tasks.

Iterativity. Due to the ambiguity of language on all of its levels, it is impossible to get a 100% precise result in the process of NLTP. Well known causes of this ambiguity are homonymy, lexical polysemy and syntactical polysemy.

The suggested solution is to reiterate different levels of analysis, repeatedly running tasks after some ambiguity was resolved.

For example, preliminary morphological analysis + chunking [1] + secondary morphological analysis (resolution of polysemy), etc.

Frequency, F1-measure, of the mini-corpus. Improvement of each type of analysis should be based on the frequency of the occurring phenomena. For example, before coping with homonymy, one should conduct its frequency analysis and work with its most frequent cases.

F1-measure is a reliable way to check the quality of the analysis. For every text processing task a reference mini-corpus (a set of texts with a reference marking) should be created to test the developed methods.

Practice shows that such mini-corpora hand-marked for 10-100 uses of the phenomena in question (1000-10000 thousand words) are sufficient at the current stage of NLTP development.

Orientation on the technologies of data extraction from the text (Information Extraction). It is already evident that Information Extraction technology is becoming the most real alternative for complete NLP, which is, probably, unattainable in the nearest decade.

The authors of the project developed a method to extract relations, which considerably lowers the work content of application creation, as it does not require a big set of teaching examples. It is planned to integrate this method in the library as a separate application for open information extraction.

Orientation on ontologies. Ontologies are the most standardized format of paradigmatic data representation, this allows to build on the existing technological systems. Languages of ontologies presentation (RDFS, OWL) are now applied for program components description (OWL-S) and their data sets (SKOS).

Machine learning. The renunciation of manual language specification. One should try to create products with minimal hand labor of linguists. Orientation on technologies that do not require great volume of manual work, such as open information extraction, will allow to efficiently adapt the created applications for the platform of cloud computing.

Multilanguage. The architecture and design of separate modules must not obstruct the creation of multilingual systems (search engines, machine translation, etc.).

Multifunction. If the functions of NLTP are well standardized, the applications of the whole system or its parts can be diverse.

Cloud computing. Technologies of cloud computing [20] seem to be the most suitable means of computation needed for large scale natural language processing. Cloud computing eliminates a broad range of issues connected with the producing capacity of the machinery, availability of services due to high technological and price barriers that companies, novices and research groups creating applications for intellectual analysis in any subject field have to overcome.

[1] Chunking is, in Russian NLTP tradition, breaking a sentence into minimal semantically meaningful parts (chunks).

Outside of Russia, besides widely spread Amazon Web Services [32], Windows Azure [33] and Google App Engine [34], there are such popular projects as OpenNebula [35] and Ubuntu Enterprise Cloud [36].

Open instruments. Application of standard platform like GATE and UIMA, openness on the level of branch standards for algorithms and formats, allows incessantly developing new instruments for NLTP systems. It is known that most companies and research groups cannot work on this problem because of a lack of access to effective text processing libraries, and the fact that development of such libraries takes a lot of time and costs. Even if there are free libraries, there arises the question of their installation, launch and integration. Creation of a complex of language processing tools on a unified technological platform will simplify the solution of this problem as well.

Commercialization of temporary results of scientific research work. There appear preconditions for integration of partial developments in the sphere of NLP into a united system, the process of program products maintenance becomes easier.

On the base of above mentioned principles, we formed a structure of program components of the future program complex (fig. 1).

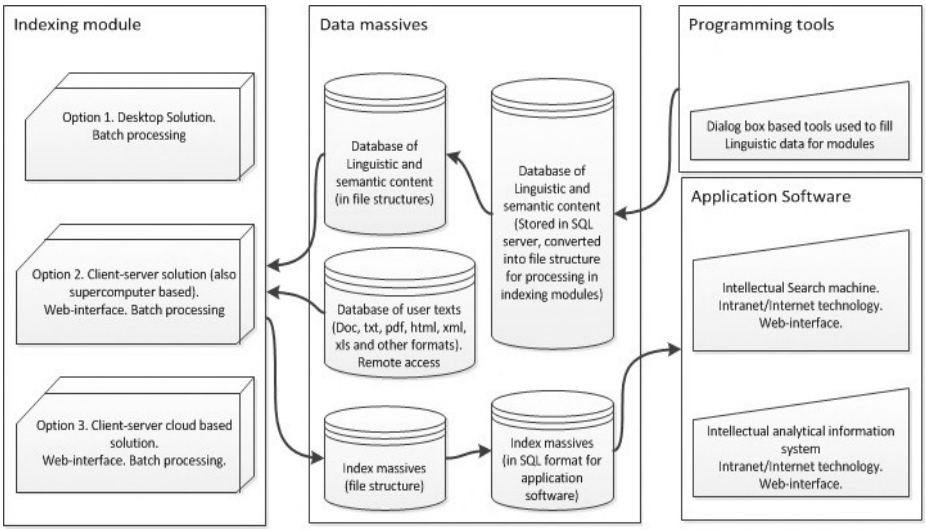

Fig. 1. Future program complex architecture

3 NLTP Tasks Included in the Library

The question of the content of the tasks to be included in the program library was deduced from the balance of desirable and maximally possible current development of science, technology and NLTP practice.

Classically [4, 8], it is believed that the process of Russian text processing is divided in to three stages:

- morphologic analysis;
- syntactical analysis;
- semantic analysis.

However, detailed study of the state of the art in the sphere of NLTP shows:

1) This statement does not take into consideration such important stages as preliminary processing of the text (conversion of formats, codes, clearance of control characters), tokenization (exposure of symbol chains, processing of extralinguistic data), pragmatic text processing (exposure of the information value with respect to the task at hand, etc., associating to the context);

2) The level of development of the tasks at each of the three macro-stages can be different. Thus, one can say that:

- The stage of morphological analysis is well elaborated from the scientific, as well as engineer point of view6;
- The stage of syntactic analysis is elaborated from the scientific point of view, but from the point of view of program engineering there is still considerable work to be done on optimal solutions in computation performance;
- The stage of semantic analysis still has a lot of gaps, and it is far from the completion of research. However some elements of semantic analysis (ontologies, semantic roles) are elaborated enough to provide the possibility of their engineer realization.

Considering these circumstances, our work group developed a list and a sequence of NLTP tasks, which we decided to include into the basic software library of NLTP[2]:

1. Preprocessing [20, 21, 24];
2. Tokenization-1 [22];
3. Morphological analysis;
4. Stemming[3] [1, 2];
5. Prediction of morphological characteristics[4];
6. Segmentation[5]-1 [7, 13];
7. Tokenization-2;
8. Set phrases and idioms identification [9, 10];
9. Building of broadened grammar vector[6];

[2] Working project title is "NLP@Cloud".

[3] In our case stemming includes not only finding the stem of a word, but also resolving the morphologic structure of a lemma.

[4] Used for words not found in the dictionary.

[5] In our case the task of segmentation is to identify syntactic constructions which are separated by punctuation marks in complex sentences.

[6] Broadened grammar vector – a special notation, used to transit from shallow to deep syntactic analysis

10. Chunking-1 [3, 6, 11, 12, 21];
11. NER (Named Entity Recognition) [14, 17];
12. IER (Identified Entity Recognition)[7][23];
13. NPR (Noun Phrase Recognition);
14. Morphological analysis-2. Homonymy resolution-1 [15, 16];
15. Segmentation-2;
16. Thematic classification of the text [18, 19];
17. Identifying communicative meaning of inquiries[8];
18. Binding to ontology. Homonymy resolution-2. Polysemy resolution-1;
19. Text nucleus detection[9];
20. Chunking-2;
21. Syntactic tree analysis[10]-1;
22. Referential analysis[11]-1;
23. Detection of actant semantic roles;
24. Binding to ontology-2. Homonymy resolution -3. Polysemy resolution -2;
25. Syntactic analysis-tree-2;
26. Referential analysis-2;
27. Connotative classification[12];
28. Identification of new concepts and their binding to ontology[13].

We can say that most tasks connected with morphology and syntax processing are included in the basic set of library functions.

It also includes tasks from the semantic level, which are mainly based on paradigmatic structures of knowledge (ontologies). Semantic roles, referential links and meanings belong to the syntagmatic sphere. Everything concerning other syntagmatic relations (temporal, spatial, causal and other links and relations) are currently left out of the basic library of NLTP. Due to incomplete distinctness in standards of semantic processing and high computation loads necessary for this kind of analysis, tasks related to syntagmatics will be included in user-end applications.

We can speak of a new standard of NLTP, which is introduced in this work. This standard set of NLTP instruments can be called paradigm-oriented text processing, or POTP.

[7] Identifying entities which have a numeric, digital or temporal quality.

[8] Identifying communicative meaning of inquiries is finding a relevant area of human usage for piece of text. This is usually different from thematic classification.

[9] By text nucleus we imply the most frequent meaningful word or entity in a text. A nucleus can be graded (i.e. include multiple words and entities ranged by frequency).

[10] Dependency grammar trees are used.

[11] Anaphora and cataphora links are marked, the denotative status of concepts is identified, referential ambiguity is resolved.

[12] Connotative classification is based on entities rather than texts as a whole.

[13] Based on WordNet [25] lexical ontology.

4 Conclusion

Let us enumerate the main innovations suggested in this work, which were not applied in industrial developments of NLTP before or were applied on a more limited scale, including:

- Iterativity;
- Chunking;
- Broadened grammar vector;
- Communicative classification;
- Connotative classification be objects;
- Text nucleus (graduated);
- Automatic building of ontology;
- Improved resolution of ambiguity (morphological, lexical, syntactic);
- Use of mechanism of semantic roles;
- Technology of text classification Rubryx [19];
- Lexical and syntactic portraits for resolution of lexical polysemy;
- Orientation to high-capacity computing.

An important feature of the suggested solution is its flexibility, that allows setting the complex of program libraries for different applied tasks. Openness (in the functional) secures the possibility of the system development in future. The suggested technologies are based on the latest achievements in Semantic Web sphere and ontological systems.

Acknowledgments. The work is supported by grants of Foundation for Assistance to Small Innovative Enterprises, Russia (program Soft-2012, contract №10151p/17593 (28.04.2012), program UMNIK, contract №16925/15-12 (21.05.2012)). The project is performed under a state assignment to universities by Ministry of Education and Sciences of Russia (№ 8.3358.2011). It is a joint project of National University of Sciences and Technology "MISiS" and Kazan Federal University.

References

1. Alvares, R. V., Mondaini, R.: Evaluation of Stemming Errors: Towards a Qualitative Analysis. In: XXXI CNMAC Conference proceedings (1999)
2. Alvares, R. V.; Garcia, A. C. B.; Ferraz, I. N.: STEMBR: A Stemming Algorithm for the Brazilian Portuguese Language. In: 12th Portuguese Conference on Artificial Intelligence (EPIA 2005), Covilhã, Portugal. Lecture Notes in Artificial Intelligence. v. 3808, pp. 693–701 (2005)
3. Attardi, G., Dell'Orletta, F.: Chunking and Dependency Parsing. In: Proceedings of LREC 2008 Workshop on Partial Parsing, Marrakech (2008)
4. Handbook of Natural Language Processing. Second Edition. Edited by Nitin Indurkhya, Fred J. Damerau. CRC Press, 666 p. (2010)

5. Antonopoulos, N., Gillam, L.: Cloud Computing: Principles, Systems and Applications. Springer, 379 p. (2010)
6. Hacioglu, K.: A lightweight semantic chunking model based on tagging / HLT-NAACL-Short '04 Proceedings of HLT-NAACL 2004: Short Papers (2004)
7. Huang, X., Peng, F., Schuurmans, D., Cercone, N., Robertson, S. E.: Applying Machine Learning to Text Segmentation for Information Retrieval. Information Retrieval (2003)
8. Jurafsky, D., Martin J. H.: Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition. Prentice Hall, 988 p. (2009)
9. Khokhlova, M.: Applying Word Sketches to Russian. In: Proceedings of Raslan 2009. Recent Advances in Slavonic Natural Language Processing. Brno: Masaryk University, pp. 91–99 (2009)
10. Khokhlova, M., Zakharov, V.: Statistical collocability of Russian verbs. In: After Half a Century of Slavonic Natural Language Processing.Brno, pp. 105–112 (2009)
11. Koeling, R.: Chunking with maximum entropy models. In: ConLL '00 Proceedings of the 2nd workshop on Learning language in logic and the 4th conference on Computational natural language learning , volume 7, pp. 139–141 (2000)
12. Kudo, T.: Japanese dependency analysis using cascaded chunking. In: Proc. of COLING-02, proceedings of the 6th conference on Natural language learning. Volume 20, pp. 1–7 (2002)
13. Lobanov, B., Tsirulnik, L.: Statistical study of speaker's peculiarities of utterances into phrases segmentation. In: Speech Prosody: proceedings of the 3rd International conference, Dresden, Germany, May 2-5, 2006. V. 2, pp. 557–560 (2006)
14. Masayuki, A., Matsumoto, Y.: Japanese Named Entity Extraction with Redundant Morphological Analysis. In: Proc. Human Language Technology conference, North American chapter of the Association for Computational Linguistics. (2003)
15. Mihalcea, R.: Using Wikipedia for Automatic Word Sense Disambiguation. In: Proc. of the North American Chapter of the Association for Computational Linguistics (NAACL 2007), Rochester, April 2007 (2007)
16. Old, L.J.: Homograph disambiguation using formal concept analysis. In: Fourth International Conference on Formal Concept Analysis, 13th-17th February 2006, Dresden, Germany (2006)
17. Poibeau, Th. and Kosseim, L.: Proper Name Extraction from Non-Journalistic Texts. In: Proc. Computational Linguistics in the Netherlands (2001)
18. Polyakov, V. and Sinitsyn, V.: Method for automatic classification of web-resource by patterns in text processing and cognitive technologies. Text Collection, No.6, Publ. House Otechestvo, pp. 120–126 (2001)
19. Polyakov, V. and Sinitsyn, V.: RUBRYX: technology of text classification using lexical meaning based approach. In: Proc. of Intern. Conf. Speech and Computing (SPECOM-2003), Moscow, MSLU p. 137–143 (2003)
20. Prabhu, C. S. R.: Grid and Cluster Computing. PHI Learning (2013)
21. Erik, F., Kim Sang, T.: Introduction to the CoNLL-2000 shared task: chunking. In: ConLL '00 Proceedings of the 2nd workshop on Learning language in logic and the 4th conference on Computational natural language learning, Volume 7, pp. 127–132 (2000)
22. Schmidt, H.: Tokenizing. In Corpus Linguistics: An International Handbook. Walter de Gruyter, Berlin (2007)
23. Soraluze, A., Alegria, I., Ansa, O., Arregi, O., and Arregi, X.: Recognition and Classification of Numerical Entities in Basque., In: Proceeding of Recent Advances in Natural Language Processing, RANLP 2011, 12-14 September, Hissar, Bulgaria (2011)

24. Zhu, X.: Common Preprocessing Steps. CS769 Spring 2010 Advanced Natural Language Processing (2010)
25. WordNet, lexical database for English, http://wordnet.princeton.edu
26. Automatic Russian Text Processing project, http://aot.ru
27. RCO, search and analytic systems developer, http://www.rco.ru
28. DICTUM, text analysis tools developer, http://www.dictum.ru
29. General Architecture for Text Engineering, http://gate.ac.uk
30. Unstructured Information Management Architecture, http://uima.apache.org
31. OpenNLP, http://opennlp.apache.org
32. Amazon Web Services, http://aws.amazon.com
33. Windows Azure, http://www.windowsazure.com
34. Google App Engine, https://appengine.google.com
35. Lutz, S., Keith, J., Burkhard, N.: The Future of Cloud Computing. Opportunities for European Cloud Computing beyond 2010. European Commission Expert Group Report (2010)
36. Ubuntu Cloud, http://help.ubuntu.com/community/UEC