Computational Linguistics and Intellectual Technologies:
Proceedings of the International Conference "Dialogue 2016"

Moscow, June 1–4, 2016

INFORMATION EXTRACTION BASED ON DEEP SYNTACTIC-SEMANTIC ANALYSIS

Stepanova M. E. (Maria_Ste@abbyy.com)[1],
Budnikov E. A. (Egor_B@abbyy.com)[1],
Chelombeeva A. N. (Antonina_C@abbyy.com)[1],
Matavina P. V. (Polina_Ma@abbyy.com)[1],
Skorinkin D. A. (Daniil_S@abbyy.com)[1,2]

[1]ABBYY, Moscow, Russia
[2]Higher School of Economics, Moscow, Russia

This paper presents a rule-based approach to Information Extraction (IE) task within FactRuEval-2016 competition. Our system is based on ABBYY Compreno Technology. The technology uses the results of deep syntactic-semantic analysis, which leads to significant reduction of the number of necessary rules and makes them laconic.
The evaluation was conducted on FactRuEval dataset. FactRuEval is an open evaluation of IE systems. The participants could take part in three tracks. The first track required to detect the boundaries and type of named entities in a text. The second track required to extract normalized attributes and perform local identification of named entities. The third track required to extract facts of certain types from a text. We took part in all three of the tracks with the nickname *violet*. Our method proved to be successful: we have achieved high F-measures in Named Entity Recognition tracks and the highest F-measure in Fact Extraction track.

Key words: information extraction, named entity recognition, syntactic analysis, anaphora and coreference resolution, fact extraction

ИЗВЛЕЧЕНИЕ ИНФОРМАЦИИ НА ОСНОВЕ ГЛУБОКОГО СИНТАКТИКО-СЕМАНТИЧЕСКОГО АНАЛИЗА

Степанова М. Е. (Maria_Ste@abbyy.com)[1],
Будников Е. А. (Egor_B@abbyy.com)[1],
Челомбеева А. Н. (Antonina_C@abbyy.com)[1],
Матавина П. В. (Polina_Ma@abbyy.com)[1],
Скоринкин Д. А. (Daniil_S@abbyy.com)[1, 2]

[1]ABBYY, Москва, Россия
[2]Высшая школа экономики, Москва, Россия

Ключевые слова: извлечение информации, распознавание именованных сущностей, синтаксический анализ, разрешение кореференции, извлечение фактов

1. Introduction

As the quantities of texts available in digital form increase rapidly, there is a growing need for Information Extraction (IE) systems to process them. In recent decades a number of competitions were held to assess performance of such systems for different languages. However, very few attempts were made to evaluate the state of the art for Russian language IE, and until recently no open-access corpora were available for such evaluation. FactRuEval-2016 was intended to amend the situation by running an independent contest between IE systems for Russian language and publishing a freely available corpus afterwards.

For those willing to participate, three independent tracks were provided. The first track tested the standard “baseline” named entity recognition (NER). Track participants were supposed to detect and annotate each entity and correctly attribute its type (Person, Organization, Location, LocOrg) without establishing any coreference chains. The second track involved local identification of entities and extraction of values for their predefined properties/attributes (e.g. name, surname, organization name). The third track evaluated fact extraction. Fact types were limited to four: occupation, business deal, ownership and meeting.

The Information Extraction system we describe in this paper is a model-based one, i.e. it combines rule-based approach, sophisticated language modelling and statistical parsing. The system took part in all three tracks with the results available below (see Results).

2. Related Work

One of the most well-known conferences that addressed the problem of IE was Message Understanding Conference (MUC). The term "Named Entity" was actually coined for the Sixth MUC (Grishman, Sundheim, 1996). ACE, TAC and CoNLL (Tjong Kim Sang, De Meulder, 2003) can be named among other conferences that addressed NER in their tasks. An overview of the main approaches to NER is given in (Nadeau, Sekine, 2007). An Overview of Event Extraction from Text is given in (Hogenboom et al., 2011).

Two main approaches to information extraction are classifier-based and pattern-based. A classifier-based system is, for example, ALICE (Chieu et al., 2003). WHISK (Soderland, 1999) and (Yakushiji et al., 2006) can be named as examples of pattern-based systems.

As for locality of the context being used for event extraction, usually event extraction systems rely on the local context around phrases that are considered as candidates for extraction. Some systems use extraction patterns (Soderland et al., 1995; Riloff, 1996; Yangarber et al., 2000; Califf and Mooney, 2003), which represent the immediate contexts surrounding candidate extractions. Similarly, classifier-based approaches (Freitag, 1998; Freitag and McCallum, 2000; Chieu et al., 2003; Bunescu and Mooney, 2004) rely on features in the immediate context of the candidate extractions.

Patwardhan and Riloff (Patwardhan and Riloff, 2009) introduced an event extraction model that consists of two parts: a sentential event recognition that determines if a sentence is discussing a domain-relevant event and a role fillers recognizing that identifies phrases as role fillers based upon the assumption that the surrounding context is discussing a relevant event. In our system we use similar approach when we determine events independently from named entities and look for the role fillers in specific semantic slots under a predicate.

Gareev et al. suggested quality baselines for the Russian NER task considering lack of works reporting results for the Russian language (Gareev et al., 2013). The authors also implemented and evaluated two approaches to NER: knowledge-based and statistical. Shelmanov et al. presented and evaluated the pipeline for processing of clinical notes in Russian (Shelmanov et al., 2015). In different tasks they used both rule-based patterns and several supervised machine-learning methods. In (Solovyev et al., 2012) the authors used extraction templates to extract events from texts in Russian. It was noted that information from different sources may be used for building those templates: intuition of the developer, examples from the texts, language model based on Chomsky grammars or other formal language models like in (Mel'čuk, I. A., 1973).

3. Method

3.1. Syntactic-semantic trees

Most of pattern-based systems rest on regular expression patterns. Often the words describing a specific type of event are grouped into semantic classes to reduce the final number of patterns. The change of the domain requires creation of new

patterns and semantic classes, which makes the development of such systems resource consuming. Furthermore, the syntactic variability may require creation of a great number of patterns. On the other hand, classifier-based systems tend to have lower recall due to the inability to take into account distant interdependencies and coreference.

To meet these challenges we use a syntactic-semantic parser that allows us to perform full syntactic-semantic analysis of a natural language text (Anisimovich et al., 2012). The syntactic-semantic analysis is based on multilevel natural language model created by linguists and then corpus-trained (Zuev et al., 2013). The output is a forest of dependency-based parse trees augmented with grammatical and semantic information. The trees can be viewed either as projective dependency tree or as constituent tree.

To perform semantic analysis the parser uses semantic hierarchy of language-independent "meanings" (semantic classes). The language-specific lexemes fit into semantic hierarchy as children of semantic classes. The parse tree nodes are augmented with semantic classes as well as the so called semantic slots (semantic roles), this information is language independent. Information on syntactic slots (extended analogue of syntactic functions) and non-tree links (conjunction, pronoun anaphora and other non-local dependencies) is also present. The latter is especially important for the needs of information extraction.

3.2. The information extraction mechanism

The input accepted by the information extraction mechanism is a sequence of syntactic-semantic trees. The output of the extraction mechanism is an RDF graph. The RDF data is consistent with an OWL-DL ontology (W3C, 2004) which is a predefined and static. Information about facts (i.e. situations and events) is modelled in a way that is ideologically similar to that proposed by the W3C consortium for defining N-ary relations (W3C, 2006)

The mechanism of information extraction is controlled by a system of production rules of three types

1. The rules of interpretation of syntactic-semantic structures
2. The rules of identification of information objects
3. Anaphoric rules

The rules are formulated on a special formal language, the syntax of which has the constructions to work with syntactic-semantic trees and information objects (fragments of RDF-graph). All the rules run "at the same time". This means that in the process of information extraction any rule, for which the input data is sufficient, may be applied at any moment.

The consistency of the extracted information is a built-in feature of the system. It is secured, firstly, by the extraction rules syntax and, secondly, by validation procedures that prevent generation of ontologically inconsistent data.

In addition to RDF graph, extraction mechanism generates annotations, i.e. the information that links extracted data to the respective parts of the original text. In this article we focus on the logical structure of the information extraction mechanism while some details about analysis algorithm may be found in (Starostin et al., 2014).

3.2.1. The rules of interpretation of syntactic-semantic structures

The rules of interpretation of syntactic-semantic structures (the interpretation rules) allow us to find subtrees inside syntactic-semantic trees that meet specific requirements. Each of these rules is a production, in the left part of which a pattern of a tree fragment is defined. In the right part of the production the statements about the information objects, induced by the left part, are grouped. An example of a rule, finding a mention of a person in a syntactic-semantic tree, looks as follows:

```
//Илья Петров пришел
this "PERSON_BY_FIRSTNAME"
[
    surname "PERSON_BY_LASTNAME"
]
=>
Person P(this),
anchor(P, surname, Coreferential),
annotation(P, this.core, surname.core),
P.firstname == Norm(this.core),
P.surname == Norm(surname.core);
```

Figure 1

The most important element of the syntax of syntactic-semantic structures interpretation rules is the construction anchor(...). Thanks to this construction the developer has the possibility to associate information objects with the constituents in a syntactic-semantic tree, which, consequently, allows to state the reference conditions (or object conditions) in the left parts of production rules.

The reference conditions allow to denote the constituents, with which other information objects (by means of other rules) have already been associated. In the following figure an example of a rule with reference condition is given. The rule states that a noun with the meaning “Generalized occupation” (semantic class “HUMAN”) refers to the same person that is syntactically dependent on it.

```
//студент Вася
head "HUMAN"
[
    (Classifier_Name|ClassifiedEntity|Specification): this <%BasicEntity:Person%>
]
=>
anchor(this.o,head, Coreferential);
```

Figure 2

The ability to establish multiple links between the world of objects and the text is as important as the ability to establish links between the words within a text. Thanks to this approach different mechanisms of coreference resolution have been organically integrated in our system.

Thus, for example, the system automatically treats the construction anchor(..) in such a way that an object O is automatically linked to the constituents, which are known (on the parser level) to be coreferent with a specific constituent X, while being linked to X. That is why other rules automatically start to “see” the object O on these constituents.

3.2.2. Identification rules

The rules of information object identification allow us to conclude that two information objects, originally considered to be separate, are in fact one and the same object. In contrast to the interpretation rules, these rules are not based on parse trees. The left part of an identification rule contains patterns, describing two information objects (fragments of an RDF graph) and special conditions applied to both objects (for example, that a specific attribute must have the same value). The right part of the rule is empty as the only thing the rule does is merging two objects in one. In the next figure an example of an identification rule merging two organizations into one is given (the reason of merging is the same value of company_by_name attribute).

```
<% Organization, company_by_name ~= null%>
<% Organization, company_by_name ~= null%>

intersects( company_by_name )
```

Figure 3

It is important to note that the constituents to which the information objects are linked are merged along with the merge of the information objects. After that the other rules get the access to the result of the identification process through the united set of constituents, and that, consequently, can lead to the extraction of new information.

We have to leave out the detailed description of anaphoric rules due to limitations on the size of the paper. For information on anaphora and coreference resolution in our system see (Bogdanov et al., 2014).

3.3. Fact Extraction

3.3.1. General Idea

Given the limits of this paper, we chose not to describe the entity extraction module of our system in details. For further information on that please refer to (Starostin et al., 2014). From this point on we will put our focus entirely on fact extraction.

The task of extraction of facts from texts differs considerably from that of named entity recognition. First of all, facts in texts are often expressed non-locally—the information making up a fact (filling specific attributes) is often contained within several sentences. It is possible because natural language provides a variety of instruments to denote coreference—from a simple repeated reference to an object by name to complex types of anaphora. Different combinations of all these instruments can be used to express facts.

It is obvious that a system that solves fact extraction problem efficiently must include coreference resolution mechanisms (Bogdanov et al., 2014). Moreover, the integration of different mechanisms within a single system is a considerable problem. It is well-known that simple architectures with sequential launch of modules (Karasev et al., 2004), responsible for processing of different phenomena of natural language, are confronted with the situation when the information to be generated at late stages is required at the early stages.

The process of fact extraction corresponds to the one described in (Ahn, 2006). There are certain differences however due to the fact that we have a trustworthy semantic parser available.

We define the following stages of fact extraction.

1. Fact occurrence identification. We look for the core of a fact occurrence (in most cases it is a predicate) and create an individual of the fact concept there.
2. Identification of attribute values around the fact core. As our parser provides us with the semantic roles of the words it is unnecessary to differentiate between identification of the attribute values and definition of the attribute type.
   We run our named entity recognition module before launching the fact extraction module. Let us note that the availability of the semantic roles allows us to assign attribute values other than the ontology predefined named entities.
3. Entity and fact coreference resolution.
4. Validation of facts. We filter out the facts that do not have certain attributes filled.

3.4. Specific facts

3.4.1. Occupation

Occupation is the fact that defines personal employment. The key objectives were to identify the employee (most often a person) and the employer (usually an organization or a LocOrg). Both slots were declared mandatory, so should any of them have been missing, the fact was to be discarded. In addition there were two optional properties named position and phase (initiation/termination).

Consider a simplified example of an extraction rule with a single production:

```
    this "TO_WORK" // check if the tree node belongs to "TO_WORK" semantic class, which
combines all sorts of work predicates
        [Agent: person <%BasicEntity:Person%>] // check if "this" has a child with Agent
semantic slot and a Person entity attached to it;
        [(Object|Locative): org <%Org:Organization%>] // check if "this" has a child with
Object or Locative semantic slot and an Organization entity attached to it;
    =>
    BasicFact:Occupation Occ(this); // create Occupation fact
```

Figure 4

It is important to point out that the general model of Occupation in our system is somewhat different from that proposed by the FactRuEval organizers. For instance, we may extract Occupation with no employer, e.g. ‘president Barack Obama’. Our system is also capable of extracting implicit positions, for example, in ‘Microsoft is led by Bill Gates’ we will infer that Bill Gates’ position in Microsoft is ‘(the) head’. Such extra-capabilities were disabled to comply with the competition rules.

3.4.2. Ownership

Ownership generally defines a fact of possession of something (property) by someone (owner). To extract the fact we rely heavily on the ‘Possessor’ semantic slot ([X]

owns Y, Y belongs [to X], [X] bought Y, Y has been acquired [by X], Y was sold [to X], the purchase of Y [by X] etc.). We also make use of semantic classes with 'possessive' meanings and of other elements of our extensive language model.

However, the rules of the competition limited the fact to cases where the property is an Organization and the owner is either a Person or an Organization. To comply with the rules we had to impose certain constraints on the existing rules for Ownership extraction. Some additions were also made for situations where a person was named a (co-)founder of an organization, although the actual *possession* of a company is disputable in this case. Another case that demanded specific adjustments was the ownership of shares.

3.4.3. Deal

Unlike Occupation and Ownership, a Business Deal was not part of the standard set of facts that our system extracts. It turned out, however, that we can cover many cases by simply generalizing several existing facts, Purchase and Transfer above all. This saved us from having to build the extraction library entirely from scratch. Such types of deal as 'loan' and 'investment', on the other hand, required building entirely new rules.

The instances of Purchase fact were converted to Deal if there were Persons or organizations among buyers, sellers and objects of sale. Transfer was specified to Deal only in cases where the object of transfer was a relevant one (e.g. a sum of money). We also considered an organization to be a participant of the deal if its representative person (e.g. an employee or an owner) was already identified as a participant.

The type of the deal was most often determined according to the semantic class of the root node (usually a predicate like 'to buy', 'to invest', 'to be indebted' etc). Sometimes, however, we had to go deeper and examine object under the predicate ('strike [a bargain]', 'sign [a contract]'). We would also like to point out that the test collection contained certain arguable examples of deal types that were not represented in the training set—e.g. economic sanctions, embargoes, fines etc.

4. Dataset

The competition data set was prepared by Opencorpora.org (Bocharov et al.). The document collection consisted of news and analytical articles provided by Wikinews and Chaskor news sites. The markup for track 1 was crowdsourced to the web, tracks 2 and 3 were prepared by FactRuEval organizing committee. The training and the test sets contained 122 and 135 texts respectively.

5. Results

5.1. Entities

The results provided by the organizers show that our system tends to do its best when it comes to Person extraction. On Track 1 we achieved 93% F-measure for Persons and 78,4% for Organizations (Table 1).

Table 1. Track 1 results (test set)

Type	P	R	F1	TP1	TP2	In Std.	In Test.
Per	0.9450	0.9155	0.9300	1,233.18	1,233.18	1,347	1,305
Loc	0.5168	0.8596	0.6455	515.76	515.76	600	998
Org	0.8162	0.7551	0.7844	1,160.57	1,160.57	1,537	1,422
locorg	0.8864	0.3122	0.4618	214.50	214.50	687	242
overall	**0.7875**	**0.7490**	**0.7678**	3,124.01	3,124.01	4,171	3,967

Table 2. Track 1 results without Loc/LocOrg division (test set)

Type	P	R	F1	TP1	TP2	In Std.	In Test.
per	0.9450	0.9155	0.9300	1,233.18	1,233.18	1,347	1,305
loc	0.9261	0.8698	0.8971	1,116.83	1,116.83	1,284	1,206
org	0.8175	0.7564	0.7858	1,162.56	1,162.56	1,537	1,422
overall	**0.8931**	**0.8427**	**0.8672**	3,512.57	3,512.57	4,168	3,933

Locations clearly fell victim to the Location/LocOrg split, which was quite hard to formalize and implement to begin with. As can be seen from the table, the relatively moderate quality of the Location extraction is mainly due to a lot of false positives, and this drop in precision corresponds to LocOrg's drop in recall. To put it simply, the organizers' understanding of when a Location becomes a LocOrg was apparently much broader than ours. There was a mode of comparison without Loc/LocOrg division (Table 2), where this split was removed. Our system, actually, ranked 1 in this mode of comparison.

Table 3 shows the results of Track 2. Note that in this case all results fall into the same precision-better-than-recall pattern typical of rule-based systems.

Table 3. Track 2 results (test set)

Type	P	R	F1	TP1	TP2	In Std.	In Test.
Per	0.8817	0.8592	0.8703	538.73	538.73	627	611
Loc	0.8430	0.7942	0.8179	494.00	494.00	622	586
Org	0.6823	0.6763	0.6793	547.17	547.17	809	802
overall	**0.7903**	**0.7677**	**0.7789**	1,579.90	1,579.90	2,058	1,999

We also noted that our system tends to demonstrate marginal or no decline in quality when we shift to the test set from the training one. For Track 1 the overall F-measure actually even went up on the test set (from 75.7% to 76.8%), and for Track 2 the drop was not dramatic (from 83.2% to 77.9%).

5.2. Facts

Facts have a more sophisticated and variable structure than entities, they are much more syntax-dependent and tend to 'spread' across large sections of text (at times much larger than a single sentence). This makes it hard to detect every participant of a single fact and account for all possible patterns and paraphrases. The results of the third track show that while we were able to achieve good precision for most facts, there is much room for improvement when it comes to recall.

Table 3. Track 3 results (test set)

TAG	P	R	F1	TP1	TP2	In Std.	In Test.
ownership	0.5379	0.1709	0.2594	24.10	26.90	141	50
occupation	0.8058	0.5679	0.6662	190.80	195.81	336	243
meeting	0.8690	0.1352	0.2340	6.08	6.08	45	7
Deal	0.6777	0.1932	0.3007	19.71	27.11	102	40
overall	**0.7526**	**0.3857**	**0.5100**	240.69	255.89	624	340

To our knowledge, these were the best results for Track 3. However, it should be noted that only two systems took part in track 3.

6. Conclusion

FactRuEval-2016 allowed us to evaluate the performance of our Information Extraction system in a competitive environment, and we are grateful to the organizing committee for this opportunity.

The results for entity extraction show that our system has a slight bias for precision over recall, which is typical for rule/pattern-based approaches. Overall, we were able to achieve good quality (especially with Persons) comparable with the results of the best machine learning systems. Moreover, our approach proved to be quite stable, showing little or no decline in F-measure

Fact extraction results turned out to be even more precision-oriented: the system returns few incorrect fact, but also misses a fair share of correct ones. However, the use of syntactic-semantic trees helped us create concise rules that covered a big subset of possible patterns for the four facts chosen by the organizing committee. And as facts depend heavily on syntactic structures, we faced very little competition from the machine learning based systems.

Analysis of mistakes suggests that further work should be concentrated on sophisticated coreference resolution (Vladimir Putin—president—leader—head). Some facts from the test set clearly required the system to make use of document-level information and pragmatics of the text, like in cases when a president or prime minister strikes a deal, and the country(s) he implicitly represents is listed as a participant.

References

1. *Ahn D.* (2006), The stages of event extraction, Proceedings of the Workshop on Annotating and Reasoning about Time and Events, Association for Computational Linguistics, pp. 1–8.
2. *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: Proceedings of the International Conference "Dialog" [Komp'iuternaia Lingvistika i Intellektual'nye Tehnologii: Trudy Mezhdunarodnoj Konferentsii "Dialog"], Bekasovo, pp. 90–103.
3. *Bocharov V., Bichineva S., Granovsky D., Ostapuk N., Stepanova M.* (2011), Quality assurance tools in the OpenCorpora project, Computational Linguistics and Intellectual Technologies: Proceedings of the International Conference "Dialog" [Komp'iuternaia Lingvistika i Intellektual'nye Tehnologii: Trudy Mezhdunarodnoj Konferentsii "Dialog"], Bekasovo, Vol. 10.
4. *Bogdanov A. V., Dzhumaev S. S., Skorinkin D. A., Starostin A. S.* (2014), Anaphora Analysis based on ABBYY Compreno Linguistic Technologies, Computational Linguistics and Intellectual Technologies: Papers from the Annual International Conference "Dialogue" (2014), pp. 659–667.
5. *Bunescu R., Mooney R. J.* (2004), Collective information extraction with relational Markov networks, Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics, Association for Computational Linguistics, p. 438.
6. *Califf M. E., Mooney R. J.* (2003), Bottom-up relational learning of pattern matching rules for information extraction, The Journal of Machine Learning Research, Vol. 4, pp. 177–210.
7. *Chieu H. L., Ng H. T., Lee Y. K.* (2003), Closing the gap: Learning-based information extraction rivaling knowledge-engineering methods, Proceedings of the 41st Annual Meeting on Association for Computational Linguistics, Association for Computational Linguistics, Vol. 1, pp. 216–223.
8. *Freitag D.* (1998), Toward general-purpose learning for information extraction, Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics, Association for Computational Linguistics, Vol. 1, pp. 404–408.
9. *Freitag D., McCallum A.* (2000), Information extraction with HMM structures learned by stochastic optimization, AAAI/IAAI, pp. 584–589.
10. *Gareev R., Tkachenko M., Solovyev V., Simanovsky A., Ivanov V.* (2013), Introducing baselines for Russian named entity recognition, Computational Linguistics and Intelligent Text Processing, Springer Berlin Heidelberg, pp. 329–342.
11. *Grishman R., Sundheim B.* (1996), Message Understanding Conference-6: A Brief History, COLING, Vol. 96, pp. 466–471.
12. *Hogenboom F., Frasincar F., Kaymak U., De Jong F.* (2011), An overview of event extraction from text, Workshop on Detection, Representation, and Exploitation of Events in the Semantic Web (DeRiVE 2011) at Tenth International Semantic Web Conference (ISWC 2011), Vol. 779, pp. 48–57.

13. *Karasev V., Khoroshevsky V., Shafirin A.* (2004), New Flexible KRL JAPE+: Development & Implementation, Knowledge-Based Software Engineering: Proceedings of the Sixth Joint Conference on Knowledge-Based Software Engineering, IOS Press, Vol. 108, p. 217.
14. *Mel'čuk I. A.* (1973), Towards a Linguistic 'Meaning⇔Text' Model, Trends in Soviet theoretical linguistics, Springer Netherlands, pp. 33–57.
15. *Nadeau D., Sekine S.* (2007), A survey of named entity recognition and classification, Lingvisticae Investigationes, Vol. 30, pp. 3–26.
16. *Patwardhan S., Riloff E.* (2009), A unified model of phrasal and sentential evidence for information extraction, Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Vol. 1, pp. 151–160.
17. *Riloff E.* (1996), Automatically generating extraction patterns from untagged text, Proceedings of the national conference on artificial intelligence, pp. 1044–1049.
18. *Shelmanov A., Smirnov I., Vishneva R.* (2015), Information Extraction from Clinical Texts in Russian, Proceedings of the International Conference "Dialog" [Komp'iuternaia Lingvistika i Intellektual'nye Tehnologii: Trudy Mezhdunarodnoj Konferentsii "Dialog"].
19. *Soderland S.* (1999), Learning information extraction rules for semi-structured and free text, Machine learning, Vol. 34, pp. 233–272.
20. *Soderland S., Fisher D., Aseltine J., Lehnert W.* (1995), CRYSTAL: Inducing a conceptual dictionary, arXiv preprint cmp-lg/9505020.
21. *Solovyev V., Ivanov V., Gareev R., Serebryakov S., Vassilieva N.* (2012), Methodology for Building Extraction Templates for Russian Language in Knowledge-Based IE Systems, HP Laboratories Technical report, HPL-2012–211.
22. *Starostin A. S., Smurov I. M., Stepanova M. E.* (2014), A Production System for Information Extraction Based on complete syntactic-semantic analysis, Computational Linguistics and Intellectual Technologies: Papers from the Annual International Conference "Dialogue" (2014), pp. 89–101.
23. *Tjong Kim Sang E. F., De Meulder F.* (2003), Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition, Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003, Association for Computational Linguistics, Vol. 4, pp. 142–147.
24. *W3C* (2004), OWL Web Ontology Language Overview, available at: http://www.w3.org/TR/2004/REC-owl-features-20040210
25. *W3C* (2006), Defining N-ary Relations on the Semantic Web, available at: http://www.w3.org/TR/swbp-n-aryRelations
26. *Yakushiji A., Miyao Y., Ohta T., Tateisi Y., Tsujii J. I.* (2006), Automatic construction of predicate-argument structure patterns for biomedical information extraction, Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, pp. 284–292.
27. *Yangarber R., Grishman R., Tapanainen P., Huttunen S.* (2000), Automatic acquisition of domain knowledge for information extraction, Proceedings of the 18th conference on Computational linguistics, Association for Computational Linguistics, Vol. 2, pp. 940–946.

28. *Zuev K. A., Indenbom M. E., Judina M. V.* (2013), Statistical machine translation with linguistic language model, Computational Linguistics and Intellectual Technologies: Proceedings of the International Conference "Dialog" [Komp'iuternaia Lingvistika i Intellektual'nye Tehnologii: Trudy Mezhdunarodnoj Konferentsii "Dialog"], Bekasovo, Vol. 2, pp. 164–172.