computing@computingonline.net
www.computingonline.net

Print ISSN 1727-6209
On-line ISSN 2312-5381
International Journal of Computing

A METHOD FOR AUTOMATIC TEXT SUMMARIZATION BASED ON RHETORICAL ANALYSIS AND TOPIC MODELING

Tatiana Batura [1,2], Aigerim Bakiyeva [3], Maria Charintseva [2]

[1] A.P. Ershov Institute of Informatics Systems (IIS) SB RAS, 6 Acad. Lavrentiev Ave., Novosibirsk, 630090, Russia,
[2] Novosibirsk State University, 1, Pirogov Str., Novosibirsk, 630090, Russia,
e-mail: tatiana.v.batura@gmail.com, m.charintseva@g.nsu.ru
[3] L.N. Gumilyov Eurasian National University, 2 Satbaev Str., Nur-Sultan 010008, Kazakhstan,
e-mail: m_aigerim0707@mail.ru

Paper history:
Received 16 October 2018
Received in revised form 18 October 2019
Accepted 21 February 2020
Available online 30 March 2020

Keywords:
natural language processing;
automatic summarization;
rhetorical structure theory;
discourse markers;
additive regularization;
topic modeling.

Abstract: This article describes the original method of automatic summarization of scientific and technical texts based on rhetorical analysis and using topic modeling. The proposed method combines the use of a linguistic knowledge base and machine learning. For the detection of key terms, we used topic modeling. First, unigram topic models containing only one-word terms are constructed. Further, these models are extended by adding multiword terms. The most significant fragments of the original document are determined in the process of rhetorical analysis with the help of discursive markers. When evaluating the importance of text fragments, keywords, multiword terms, and scientific lexicon characterizing scientific and technical texts are also taken into account. A linguistic knowledge base has been created to store information about the markers and scientific lexicon. The experiments showed that this method is effective, needs a comparatively small amount of training data and can be adapted to processing texts of different subject fields in other languages.

Copyright © Research Institute for Intelligent Computer Systems, 2020.
All rights reserved.

1. INTRODUCTION

Due to a rapid increase in the bulk of textual information in the Internet, active research in the field of computer linguistics remains to be highly demanded. Among other tasks, automatic summarization also plays an important role.

There are many ways to solve this problem, which are quite clearly divided into three areas: extraction, abstraction and a hybrid approach. *Extraction* is an action of taking out the most informative sentences from the source text. This method is sometimes called a superficial one. The advantages of extracting methods include independence from the subject field, as well as the comparative simplicity of development: it does not require the creation of extensive knowledge bases or a detailed linguistic analysis of the text. The disadvantages of extracting methods include the fact that the obtained summaries are often incoherent. *Summarization* is a generation of a summary that takes into account morphology, syntax, semantics, due to which a coherent text is formed. This method is called a deep one. The advantages of abstracting methods are in obtaining a summary of a higher quality than when using extracting methods. The disadvantages of these methods include the difficulty of their practical implementation and the need to collect a large amount of linguistic knowledge.

In order to overcome the shortcomings of abstracting and extracting methods, hybrid methods that combine the sides of the above approaches are developed. For example, first, the most significant fragments are extracted and their subsequent processing is performed, then sentences are merged, uninformative parts are deleted, etc. The difficulty in developing hybrid methods lies in choosing the most successful combination of generation and extraction techniques. Hybrid methods compared to abstract methods are easier to develop, and compared to purely extracted methods, they can provide a better quality of the output.

For instance, in the COMPENDIUM system [1], the hybrid approach is implemented as follows: an abstract drawn up by the extraction method is fed to the input. For this abstract, a weighted graph is constructed, the vertices of which are represented by words, and the edges reflect the adjacency relation between words. The weight of the arcs is determined by the PageRank algorithm. Then, the shortest path is constructed between the vertices of the graph using Dijkstra's algorithm. Thus, a set of candidate sentences is formed. The next step is to filter the wrong paths. The authors identified the following criteria for correct sentences: the sentence must be at least three words long; each sentence must have a verb; the sentence must not end with the article, preposition, pronoun or conjunction. At the last stage, there is a selection of sentences for inclusion in a new abstract from an abstract compiled by the extraction methodology or from a set of candidate sentences.

An automated multilingual text summarization system called SUMMARIST is described in [2]. This system combines symbolic concept-level world knowledge with information retrieval and statistical techniques. The algorithm consists of three steps: topic identification, interpretation, and generation. SUMMARIST produces extracted summaries in five languages: English, Japanese, Spanish, Indonesian, and Arabic.

There is also a hybrid SumUM system [3], which generates summaries for scientific and technical documents. The authors conducted a study of the corpus of summaries written by people and revealed a number of transformations that referents used, for example, merging information from different parts of a document, paraphrasing the original.

The authors' approach [4] to abstracting is based on a superficial analysis of the source document, extracting information of a certain kind, and text generation. The system also uses a parts-of-speech tagger; linguistic and conceptual patterns defined by regular expressions; syntactic categories; a conceptual dictionary.

In [5], a summarization method based on the conversion of text into concepts with the subsequent representation of the document in the form of a graph is proposed. The method uses additional resources – the English-language biomedical thesaurus UMLS [6] and the MetaMap tool [7] for converting text into concepts from this thesaurus. The method consists of the following steps: representation of the document in the form of a graph, clustering of concepts, sentence selection. In such graph, the nodes are concepts of the UMLS thesaurus and the edges indicate the relations between the nodes. To do this, all document sentences are processed by MetaMap; UMLS concepts are complemented by their hypernyms. Next, each node is assigned a rating directly proportional to the depth of the hierarchy of concepts. After that, all sentence graphs are combined into one text graph. Then concept clustering is performed. Each cluster is a set of concepts that are close in meaning and can be considered as the theme of the document. The sentence selection process is based on the similarity between clusters and sentences. The authors use several heuristics to select sentences.

Natural language is very difficult for automatic processing. Therefore, researchers tend to solve abstracting problems for certain subject areas to improve the quality of the obtained results. The authors of [8] investigate the summarization problem for texts of court decisions. The processing of such texts is also the subject of [9, 10]. The authors of [11] propose an approach to summarization of reviews or comments of Internet users. They put together a corpus of user ratings from reviews about cell phones and cars in English on Amazon.com, WhatCar.com and the social network Twitter. This corpus was marked up by an expert who determined the tonality of the comment (negative, neutral, positive) and the rating intensity. The authors of [12] propose a hybrid approach to the summarization of patent texts in English, French, and German.

Our method proposed in this paper is hybrid. The discursive analysis of the text was taken as a basis. All experiments were conducted with scientific and technical texts in Russian.

The attempts of applying discursive analysis to solving various tasks of computer linguistics can be found in current practice. A detailed review of the literature reveals that, in most cases, discursive analysis can enhance the quality of automatic systems, depending on the specific task.

The RST approach has been widely used. In [13], RST was applied to identify important units in a document. The author proposed to use a constraint satisfaction algorithm to assemble all the trees that organize the input text, and then employed several heuristics to prefer one tree to the others.

Some authors [14] consider the summarization problem as a reduction of data, namely, the original document is considered as a high-dimensional data, and the summarization task is to reduce the dimension of the document and keep the main content of it.

An RST-based summarization system for scientific articles, identifying seven rhetorical categories, is described in [15]. Structural analysis formed the basis of sentence weights [16]; the author applied RST to create a graph representation of a document from which a query-based summarization

was produced. Building a generic and extractive summary for a single document that includes information from its neighboring documents is discussed in [17]. The produced summary has sentences extracted from the single document and makes use of the additional knowledge from its multi documents.

S. Mithun describes a schema-based summarization approach for query-based blog summaries that utilizes discourse structures [18]. This approach performs four main tasks, namely: question categorization, identification of rhetorical predicates, schema selection, and summary generation. The author built a system named BlogSum and evaluated its performance for question relevance and coherence. The achieved results show that the proposed approach can effectively reduce question irrelevance and discourse incoherence of automatic summaries.

Research in this field for the English language has reached a sufficiently high level, but there is not enough research for Russian. The summarization problem in scientific and technical texts in Russian using similar approaches was stated by scientists in [19] and [20]. The research [19] describes the methods and algorithms that take into account the nonlinear and hierarchical nature of the text. With the help of rhetorical relations, the problem of extraction is solved. The author has developed a system based on inference rules and a highly specialized dictionary of key phrases. A hybrid approach is proposed in [20], combining extraction and abstraction methods. This approach was implemented by the author in a summarization system focused on automatic translation. The described system is constructed for texts on mathematical modeling. In our work, we solve a broader problem; we analyze any scientific texts (articles, theses, reports) on any topics. In addition, we solve also the topic detection problem and the problem of searching for keywords and multiword expressions.

Our work describes an approach that allows forming brief abstracts of scientific and technical texts and determining their topics. The proposed method forms an abstract based on the most essential sentences of an original document. The importance of a sentence is determined in the process of rhetorical analysis. In its turn, the method of additive regularization of topic models (ARTM) is applied to determine the themes of the texts. ARTM [21] allows one to solve the problems of non-uniqueness and instability by introducing additional restrictions on the required solution. The following methods of smoothing and sparsing out the distribution of terms in topics, topics in documents, and others can be utilized as regularizers.

2. MAIN FEATURES OF RHETORICAL RELATIONS

The Rhetorical Structure Theory is one of the most widely used theories of texts organization [22]. According to it, initially, a text is divided into non-overlapping fragments, namely, elementary discursive units (EDUs).

Further, sequential EDUs are interconnected by rhetorical relations. These parts are known as elements from which larger fragments of texts and whole texts are built. Each fragment, in relation to other fragments, performs its certain role. Textual connectivity is formed by means of those relations that are modeled between fragments within the text [23].

In the theory of rhetorical structures, two types of EDUs can be defined. One of them, a nucleus, is considered to be the most important part of an utterance, while the latter one, called a satellite, explains the nucleus and is considered to be secondary. The nucleus contains the basic information, whereas the satellite keeps additional information about the nucleus. The satellite is often incomprehensible without the nucleus. Meanwhile, the expressions where the satellites were deleted can be understood to a certain extent.

Consider the following example:

Text: *There may be some regularity in the target function,* ***besides*** *being smooth.*

Marker: *besides*

Relation name: Elaboration

The following notation is provided for convenience in the example below.

Suppose, x is a nucleus; y is a marker; z is a satellite;

$S(x)$ is a predicate for EDU, which is a nucleus;

$S'(x)$ is a predicate for EDU (which is a nucleus) beginning with a capital letter;

$S(z)$ is a predicate for EDU, which is a satellite;

$S'(z)$ is a predicate for EDU (which is a satellite) beginning with a capital letter;

y' is a marker beginning with a capital letter;

$p(\ )$ is a punctuation character, the argument can be ".", ",", ":", ";".

Now the provided example can be represented as a formula of the predicate calculus:

$$S'(x) \wedge p(,) \wedge y \wedge S(z) \wedge p(.). \quad (1)$$

3. FORMAL DESCRIPTION OF TEXT TRANSFORMATION

Initially, the nucleus EDUs are essential to be found in the text in order to obtain a short abstract

automatically. The step after that is transformation of the statements containing these nucleus EDUs, so that the text of the resulting abstract turns out to be connected. Depending on different markers and discursive relations, these transformations will be different. Further, some of the considered transformations are provided. For a formal description of the actions performed by the system, it was decided to use the predicate logic of the first and second orders.

3.1 FIRST-ORDER PREDICATES

According to the notations introduced in the previous section, the actions performed by the system can be described as follows.

In the example from the previous section with the marker y = *'besides'*, it is necessary to delete the satellite along with the marker and leave the previous clause, which is nucleus EDU that can be illustrated as:

$$S'(x) \wedge p(,) \wedge y \wedge S(z) \wedge p(.) \rightarrow S'(x) \wedge p(.) \wedge \neg\left(y \wedge S(z) \wedge p(.)\right). \quad (2)$$

3.1.1 ACTIONS

In the proposed approach, a rhetorical analysis is used at the stage of forming a quasi-abstract. A quasi-abstract is a list of the most significant sentences of a text. Simplified, this stage can be described as follows. First, we find nuclear EDU in the text. Further, statements containing these EDUs should be transformed so that an abridged text, which is intermediate between the original text and the final summary, is obtained. Discursive markers are used to define EDU boundaries.

Markers (Discourse markers) are words or phrases that do not have any real lexical meaning. They have an important function to form the structure of a text. They are used to connect, organize and manage the authors' intentions. Markers provide inter-phrase connection. Table 1 shows the actions for markers, that helps to form a quasi-abstract.

Table 1. Actions of markers

Rhetorical relations	Markers	Actions
Cause-Effect	*'therefore'*	keep_remove
Contrast	*'however'*	remove_keep
Elaboration	*'besides'*	remove_keep
Evidence	*'in this way'*	keep_remove
Restatement	*'in other words'*	remove_keep

During the research, we created a linguistic knowledge base consisting of 121 markers, 120 nouns and 108 verbs with weights that are often found in scientific and technical texts. In total, eight actions were considered. Some actions are explained below.

remove_ keep: This action removes the forthcoming clause and keeps the clause with the given marker.

keep_remove: This action keeps the preceding clause and removes the clause with the given marker.

3.2. SECOND-ORDER PREDICATES

The cases of nested EDUs, when lower-level EDUs are embedded in higher-level EDUs, are more convenient to describe using second-order predicates. Moreover, a separate predicate is introduced for each marker. To illustrate how the text is transformed in the cases of nested EDUs, the following example is provided.

"Most software implementations need to support operations that can return more than one tensor. ***For example****, if we wish to compute both the maximum value in a tensor and the index of that value, it is the best to compute both in a single pass through memory,* ***so*** *it is the most efficient to implement this procedure as a single operation with two outputs".*

In order to write down our example in a formal form, we add the following notation.

Suppose m is a nucleus in a dependent clause; n is a satellite in a dependent clause;

$S(m)$ is a predicate for EDU, which is a nucleus in a dependent clause;

$S'(m)$ is a predicate for EDU (which is a nucleus in a dependent clause) beginning with a capital letter;

$S(n)$ is a predicate for EDU, which is a satellite in a dependent clause;

$S'(n)$ is a predicate for EDU (which is a satellite in a dependent clause) beginning with a capital letter;

y_i are markers.

$$S'(x) \wedge p(.) \wedge S'\left(y_1 \wedge S(n) \wedge p(,) \wedge y_2 \wedge S(m)\right) \wedge p(.) \rightarrow S'(x) \wedge p(.) \wedge \neg S'\left(y_1 \wedge S(n) \wedge p(,) \wedge y_2\right) \wedge S'\left(S(m)\right) \wedge p(.),$$

where y_1 = *"for example"*; y_2 = *"so"*.

As a result, we will get the following text: *"Most software implementations need to support operations that can return more than one tensor. It*

is the most efficient to implement this procedure as a single operation with two outputs".

It should be noted that the use of first and second order formalisms for this purpose has not yet been sufficiently investigated. In the future, it may be necessary to extend it to take into account the order of the elements in the text and the order of transformations.

4. GENERAL DESCRIPTION OF THE SYSTEM

Let T be the text of the article cleared after preprocessing. It consists of sentences $T = [s_1,...,s_P]$.

The main goal of text summarization task is to find the transformation of the text T into a summary $\check{T}$, such that $\Psi: T \rightarrow \check{T}$, $|\check{T}| < |T|$, $|\check{T}| \approx 250$ words.

The main steps of our algorithm are described below.

1. Preprocessing. At the preprocessing stage, all images, tables, sentences with formulas, information about authors and bibliographic references were deleted from the source text. The author's abstracts were cut and saved separately so that we could evaluate the system afterwards, by means of comparing the result with the original abstract.

2. Building topic models, extracting keywords and multiword expressions. Topic modeling consists of building a model of a collection of text documents. In such a model, each topic is represented by a discrete probability distribution of words, and documents are represented by a discrete probability distribution of topics.

In other words, *a topic* is a set of words that describe a subject area. *A topic model* is a set of topics. Topics are not known in advance, they are discovered during the work of the probabilistic algorithm. This algorithm allows us to find the Phi and Theta matrices from the input plain text (see Fig. 1).

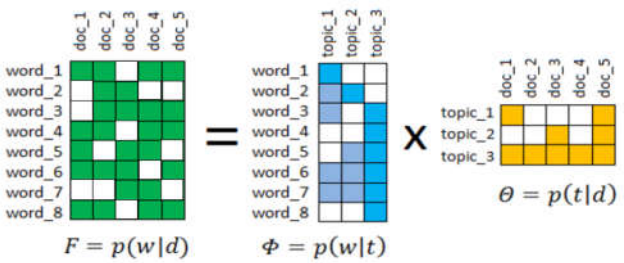

Figure 1 – Building a topic model

A unigram topic model is a model in which topics are described with one-word terms. However, sometimes we use expressions instead of single words. *A multi-word expression (MWE)* is a stable combination of several words. For example, *'linear equation system'*, *'image processing'*, *'machine learning'*, etc. *An extended topic model* is a model in which topics are described not only with one-word terms, but also with multi-word expressions. Schematically, these concepts are presented in Fig. 2.

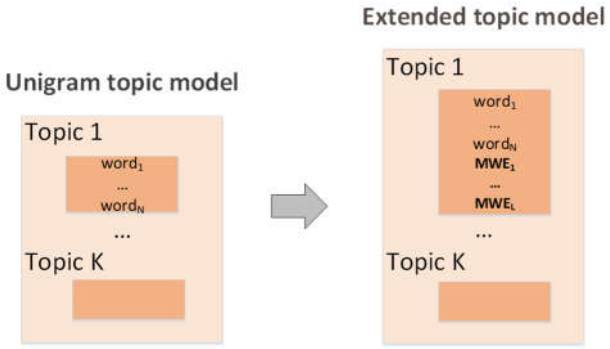

Figure 2 – Unigram and extended topic models

Currently, there are different methods of topic modeling, such as PLSA, LDA, ARTM. The main advantage of topic models in comparison with neural networks is that they are easy to interpret; the user understands the reasons for finding certain topics in a text and the structure of the topics themselves. In addition, it is often required that topic models take into account a heterogeneous data, identify the dynamics of topics in time, automatically separate topics into sub-topics, use not only one-word keywords but also multiword terms, etc.

To select the algorithm of topic modeling, we performed a number of experiments and decided to use the ARTM algorithm in the implementation of the BigARTM library [23]. Due to its versatility and flexibility in parameter settings, ARTM allows you to combine regularizers, thereby combining topic models. This method guarantees the uniqueness and stability of the solution. ARTM does not see an increase in the number of model parameters with an increase in the number of documents, so it can be applied to large sets of data.

Initially, a unigram model of the text is built; then the model expands with multiword expressions. The modification we proposed allows us to use not

only single-word terms, but also multiword expressions, which, in our opinion, increases the interpretability of the model. The algorithm of building extended topic models is described below.

1. Lemmatization.
2. Building a morphological dictionary.
3. Extraction of *n*-grams of words. To extract MWEs from the texts, the RAKE algorithm [24] is used. We adapted this algorithm for processing the texts in Russian and considered *n*-grams for *n* no more than five.
4. Grammatical agreement. Since the MWEs consist of word stems and not of words in the grammatical agreement, we need to perform a backward operation of transforming those stems into consistent phrases.
5. Building unigram topic models.
6. Building extended topic models.
7. Building a dictionary with weights for ranking topics.
8. Distribution of topics and key terms by document.

An example of one of extended models we built for the document with a title "*Algorithm for detecting objects in photographs with low image quality*" is as follows.

Topic: ['method', 'data', 'algorithm', 'classification', 'image', 'quality', 'learning', 'data set', 'parameter value', 'feature set', 'learning process', 'classification method', 'model building', 'classification task', 'classification quality', 'image classification', 'image classification quality'].

3. Rhetorical analysis and text transformation. At this step, we find sentences containing the discursive markers. To these sentences, certain actions are applied (see Section 3 for detailed information). As a result, we obtain a quasi-abstract. A quasi-abstract is a list of the most important sentences (or its fragments) in the text: $T' = [s'_1,...,s'_{P_1}], T' \subset T$.

In fact, a quasi-abstract does not consist of sentences in the usual sense, but of some fragments representing EDUs. However, to simplify further discussion, when it comes to a quasi-abstract, we will use the term "sentence".

4. Evaluation of sentence weights. The weight of each sentence of the quasi-abstract is calculated depending on whether it contains keywords (or multiword terms), discourse markers, and some special lexicon that are often found in scientific and technical texts. As a result, the weight of each sentence is calculated by the following formula:

$$SW(s') = \frac{1}{L} \cdot \sum_{i=1}^{L} w_i + \frac{1}{M} \cdot \sum_{j=1}^{M} v_j + \frac{1}{N} \cdot \sum_{k=1}^{N} d_k , \quad (3)$$

where $W = \{w_1, w_2, ..., w_L\}$ is a set of weights of keywords and multiword expressions ($|W| = L$). The weight w_i is defined as the frequency of the keyword (or the multiword expression) in the text;

$V = \{v_1, v_2, ..., v_M\}$ is a set of weights of significant verbs and nouns that are often found in scientific and technical texts ($|V| = M$). The weight v_k is determined using a linguistic knowledge base;

$D = \{d_1, d_2, ..., d_N\}$ is a set of weights of discursive markers ($|D| = N$). The weight d_j is determined using a linguistic knowledge base.

5. Sentence selection. From the obtained set of sentences (see item 3), only those whose weight exceeds a predetermined threshold value (see item 4) are selected for the summary $\breve{T} = [s' \in T' : SW(s') > \beta]$ where $\beta = 0.15$ is a constant defined empirically; it determines how much the text will be shortened.

6. Smoothing makes the resulting abstract more coherent and readable. While smoothing, some words are replaced with ones that are more suitable or they can be deleted. To smooth sentences, we used two types of templates: for removing fragments of sentences (in the case when the received summary is longer than 250 words) and for addition (in the case when a fragment of an unfinished sentence was included in the summary).

For example, let us consider the fragment *"**Indeed, we can show how —** in the case of a simple linear model with a quadratic error function and simple gradient descent — early stopping is equivalent to L2 regularization. **In order to** compare with classical L2 regularization, we examine a simple setting where the only parameters are linear weights".* After smoothing, we will get *"In the case of a simple linear model with a quadratic error function and simple gradient descent — early stopping is equivalent to L2 regularization. We examine a simple setting where the only parameters are linear weights".*

A flowchart of the system we developed ('Scientific Text Summarizer') is shown in Fig. 3.

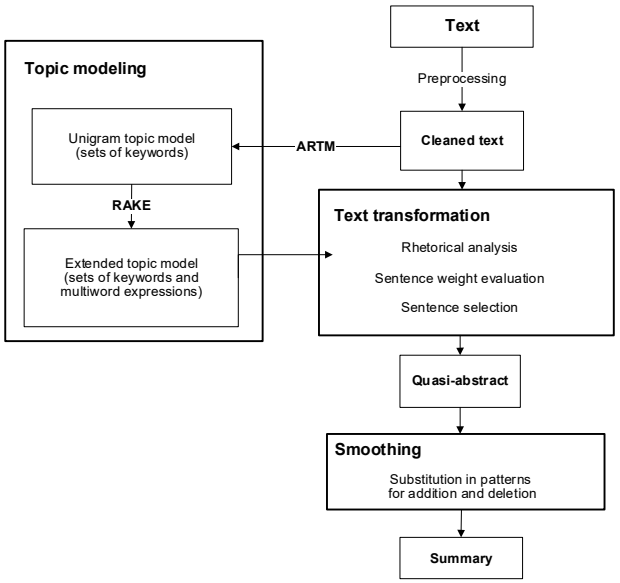

Figure 3 – System flowchart

5. RESULTS

Our system was tested on a collection of 1200 scientific articles in the Russian language taken from the open-access journal archives "Software & Systems" for 2013–2018.[1]

There is still no generally accepted effective method for automatic evaluation of summarization systems [25]. Firstly, we tried to assess the quality of the received abstract with the ROUGE metric [26], based on counting the number of matching text elements, for example, n-grams, or sentences. In this metric, the summary sentence is considered as a sequence of words. The main point is that the longer the *LCS* (the longest common subsequence) of the two summary sentences, the more similar the two summaries are. It is suggested to use the F-measure based on *LCS* to evaluate the similarity between the two sums *X* length *m* and *Y* length *n*, assuming that *X* is a reference aggregate sentence, and *Y* is the summary sentence for viewing as follows:

$$P_{lcs} = \frac{LCS(X,Y)}{n}, \quad R_{lcs} = \frac{LCS(X,Y)}{m}, \tag{4}$$

where $LCS(X, Y)$ is the length of a longest common subsequence of *X* and *Y*, and $\beta = P_{lcs} / R_{lcs}$.

The following values of the ROUGE metric were obtained: precision 32.8 %, recall 59.04 %, F-measure 34.47 %. Unfortunately, in works [19, 20], which describe summarization systems of texts in Russian, ROUGE values are not given, so it is not possible to compare those results with ours. We concluded that it is incorrect to compare our results with the systems for the English language, such as, for example, [27], since such low values of ROUGE can be associated with the peculiarities of the language type. Russian is an inflected language with developed morphology.

Secondly, we used an expert evaluation. The precision of the obtained summaries estimated by experts was significantly higher. An expert is a person who evaluates whether the content of the original article matches the text of the automatically received summary. An expert evaluation showed that 86.43 % of the generated abstracts coincided with the author's abstracts or to some extent differed in meaning from the author's one (which in fact does not always indicate a low quality of the abstract) and 13.57 % were incorrectly selected fragments of the texts. It should be noted that the expert evaluation we obtained is higher than 71.6% in [20] and 80.84% in [19].

We have noticed that authors often use synonyms, paraphrase and change sentences in places. The expert evaluation confirms that the order of sentences in an abstract, as a rule, does not affect its general meaning. However, the ROUGE value does not consider this. In addition, sometimes automatically generated summary is longer than we would like to have (about 500 words instead of 250). This is due to the large number of meaningful sentences in the text.

However, it is believed that the expert assessment depends on a particular expert, and therefore, is subjective. Therefore, along with the expert evaluation, an automatic evaluation was carried out.

Thirdly, we examined the precision, recall and F-measure calculated in a way similar to [13, 19]. Let us explain in more detail. Suppose that the automatically generated summary contains a set W_1 of keywords and multiword terms, a set V_1 of special words that are often found in scientific texts, and a set D_1 of discursive markers. The union of these sets is denoted by $N_1 = W_1 \cup V_1 \cup D_1$. Similar sets can be defined for the author's summary $N_2 = W_2 \cup V_2 \cup D_2$. Then the precision, recall and F-measure will be calculated by the following formulas:

$$Precision = \frac{|N_1 \cap N_2|}{|N_1|}, \quad Recall = \frac{|N_1 \cap N_2|}{|N_2|}, \quad F_{measure} = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall}. \tag{5}$$

The advantage of the proposed formulas is that

[1] International research and practice journal "Software & Systems. URL: http://www.swsys.ru/index.php?lang=en

they allow us to evaluate the contribution of each characteristic and various their combinations to the overall evaluation of the result. For example, you can evaluate the contribution of only markers, or only special scientific vocabulary, or both but without key words and expressions, etc. In the future, we plan to conduct a similar study of this issue.

The comparison of the results is given in Table 2 (our system is called 'Scientific Text Summarizer').

Table 2. Evaluation of automatic text summarization

System	Method	Precision, %	Recall, %	F-measure, %
Russian collection				
Trevgoda (2009)	Templates	67.03	64.81	66.03
Open Text Summarizer (2016)	Statistical	12.00	24.20	38.50
Scientific Text Summarizer (2018)	**Combined**	**75.23**	**68.21**	**71.55**
English collection				
Marcu (1998)	Heuristics combination	73.53	67.57	70.42

We evaluate the running time of the algorithm (RAM 6 GB, Intel Core i5-4210U 1.7 GHz). The algorithm worked for about 3 min on a small collection, and 8 min on a large one (see Table 3).

Table 3. Evaluation of running time

Step	Time	
	Collection of 260 texts	Collection of 1200 texts
Preprocessing	15 sec	1 min
RAKE	5 sec	20 sec
ARTM (training)	2 min 13 sec	5 min 37 sec
ARTM (testing)	5 sec	15 sec
Summarization	10 sec	40 sec
Total	~ 3 min	~ 8 min

Possible improvement of the algorithm proposed in this article, in our opinion, is to take into account the cases of anaphora [28] and part-of-speech homonymy [29], and fill up the linguistic knowledge base with markers.

6. CONCLUSION

In this paper, we described an approach to automatic summarization of scientific and technical texts in Russian. We extracted most significant sentences based on discursive markers. Keywords, multiword terms, and some special words that are often occur in scientific and technical texts were also taken into account. Experiments have shown the high quality of the proposed algorithm. However, it should be noted that in the case of a large number of formulas, drawings and graphs in the source text, the method works worse. Among the shortcomings, it should be noted that manual tuning of the knowledge base is necessary. Nevertheless, the experiments showed that this method is effective, it needs a comparatively small amount of training data and can be adapted to processing texts from different subject fields in other languages.

7. ACKNOWLEDGMENTS

This study was funded by RFBR according to the research project N 19-07-01134.

8. REFERENCES

[1] E. Lloret, M.T. Roma-Ferri, M. Palomar, "COMPENDIUM: A text summarization system for generating abstracts of research papers," *Data & Knowledge Engineering*, vol. 88, pp. 164–175, 2013.

[2] E. Hovy, Ch.-Y. Lin, "Automated text summarization and the SUMMARIST system," *Proceedings of the TIPSTER Text Program*, 1998, pp. 197–214.

[3] H. Saggion, G. Lapalme, "Generating indicative-informative summaries with SumUM," *Computational Linguistics*, vol. 28, no. 4, pp. 497–526, 2002.

[4] G.F. Foster, *Statistical Lexical Disambiguation*, *Master's Thesis*, 1991, 340 p.

[5] L. Plaza, A. Diaz, P. Gervas, "Concept-graph based biomedical automatic summarization using ontologies," *Proceedings of the 3rd Textgraphs workshop on Graph-Based Algorithms in Natural Language Processing, Coling'2008*, Manchester, 2008, pp. 53–56.

[6] Unified Medical Language System (UMLS), 2016, [Online] Available at: http://www.nlm.nih.gov/research/umls/

[7] A.R. Aronson, "Effective mapping of biomedical text to the UMLS metathesaurus: The MetaMap program," *Proceedings of the American Medical Informatics Association*, 2001, pp. 17–21.

[8] A. Farzindar, G. Lapalme, "Legal text summarization by exploration of the thematic

structures and argumentative roles," *Proceedings of the Workshop on Text Summarization Branches Out*, ACL, Barcelona, Spain, 2004, pp. 27–38.

[9] F. Galgani, P. Compton, A. Hoffmann, "Combining different summarization techniques for legal text," *Proceedings of the Workshop on Innovative Hybrid Approaches to the Processing of Textual Data (Hybrid 2012), EACL'2012*, Avignon, France, 2012, pp. 115–123.

[10] S. Megala, A. Kavitha, A. Marimuthu, "Feature extraction based legal document summarization," *International Journal of Advance Research in Computer Science and Management Studies*, vol. 2, issue 12, pp. 346–352, 2014.

[11] E. Lloret, E. Boldrini, T. Vodolazova, P. Martínez-Barco, R. Muñoz, M. Palomar, "A novel concept-level approach for ultra-concise opinion summarization," *Expert Systems with Applications*, vol. 42, issue 20, pp. 7148–7156, 2015.

[12] S. Brügmann, N. Bouayad-Aghab, A. Burga, S. Carrascosa, A. Ciaramella, M. Ciaramella, J. Codina-Filba, E. Escorsa, A. Judea, S. Mille, A. Müller, H. Saggion, P. Ziering, H. Schütze, L. Wanner, "Towards content-oriented patent document processing: Intelligent patent analysis and summarization," *World Patent Information,* vol. 40, pp. 30–42, 2015.

[13] D. Marcu, "Improving summarization through rhetorical parsing tuning," *Proceedings of the Sixth Workshop on Very Large Corpora*, 1998, pp. 206–215.

[14] F. Andonov, V. Slavova, G. Petrov, "On the open text summarizer," *International Journal "Information Content and Processing"*, vol. 3, no. 3, 2016. [Online]. Available at: http://www.foibg.com/ijicp/vol03/ijicp03-03-p05.pdf

[15] S. Teufel, M. Moens, "Summarizing scientific articles: experiments with relevance and rhetorical status," *Computational Linguistics*, vol. 28, issue 4, pp. 409–445, 2002.

[16] W. Bosma, "Query-based summarization using rhetorical structure theory," *Proceedings of the 15th Meeting of CLIN*, 2005, pp. 29–44.

[17] S.H. Huspi, *Improving Single Document Summarization in a Multi-Document Environment*, *PhD Thesis*, RMIT University, Melbourne, Australia, 2017, 190 p.

[18] S. Mithun, *Exploiting Rhetorical Relations in Blog Summarization*, PhD Thesis, Concordia University, Montreal, Canada, 2012, 230 p.

[19] S.A. Trevgoda, "Methods and algorithms of automatic text summarization based on the analysis of functional relations," *Abstract of PhD Thesis*, St. Peterburg, Russia, 2009, 15 p.

[20] P.G. Osminin, *Construction of a Model for Abstracting and Annotating Scientific and Technical Texts Focused on Automatic Translation*, PhD Thesis, Chelyabinsk, Russia, 2016, 239 p.

[21] K. Vorontsov, O. Frei, M. Apishev, P. Romov, M. Dudarenko, "BigARTM: open source library for regularized multimodal topic modeling of large collections," *Proceedings of the International Conference on Analysis of Images, Social Networks and Texts (AIST)*. Yekaterinburg, Russia, 2015, pp. 370–384.

[22] W. Mann, C. Thompson, "Rhetorical structure theory: Toward a functional theory of text organization," *Text-Interdisciplinary Journal for the Study of Discourse*, vol. 8, no. 3, pp. 243–281, 1988.

[23] M. Louwerse, "An analytic and cognitive parameterization of coherence relations," *Cognitive Linguistics*, vol. 12, issue 3, pp. 291–315, 2001.

[24] S. Rose, D. Engel, N. Cramer, W. Cowley, "Automatic keyword extraction from individual documents," *Text Mining: Applications and Theory*, 2010, pp. 3–20.

[25] D. Das, A. Martins, "A survey on automatic text summarization. Literature," *Survey for the Language and Statistics II Course at CMU*, vol. 4, pp. 192–195, 2007.

[26] Ch.Y. Lin, "ROUGE: A package for automatic evaluation of summaries," *Proceedings of the Workshop on Text Summarization Branches Out*, 2004, pp. 74–81.

[27] J.J. Zhang, H.Y. Chan, P. Fung, "Improving lecture speech summarization using rhetorical information," *Proceedings of the 2007 IEEE Workshop on Automatic Speech Recognition & Understanding (ASRU)*, 2007, pp. 195–200.

[28] A. Kozlova, O. Gureenkova, A. Svischev, T. Batura, "A hybrid approach for anaphora resolution in the Russian language," *Proceedings of the 2017 Siberian Symposium on Data Science and Engineering (SSDSE)*. Russia, 12-13 April 2017, pp. 36–40.

[29] T. Batura, E. Bruches, "Combined approach to problem of part-of-speech homonymy resolution in Russian texts," *Proceedings of the 2018 International Russian Automation Conference, RusAutoCon 2018*, 9-16 September 2018, pp. 4–9.

Tatiana Batura *is an expert in natural language processing, computer and mathematical linguistics. She has PhD in Physics and Mathematics. She is a Senior Researcher at the A. P. Ershov Institute of Informatics Systems, Siberian Branch of the Russian Academy of Sciences (IIS SB RAS); Associate Professor at the Novosibirsk State University. Tatiana Batura is an author and co-author of more than 80 publications, including four scientific monographs.*

Her scientific interests are text mining, data mining, text semantics, information search and retrieval, machine learning.

Aigerim Bakiyeva *is an expert in computer linguistics. She has PhD in Technical Sciences. She is an Assistant Professor at the L.N. Gumilyov Eurasian National University. Aigerim Bakiyeva is an author and co-author of more than 30 publications.*

Her scientific interests are text summarization and formalization of natural language semantics.

Maria Charintseva *is a linguist and an expert in natural languages. She is a Senior Lecturer of the English language, Department of Information Technologies, Novosibirsk State University. Maria Charintseva is an author and co-author of 3 publications.*

Her scientific interests are text mining, text semantics, information search and retrieval.