O. V. Bisikalo, Dr. Sc. (Eng.), Prof.; I. O. Nazarov

REVIEW OF THE METHODS FOR AUTOMATED ABSACTING OF THE TEXTS

The paper considers methods for automated text abstracting. On the basis of the conducted review application of the constraint propagation model is proposed for improving the method of text relation maps (TRM).

***Keywords**: automated abstracting, constraint propagation model, TRM method.*

Introduction

Abstract is a brief summary of the text where the main issues covered are listed. Abstracts are classified according to the content and purpose, to the fullness of covering the content and the type of readers they are designed for. According to the first criterion, reference and recommendatory abstracts are distinguished, according to the second criterion, there are general and specialized abstracts and as a separate type, survey abstracts exist [1].

For the first time the notion of abstract appeared in the second half of the I century AD, but functionally abstracts were present even in the catalogs of the Library of Alexandria (III c. BC). Steady accumulation and increase in the volumes of textual information under the conditions of the development of information technologies determine the current importance of the problem of automated abstracting the natural language textual materials. This problem is one of the main trends in computer linguistics which is closely connected with automated summarization. Taking into account differences in the concepts of summary and abstract, such problems are solved by similar methods.

Automatic processing of the natural language texts involves difficulties in the process of formalizing the problems set. On the other side, the formalization method is influenced by the existence of different abstract types (the actual result of system operation) and the approach to their structuring. In general case automated abstracting means that for a given text T another text A (abstract) is formed, text A containing brief statement of the issues covered in T:

$$T \to A. \tag{1}$$

Due to their discrete nature texts are convenient to be considered as finite sets $T = \{t_1, t_2, ..., t_n\}$ and $A = \{a_1, a_2, ..., a_m\}$. Different lexical units could be the elements of set T (sentences, paragraphs, etc.) depending on its textual size, while the elements of set A are sentences (due to its limited textual size). If sentences are considered as the elements of the both above-mentioned sets, then from the last statement it follows that it is necessary to satisfy the condition:

$$\{A\} << \{T\}. \tag{2}$$

The main difficulty of automatic abstracting problem is to ensure coincidence of the basic meaning of text T with abstract A and, actually, the search for such a sense.

Problem statement

The research goal is consider the main algorithms of generation and extraction for solving the problem of automatic abstracting of natural language texts; proceeding from the priority of semantic analysis, to find the method among the generation algorithms for its modification using the approach based on the constraint propagation model.

Automated abstracting methods

Currently, there are a number of approaches to solving the automatic abstracting problem. It is common to divide them into two groups: the methods for composing extracts (extraction algorithms) and those for forming brief summaries (generation algorithms). Extraction algorithms form an abstract using text fragments of the source document. For this the blocks of the highest lexical and statistical importance are identified. In this case abstract is a connection of the selected fragments. Generation algorithms analyze the source document searching for information on the basis of which the abstract text is formed. Evidently, the first of the above approaches is simple for implementation and does not require big computational resources. However, it does not ensure appropriate quality due to the absence of semantic analysis of the text. The second approach provides a number of advantages: absence of duplication of the information in the source text and in the abstract, comprehensiveness of the abstract, consideration of semantic relations in the text. Therefore, this work recognizes the priority of generation algorithms as those having prospects for application in the creation high-level automatic abstracting systems.

Fig. 1 presents a diagram of the existing abstracting methods taking into account the above classification criterion.

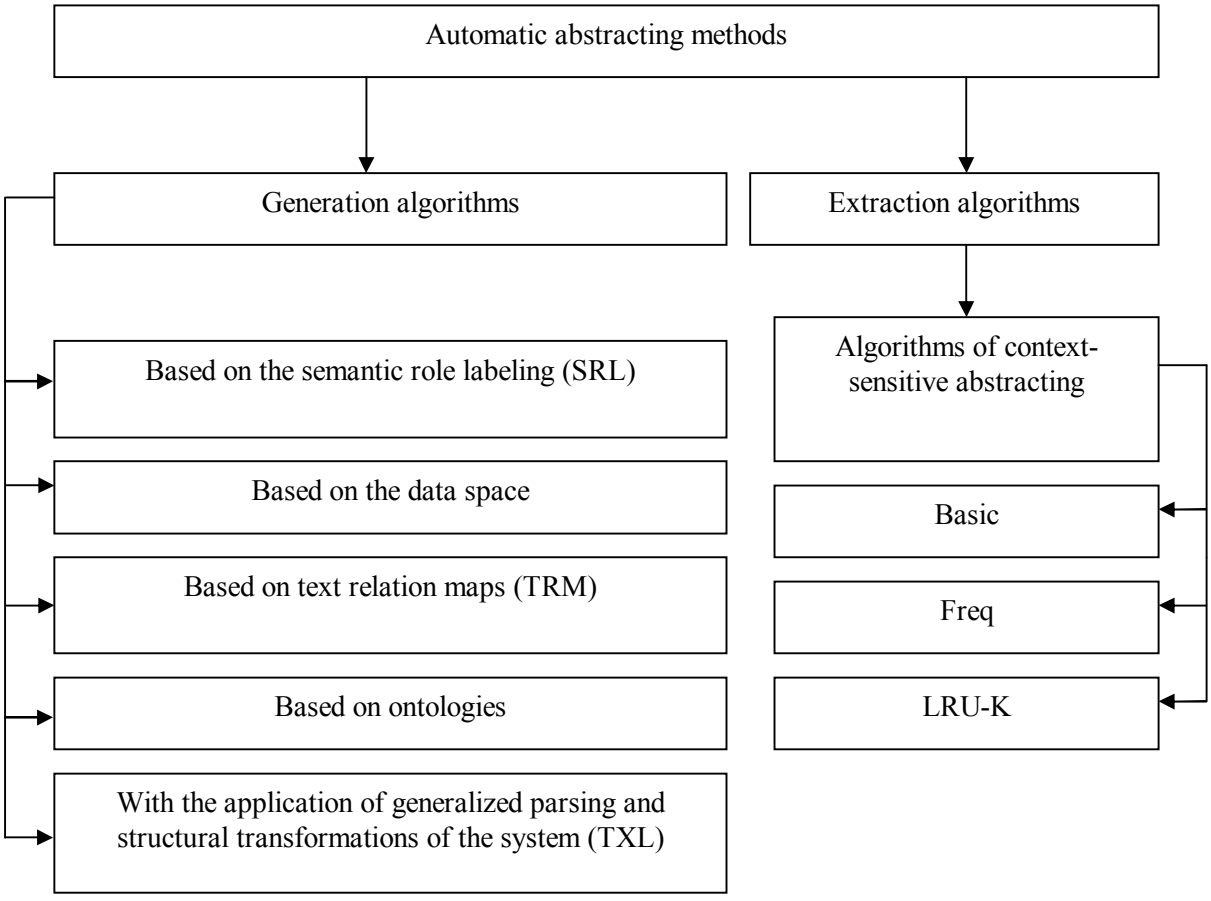

Fig. 1. Classification of automatic abstracting methods

The algorithms of context-sensitive abstracting of HTML documents are the most popular ones among the extraction algorithms. Abstracts composed by such methods are used by search engines to describe the results in the form of short continuous text fragments in accordance with the reader's query. Selection of the optimal text fragment is based on the fragment weight calculations.

The basic algorithm

To calculate the fragment weight, this algorithm uses the formula

$$W = \sum_{i=1}^{n} W_i + K\frac{n}{L}, \tag{3}$$

where W_i is weight of the i^{th} word of the query which is present in the fragment; $K = const$; n – the number of query words that are present in the fragment; L – distance between the first and the last words of the query.

The weight of the i^{th} word W_i of the query is calculated as

$$W_i = \frac{\log_2 N_i}{\log_2 N}, \tag{4}$$

where N_i is the number of documents where the i^{th} word occurred; N – total number of the documents.

Text fragment with the largest weight is included into the list of search results. If the number of such fragments is more than one, the algorithm uses a simple rule: the fragment closest to the beginning of the text is included into the list.

The experiments show [2] that the basic algorithm is the most efficient as to its speed but (according to the expert estimates) it has inferior quality of abstracting as compared with other algorithms.

Freq algotithm

This algorithm is an improvement of the previous one and, except the number of words in the query, takes into account document words with maximal frequency of occurrence. The fragment weight is calculated by the formula

$$W = W_b + \sum_{i=1}^{n} \log_2 F_i, \tag{5}$$

where W_b is the weight calculated by the basic algorithm; n – the number of words with the highest frequency of occurrence; F_i – frequency of occurrence of the i^{th} word.

Freq algorithm is much inferior in its speed as compared with the basic one but provides higher abstracting quality.

LRU-K algorithm

This algorithm, proposed in [2], is a variant of the “last recently used’ algorithm. The authors use estimation of the local frequency of the word occurrence on the condition of uniform distribution of the words. Experimental studies have shown efficiency of the proposed algorithm application for context-sensitive abstracts: abstracting quality is somewhat better than that achieved by Freq algorithm, its speed being much higher

Algorithm based on semantic role labeling (SRL)

The algorithm is based on the block for analyzing semantic structures of argument-predicates. The essence of the algorithm is semantic labeling of the relations in the text. To provide coherence of the abstract sentences, labeling is checked by a predicate structure. The abstract is formed by a three-level processing:

1. Syntactic analysis
2. Building the dependency tree

3. Lexical construction.

The main advantages of this method are the possibility to form a comprehensive and complete abstract as well as absence of the source text duplications in the abstract. Determination of relations between the words, their gender, case and number enables replacement, rejection and abstraction of the words. Disadvantages of the semantic labeling-based algorithm are difficulty of its implementation as well as the necessity of knowing all the relations in the text. The latter is a separate complex problem without solving which the algorithm practical implementation is impossible.

Algorithm based on data space

Automated abstracting of data about a certain event is considered in [3]. The abstract generation problem is solved in two stages:

1. Integration of the scattered information and search for information about the event
2. Abstracting of the event and calculating the coefficients of confirmation and refutation of the information.

For solving the first of the above subproblems the approach based on data space is proposed. Data space is a structure composed from data (presented in the form of data bases, data warehouses, statistical web-pages), local repositories and indices, tools for searching, processing and integrating the information).

Solving subproblem of calculating the coefficients of confirming and rejecting the information involves building the adaptive ontology of information means. The derived formulas enable numerical evaluation of these coefficients. The obtained values are used for building the abstract that consists from two paragraphs: the first one confirms the event considered and the second disproves it. Relationship between them makes it possible to determine, with a certain degree of probability, whether the event has really occurred.

Algorithm based on text relation maps (TRM)

The algorithm is based on the Text Relationship Map (TRM) [4]. The idea consists in formalization of the text in the form of a graph

$$G = (P, V), \tag{6}$$

where $P = \{\overline{p_1}, \overline{p_2}, ..., \overline{p_n}\}$ is a set of the graph vertices; $E = \{e_1, e_2, ..., e_m\}$ – a set of edges between the vertices.

Each vertex of such a graph represents a fragment of the source text and is a weighted vector that includes weights of separate words in the fragment:

$$\overline{p_i} = (p_{i1}, p_{i2}, ..., p_{ik}). \tag{7}$$

The edges connect the vertices with a high degree of similarity that is determined as a scalar product of the vectors of vertices:

$$m_{ij} = \overline{p_i}\,\overline{p_j}. \tag{8}$$

Presence of the edge between a pair of vertices is the evidence of semantic proximity of the given text fragments. The number of edges connected with a vertex determines the importance of the text fragment represented by this vertex. Building the abstract makes it possible to identify the most significant fragments by means of sorting them according to the number of connected edges.

This algorithm provides performing semantic analysis of texts with the purpose of forming abstracts. Besides, it could be used to search for semantically close documents, dividing them into groups according to a certain subject, etc. The main difficulty in the algorithm implementation is building the Text Relation Map, which involves numerical estimation of the weights of words in the text fragments and the degree of similarity between the fragments.

Algorithm with the application of generalized parsing and structural transformations of TXL system

A semantic abstracting metnod proposed in [5] uses generalized parsing and structural transformations of TXL system. TXL is a programming language developed in order to support computer software analysis and document transformation problems. In this case the abstracting process includes three stages:

1. Parsing tools available in TXL are used for parsing the source text and obtaining an approximated structure of the phrases. 2. Positive and negative indices of the semantic categories for the word list are given to obtain an initial semantic abstract of the document.

3. The marked-up XML text is used for filling XML database.

This algorithm is used for semiautomatic abstracting of natural language texts. An initial abstract, obtained at the second stage, is corrected by an expert during the next stage. This limitation is a significant drawback of this approach.

Conclusions

Due to the limited scope of this work, it covers far from all currently existing approaches to solving the problem of automatic abstracting of natural language texts. There is a great number of semiautomatic abstracting methods that should be considered as a separate group as they require participaton of an expert in the process of compiling abstracts. During this study the authors noted similarity of the approaches to automatic abstracting and summarization. Therefore, appropriately modified summarization techniques could be used for compiling abstracts.

As it follows from the conducted review, generation algorithms use, to a greater or lesser degree, semantic analysis of the source text. The main drawback of the existing approaches is imperfectness of such an analysis and, consequently, absence of a noticeable success in solving the problem. Therefore, it is feasible to use a method of determining the sense of textual information on the basis of constraint propagation model proposed in [6]. Effective semantic analysis could be used to improve the above algorithm based on text relation maps (TRM).

In order to adapt the text relation model to the proposed approach, it is feasible to represent sentences as vertices of a graph instead of classical usage of paragraphs as vertices. This will make it possible to improve decision making during abstracting process. As different from the classical approach, the possibility is provided to build a graph not for a separate text but for a collection of documents. In such case the graph is build in the following way: each sentence of the text is represented as a vertex; edges between the vertices determine the degree of semantic connecton between sentences. It is rational to represent the text in an adapted form – without language units that do not bear semantic meaning. As a lexical similarity measure of sentences the cosine measure may be used [7]. Improvement of the mathematical apparatus of this method is considered to be a prospective trend of the research.

REFERENCES

1. Ильичева Н. В. Аннотирование и реферирование / Н. В. Ильичева, А. В. Горелова, Н. Ю. Бочкарева. – Самара: Изд-во Самарского госуниверситета, 2003. – 100 с.

2. Губин М. В. Эффективный алгоритм формирования контекстно-зависимых аннотаций / М. В. Губин, А. И. Меркулов // Компьютерная лингвистика и интеллектуальные технологии. Труды международной конференции «Диалог-2005» (Звенигород, 1-6 июня 2005 г.). – 2005. – С. 116 – 120.

3. Про задачу автоматичного анотування події на основі простору даних [Електронний ресурс] / Шаховська Н. Б., Литвин В. В. // Науковий вісник Чернівецького національного університету ім. Юрія Федьковича. Збірник наук. праць. – Вип. 426: Фізика. Електроніка. – 2008. Режим доступу до журн.: http://www.nbuv.gov.ua/portal/natural/Nvchnu_ks/2008_426/426_09_Shakhovska.pdf.

4. Митрофанов М. С. Автоматическое аннотирование документов в многокомпонентной системе поиска и анализа естественно-языковой инфорации / М. С. Митрофанов, И. Е. Чижевский // Научная сессия МИФИ-2010. Ч. 1. XIV выставка-конференция. Телекоммуникации и новые информационные технологии в образовании. – С. 156 – 159.

5. Kiyavitskaya N. Text Mining through Semi Automatic Semantic Annotation / N. Kiyavitskaya, N. Zeni, L. Mich, J. Cordy, J. Mylopoulos // PAKM 2006. LNCS (LNAI). – vol. 4333. – 2006. – P. 143 – 154.

6. Квєтний Р. Н. Визначення сенсу текстової інформації на основі моделі розповсюдження обмежень / Р. Н. Квєтний, О. В. Бісікало, І. О. Назаров // Вимірювальна та обчислювальна техніка в технологічних процесах. – 2012. – № 1. – С. 93 – 96.

7. Salton G. Automatic text processing / G. Salton. – Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, – 1988. – 450 p.

Bisikalo Oleg – Dr. Sc. (Eng.), Prof. of the Department of Automatics and Information Measuring Equipment.

Nazarov Igor – Student, Department of Automatics and Information Measuring Equipment.
Vinnytsia National Technical University.