AUTOMATIC DOCUMENT SUMMARIZATION BY SENTENCE EXTRACTION

R. M. Aliguliyev
Institute of Information Technology
of the National Academy of Sciences of Azerbaijan, Baku
e-mail: a.ramiz@science.az

Представлен метод автоматического реферирования документов, который генерирует резюме документа путем кластеризации и извлечения предложений из исходного документа. Преимущество предложенного подхода в том, что сгенерированное резюме документа может включать основное содержание практически всех тем, представленных в документе. Для определения оптимального числа кластеров введен критерий оценки качества кластеризации.

Introduction

Automatic document processing is a research field that is currently extremely active. One important task in this field is automatic document summarization, which preserves its information content [1].

With a large volume of text documents, presenting the user with a summary of each document greatly facilitates the task of finding the desired documents. Text search and summarization are the two essential technologies that complement each other. Text search engines return a set of documents that seem to be relevant to the user's query, and text enable quick examinations through the returned documents.

In general, automatic document summarization takes a source document (or source documents) as input, extracts the essence of the source(s), and presents a well-formed summary to the user. Mani and Maybury [1] formally defined automatic document summarization as the process of distilling the most important information from a source(s) to produce an abridged version for a particular user (or users) and task (or tasks). The process can be decomposed into three phases: analysis, transformation and synthesis. The analysis phase analyzes the input document and selects a few salient features. The transformation phase transforms the results of analysis into a summary corresponding to users' needs. In the overall process, compression rate, which is defined as the ratio between the length of the summary and that of the original, is an important factor that influences the quality of the summary. While the compression rate increases, the summary will be more copious, relatively, more insignificant information is contained. In fact, when the compression rate is 5–30 %, the quality of the summary is acceptable.

Document summaries can be abstracts or extracts. An extract summary consists of sentences extracted from a document while an abstract summary may contain words and phrases which do not exist in the original document [1–3]. A document summary can also be either generic or query-dependent (user-focused). A user-focused summary presents the information that is most relevant to the initial search query, while a generic summary gives an overall sense of the documents content. A generic summary should meet two conditions: maintain a wide coverage of the document topics and keep low redundancy at the same time [4].

This paper presents generic summarization method that generates document summaries by clustering and extracting salient sentences from the source document.

1. A review of document summarization

Automatic document summarization has been actively researched in recent years and several automatic summarization methods proposed in the literature [2, 4–14].

Authors of the paper [2] propose two generic text summarization methods that create generic text summaries by ranking and extracting sentences from the original documents. The first method uses information retrieval methods to measure sentence relevance, while the second method uses the latent semantic analysis (LSA) technique to identify semantically important sentences, for summary creations. Both methods strive to select sentences that are highly ranked and different from each other. This is an attempt to create a summary with a wider coverage the main content of document and less redundancy. First main issues of summarization by context have been introduced and studied in paper [7]. Two new algorithms were proposed. Their efficiency depends on the size of the content and the context of target document. Goldstein et al. [4] present an analysis of news-article summaries generated by sentence selection. Sentences are ranked for potential inclusion in the summary using a weighted combination of statistical and linguistic features. The statistical features were adapted from standard information retrieval methods. Potential linguistic ones were derived from an analysis of news-wire summaries. To evaluate these features a modified version of precision-recall curves has been used.

Summarizing Web pages have recently gained much attention from researchers. Current methods of Web page summarizing can be basically divided into two groups: content- and context-based ones. Both of them assume fixed content and characteristics of Web documents without considering their dynamic nature. Approach proposed in the work [10] is an extension of content-based techniques but differently to them it considers a Web page as a dynamic object. Several Web-page summarization algorithms are proposed by authors of the paper [13] for extracting the most relevant features from Web pages for improving the accuracy of Web classification. To leverage the extracted knowledge, in the paper [3] proposed two text-summarization methods to summarize Web pages. The first approach is based on significant-word selection adapted from Luhn's method. The second method is based on LSA.

A practical approach for extracting the most relevant sentences from the original document to form a summary is presented in work [11]. A proposed approach takes advantages of both the local and global properties of sentences. The algorithm that combines these properties for ranking and extracting sentences is given. The local property can be considered as clusters of significant words within each sentence, while the global property can be thought of as relations of all sentences in a document. Authors of the paper [14] propose two text summarization approaches: modified corpus-based approach (MCBA) and LSA-based Text Relationship Maps (LSA+TRM). The first is a trainable summarizer, which considers several kinds of document features, including position, positive keyword, negative keyword, centrality, and the resemblance to the title, to generate summarizers. The second uses LSA to derive the semantic matrix of a document (in single-document level) or a corpus (in corpus level), and uses semantic sentence representation to construct a semantic TRM. Let's note that technique TRM is offered in work [12].

The paper [9] presents a special method for Chinese document summarization. The specificity of this method is that the generated summary can contain the main contents of different topics as many as possible and reduce its redundancy at the same time. By adopting K-means clustering algorithm as well as a novel clustering analysis algorithm, the number of different latent topic regions can be captured in a document adaptively.

In recent years, information retrieval techniques have been used for automatic generation of semantic hypertext links. This study applies the ideas from the automatic link generation research to attack another important problem in text processing – automatic text summarization. An automatic "general purpose" text summarization tool would be of immense utility in this age of information overload. Using the techniques used (by most automatic hypertext link generation algorithms) for inter-document link generation, generate intra-document links between passages of a document. Based on the intra-document linkage pattern of a text, characterize the structure of the text. Apply the knowledge of text structure to do automatic text summarization by passage extraction. Evaluate a set of fifty summaries generated using our techniques by comparing them to paragraph extracts constructed by humans. The automatic summarization methods perform well, especially in view of the fact that the summaries generated by two humans for the same article are surprisingly dissimilar [12].

Alguliev and Aliguliyev [5] present effective summarization method of text documents which consists of the following steps: 1) Informative features of words are selected, which affect the accuracy of summarization results; 2) Using genetic algorithms the weights of informative features influencing the relevance of words are calculated; 3) The aggregate similarity measure between each sentence and the rest of the sentences is calculated by cosine measure; 4) The similarity measure between each of the sentences and the title is calculated; 5) Weighted score of relevance is defined for each sentence; 6) The scores of relevance are ranked; 7) Starting with the highest score the sentences which the relevance score is higher than the threshold value set are included in the summary, and the process continues until the ratio of compression satisfies the limitation set in advance. In the article [6] for the purpose of the guarantee of minimum redundancy in the summary and a maximally possible degree of the coverage of document contents the summarization method is proposed. The proposed method based on clustering of sentences. Clustering of sentences is reduced to determination of the cluster centers, mathematical realization which based on a problem of global optimization. In fact, the number of clusters is a parameters related to the complexity of the cluster structure. The step-by-step algorithm for determination of the number of clusters is also proposed in this work [6].

In paper [15] an evaluation of a novel hierarchical text summarization method that allows users to view summaries of Web documents from small, mobile devices is presented. Unlike previous approaches, this method does not require the documents to be in HTML since it infers a hierarchical structure automatically. Currently the method is used to summarize news articles sent to a Web mail account in plain text format. Hierarchical summarization operates in two stages. First, it computes the salience of each sentence in the document and ranks the set of sentences accordingly. In the second stage, a tree is constructed from all of the sentences such that its root is the sentence with the highest salience and, given any sentence node with salience s at depth d, all sentences above that depth have a salience higher than s, while the salience of the rest of the sentences is below s.

Multidocument summarization poses a number of new challenges over single document summarization. Researchers have already investigated issues such as identifying repetitions or contradictions across input documents and determining which information is salient enough to include in the summary [16, 17].

Multidocument summaries need to be both informative and coherent. Informativeness is rendered by the methods of selecting the information from documents to incorporate it into the summary. The coherence of the summary is obtained by ordering the information originating in different documents.

Authors of the paper [16] describe five topic representations that use before in multidocument summarization and introduce a novel representation based on topic themes. In this paper also presented a total of six new multidocument summarization methods that use different information extraction and information ordering methods. The theme representations proposed in this paper are based on the shallow semantic information provided by semantic parsers, a source of linguistic information much more sophisticated than those employed by previous thematic representations.

Diao and Shan [17] present a novel multi-web page summarization algorithm. It adds the graph based ranking algorithm into the framework of maximum marginal relevance method, to not only capture the main topic of the web pages but also eliminate the redundancy existing in the sentences of the summary result.

2. Sentence representation

Text mining is an emerging field at the intersection of several research areas, including data mining, natural language processing, and information retrieval. The main differences between data mining and text mining are as follows [18]. Data mining usually deals with structured data sets — typically in the first normal form, in the terminology of relational databases. By contrast, text mining deals with unstructured or semi-structured data, namely the text found in articles, documents, etc. In data mining the choice of data representation is usually straightforward. In general, data is represented as a set of records, where each record is a set of attribute values. In text mining the situation is more complex. Due to the little (if any) structure of texts, the choice of data representation is not so obvious. We have chosen to use a vector space model. The basic idea is to represent sentences as multi-dimensional vectors.

Let document D be represented as collection of sentences $D = \{S_1, S_2, ..., S_N\}$. According to vector space model, each sentence can be represented as a vector of features with associated weights. The features are usually derived from the words/terms appearing in the document D. Let S_i be a sentence in the document D and which is represented as $(w_{1i}, w_{2i}, ..., w_{ni})$, where w_{ti} is the numeric weight for the feature T_t, and n is the number of words/terms in a document D. A weighting scheme, called TF-ISF (Term Frequency-Inverse Sentence Frequency) scheme, which is derived from the well-known TF-IDF weighting scheme, is composed of two components, namely, word/term frequency and inverse sentence frequency.

The word/term frequency, TF_{ti}, of a word/term T_t is defined as the number of occurrences of the word/term T_t in sentence S_i. The inverse sentence frequency, ISF_t, of a word/term T_t is defined as: $\mathrm{ISF}_t = \log\left(\frac{N}{N_t}\right)$, where N_t is the number of sentences in a document D in which the word/term T_t occurs. The weight w_{ti} is computed by: $w_{ti} = \mathrm{TF}_{ti} \cdot \mathrm{ISF}_t$, $t = 1, ..., n$, $i = 1, ..., N$.

3. A partitioning-based clustering method

A fundamental problem that frequently arises in a great variety of fields such as pattern recognition, machine learning, statistics, image processing, text mining, and many others is the problem of clustering [18].

Clustering methods have been studied for many years, but they continue to be the subject of active research. Clustering is an important unsupervised classification technique where a set of objects, usually vectors in multi-dimensional space, are grouped into clusters in such way that objects in the same cluster are similar in some sense and objects in different clusters are dissimilar in the same sense.

The four main classes of clustering algorithms available nowadays are partitioning methods, hierarchical methods, density-based clustering and grid-based clustering [18].

Partitioning methods are among the most popular approaches to clustering. Clustering methods of this type start with an initial partitioning of the data, and iteratively improve it by means of a greedy heuristic: data items are repeatedly reassigned to cluster according to a specific optimization criterion. The K-means algorithm is the best-known algorithm within this class. The criterion-function used by K-means is that of minimum variance, that is, the sum of squares of the differences between data items and their assigned cluster center is minimized. Starting from a random partitioning, the algorithm repeatedly (1) computes the current cluster centers and (2) reassigns each data item to the cluster centre closest to it. K-means terminates when no more reassignments take place, which is usually the case after only few iterations.

Note that the clustering and the document summarization can be used together in a synergetic, complementary way. For instance, the user can first perform a clustering of some sentences, to get an initial understanding of the document base. Then, supposing the user finds a particular cluster interesting, she/he can perform a summarization of the documents in that cluster [6, 9].

Automatic clustering is a process of dividing a set of objects into unknown groups, where the best number K of groups is determined by the clustering algorithm. That is, objects within each group should be highly similar to each other than to objects in any other group. Finding the K automatically is a hard algorithmic problem. The automatic clustering problem can be defined as follows:

Let $X = \{X_1, X_2, ..., X_n\}$ be a set of n objects. These objects are clustered into non-overlapping groups $C = \{C_1, C_2, ..., C_K\}$, where C is called a cluster, K is the unknown number of clusters, $C_k \cap C_l = \emptyset$ for $k \neq l$ $(k = 1, ..., K,\ l = 1, ..., K)$, $\bigcup_{k=1}^{K} C_k = X$, $C_k \subseteq X$ and $C_k \neq \emptyset$.

Definitions of the clustering problem vary in the optimization criterion and the distance function used. A multitude of possible optimization criteria exists; examples are the minimization of the intra-cluster variances or the maximization of the inter-cluster distances.

The definitions of the metric in sentence space require the selection of a suitable distance function. This function measures similarity and dissimilarity between individual sentences based on their vector representation.

Many measures are available for the calculation of inter sentence relationships [19] and the choice of a specific measure may influence the outcome of the calculations. Possible choices for the distance function include the Euclidean distance, the cosine similarity or the correlation coefficient. One of the distance functions widely used in text mining is the cosine measure, which is counted amongst the coefficients of association. The cosine similarity between two sentences S_i and S_j is defined as:

$$sim(S_i, S_j) = \cos(S_i, S_j) = \frac{\sum_{t=1}^{n} w_{ti} w_{tj}}{\sqrt{\sum_{t=1}^{n} w_{ti}^2 \cdot \sum_{t=1}^{n} w_{tj}^2}}. \tag{1}$$

Since the calculation of the cosine similarity can be computationally very expensive, it is often replaced by the weighted inner dot product, which is represented as follows [20]:

$$sim_\alpha(S_i, S_j) = \sum_{t=1}^{n} \alpha_t w_{ti} w_{tj}. \tag{2}$$

Where α_t is the weight of word/term t in the document with respect to informative feature weights [5, 20]. For all the informative features which are determined by word/term t α_t is the product of the informative feature weights. When a word/term written in regular size fonts in a document, $\alpha_t = 1.0$.

Let S_i and S_j be two sentences in the document D. If these sentences belong to the different clusters $S_i \in C_k$, $S_j \in C_l$, $i \neq j$, $k \neq l$, then with an optimistic view, we derive that the measure of proximity between them must be minimum:

$$sim_\alpha(S_i, S_j) \to \min \tag{3}$$

and the measures of proximity between them and the corresponding cluster centers O_k and O_l, respectively, must be maximum:

$$sim_\alpha(S_i, O_k) + sim_\alpha(S_j, O_l) \to \max . \tag{4}$$

The cluster center O_k is calculated as follows:

$$O_k = \frac{1}{N_k} \sum_{S_i \in C_k} S_i, \quad k = 1, 2, ..., K. \tag{5}$$

Where N_k is the number of sentences belonging to cluster C_k. It's clear that $\sum_{k=1}^{K} N_k = N$.

The proposed approach for clustering two sentences can be extended to a set $D = (S_1, S_2, ..., S_N)$ by preserving the general principles. Thus, if in formula (3) produce summation over all possible pairs (S_i, S_j), $i \neq j$, then we will obtain:

$$\sum_{i=1}^{N-1} \sum_{j=i+1}^{N} sim_\alpha(S_i, S_j) \to \min . \tag{6}$$

Let variable x_{ik} equal to 1, if sentence S_i belong to cluster C_k, and equal to 0 otherwise. After this designation the formulas (4) and (6) will take the following forms, respectively:

$$\sum_{i=1}^{N}\sum_{k=1}^{K} sim_\alpha(S_i, O_k)\, x_{ik} \to \max \tag{7}$$

and

$$\sum_{i=1}^{N-1}\sum_{k=1}^{K}\sum_{j=i+1}^{N}\sum_{\substack{l=1\\ l\neq k}}^{K} x_{ik}x_{jl} \to \min\,. \tag{8}$$

Combination of the formulas (7) and (8) gives the following task of clustering:

$$\text{Maximize}\left(\sum_{i=1}^{N}\sum_{k=1}^{K} sim_\alpha(S_i, O_k)\, x_{ik} - \sum_{i=1}^{N-1}\sum_{k=1}^{K}\sum_{j=i+1}^{N}\sum_{\substack{l=1\\ l\neq k}}^{K} sim_\alpha(S_i, S_j)\, x_{ik}x_{jl}\right), \tag{9}$$

subject to

$$\sum_{k=1}^{K} x_{ik} = 1 \text{ for any } i = 1, ..., N, \tag{10}$$

$$1 \leq \sum_{i=1}^{N} x_{ik} < N \text{ for any } k = 1, ..., K, \tag{11}$$

$$x_{ik} \in \{0, 1\} \text{ for any } i = 1, ..., N \text{ and } k = 1, ..., K. \tag{12}$$

The formulation (9) simultaneously ensures maximization of intra-cluster proximity (first term) and the minimization of inter-cluster proximity (second term) of sentences. The constraint (10) ensures that the sentence S_i belongs to only one cluster. The constraint (11) ensures that each cluster must contain at least one sentence and it must not contain all sentences.

A distance function that is rarely mentioned in text mining (in contrast so that, the Euclidean distance is commonly applied in the field of data mining) is the Euclidean distance function. It simply measures the spatial distance between sentence vectors and its computation is straightforward:

$$dist(S_i, S_j) = \sqrt{\sum_{t=1}^{n} (w_{ti} - w_{tj})^2}.$$

With this choice of function measure the problem (9) gets a following form:

$$\text{Minimize}\left(\sum_{i=1}^{N}\sum_{k=1}^{K} dist(S_i, O_k)\, x_{ik} - \sum_{i=1}^{N-1}\sum_{k=1}^{K}\sum_{j=i+1}^{N}\sum_{\substack{l=1\\ l\neq k}}^{K} dist(S_i, S_j)\, x_{ik}x_{jl}\right). \tag{13}$$

The first term in formula (13) is K-means algorithm which ensures the compactness of the obtained clusters. The second term ensures the separation between clusters.

4. A validity index and optimal number of clusters

Several validity indices have been proposed, using different definitions of compactness and separation. The validity index, which we propose $V_{\text{ram}}(K)$ has the following form:

$$V_{\text{ram}}(K) = \frac{Comp(K)}{Separ(K)}. \tag{14}$$

Here numerator represents compactness and a denominator represents separation between clusters:

$$Comp(K) = \sum_{k=1}^{K} sim(C_k), \tag{15}$$

$$Separ(K) = \sum_{k=1}^{K} \sum_{\substack{l=1 \\ l \neq k}}^{K} sim(C_k, C_l). \tag{16}$$

Where

$$sim(C_k) = \sum_{S_i, S_j \in C_k} sim(S_i, S_j), \tag{17}$$

$$sim(C_k, C_l) = \sum_{S_i \in C_k} \sum_{S_j \in C_l} sim(S_i, S_j), \quad k \neq l. \tag{18}$$

The value of $Comp(K)$ ($Separ(K)$) generally decreases (increases) when K increases because the clusters become compact (separation). Consequently, validity index $V_{\text{ram}}(K)$ is monotonously decreasing function with respect to K, i. e. $V_{\text{ram}}(K+1) < V_{\text{ram}}(K)$ for any K.

The cluster validation problem involves measuring how well the clustering results reflect the structure of the data set, which is an important issue in cluster analysis. The most important indicator of the structure is the number of clusters. Since most basic clustering algorithms assume that the number of clusters in a data set is a user-defined parameter (one that is difficult to set in practical applications), the common approach is an iterative trial-end-error process. For this purpose, algorithms have been proposed in the literature [6, 8, 21].

The minimum number of clusters that satisfies the following condition:

$$\frac{V_{\text{ram}}(K) - V_{\text{ram}}(K+1)}{V_{\text{ram}}(K)} < \varepsilon \tag{19}$$

is taken as the optimal number of clusters, K. Here ε is a given tolerance.

5. Extraction of representative sentences

After clustering the following step is determination of representative sentences and their quantity in clusters. Representative sentences are determined thus. Within each cluster C_k for each sentence S_i calculate the aggregate proximity measure which determines by the following formula [5]:

$$sim_{\Sigma}(S_i \in C_k) = \sum_{\substack{S_j \in C_k \\ j \neq i}} sim(S_i, S_j) + sim(S_i, O_k). \tag{20}$$

Where $i = 1, ..., N_k$, $k = 1, ..., K$.

It's known that the title is one of information carriers in the document. Therefore to take into account contribution of title into summarization, let's define the proximity measure between sentence S_i and the title:

$$sim_{\text{title}}(S_i \in C_k) = sim(S_i, \vec{T}). \tag{21}$$

Where $\vec{T}$ — the characteristic vector of terms (words), corresponding to title.

Then score of relevance of the sentence S_i it will be determined by the weighted sum [5]:

$$score(S_i \in C_k) = \beta_1 sim_\Sigma(S_i \in C_k) + \beta_2 sim_{\text{title}}(S_i \in C_k). \tag{22}$$

Where weights $\beta_1, \beta_2 \in [0,1]$ and $\beta_1 + \beta_2 = 1$.

Prior to inclusion of sentences into summary, on each cluster they are ranked in descending order by their relevancy scores. After ranking, the sentences are included into summary starting with the highest score of relevance. This process continues up to the points when compression rate $rate_{\text{comp}}$ satisfies limit set. Compression rate is defined as the ratio between the length of the summary and that of the original [1]:

$$rate_{\text{comp}} = \frac{length(summary)}{length(D)}. \tag{23}$$

Where $length(summary)$, $length(D)$ — lengths (number of terms/words) of summary and the document D, respectively. The majorities of text documents usually consist of several themes. Some themes are described by many sentences and, consequently are formed main content of document. Other themes can only be briefly mentioned, in order to supplement main thematic. Consequently, the number of sentences on each cluster will differ from each other. In this case the number of selective sentences from each cluster will be also different. The number N_k^{repr} of representative sentences in the cluster C_k is determined according to formula:

$$N_k^{\text{repr}} = \left| \frac{length(C_k) \cdot rate_{\text{cmpr}}}{length^{\text{avg}}(S)} \right|. \tag{24}$$

Where $k = 1, ..., K$, $length(C_k)$ — the length of cluster C_k, $length^{\text{avg}}(S) = \dfrac{length(D)}{N}$ — average length of sentences in a document D, and $|\cdot|$ it indicates whole part.

This approach ensures it possible to a maximally possible degree to cover main content of document and to avoid redundancy.

Summary evaluation methods attempt to determine how adequate or how useful a summary is relative to its source.

To evaluate the results of summarization we shall use F_1 — measure. Let N_D^{relev} — a number of relevant sentences in the document D, $N_{\text{summary}}^{\text{relev}}$ — a number of relevant sentences in the summary, N_{summary} — a number of sentences in summary, P — precision, R — recall. Then it follows that [4]:

$$P = \frac{N_{\text{summary}}^{\text{relev}}}{N_{\text{summary}}}, \tag{25}$$

$$R = \frac{N_{\text{summary}}^{\text{relev}}}{N_D^{\text{relev}}}, \tag{26}$$

$$F_1 = \frac{2PR}{P+R}. \tag{27}$$

Conclusion

As amount of textual information available electronically grows rapidly, it becomes more difficult for a user to cope with all the text that is potentially of interest. Automatic document summarization methods are therefore becoming increasingly important.

Document summarization is a problem of condensing a source document into a shorter version preserving its information content. The generic summarization method that extracts the most relevance sentences from the source document to form a summary in this paper is proposed. The proposed method is based on clustering of sentences. The specificity of this approach is that the generated summary can contain the main contents of different topics as many as possible and reduce its redundancy at the same time. By adopting a new cluster analysis algorithm, we determine the different topics in a document.

Several clustering techniques are available in the literature which optimize of the distance criterion either by minimizing the within cluster spread, or by maximizing the inter-cluster separation. The proposed clustering method satisfies as much homogeneity within each cluster as well as much separability between the clusters as possible.

References

[1] Mani I., Maybury M.T. Advances in automated text summarization. Cambridge: MIT Press, 1999. 442 p.

[2] Gong Y., Liu X. Generic text summarization using relevance measure and latent semantic analysis // Proc. of the 24th Annual Intern. ACM SIGIR Conf. on Research and Development in Information Retrieval. New Orleans, USA, 2001. P. 19–25.

[3] Sun J.-T., Shen D., Zeng H.-J. et al. Web-page summarization using clickthrough data // Proc. of the 28th Annual Intern. ACM SIGIR Conf. on Research and Development in Information Retrieval. Salvador, Brazil, 2005. P. 194–201.

[4] Goldstein J., Kantrowitz M., Mittal V., Carbonell J. Summarization text documents: sentence selection and evaluation metrics // Proc. of the 22nd Annual Intern. ACM SIGIR Conf. on Research and Development in Information Retrieval. Berkeley, USA, 1999. P. 121–128.

[5] Alguliev R.M., Aliguliyev R.M. Effective summarization method of text documents // Proc. of the 2005 IEEE/WIC/ACM Intern. Conf. on Web Intelligence. Compiegne, France, 2005. P. 264–271.

[6] Alguliev R.M., Aliguliyev R.M., Bagirov A.M. Global optimization in the summarization of text documents // Automatic Control and Computer Sciences. N.Y.: Allerton Press, Inc. 2005. Vol. 39, N 6. P. 42–47.

[7] Delort J.-Y., Bouchon-Meunier B., Rifqi M. Enhanced Web document summarization using hyperlinks // Proc. of the 14th ACM Conf. on Hypertext and Hypermedia. Nottingham, United Kingdom, 2003. P. 208–215.

[8] Harabagiu S., Lacatusu F. Topic themes for multi-document summarization // Proc. of the 28th Annual Intern. ACM SIGIR Conf. on Research and Development in Information Retrieval. Salvador, Brazil, 2005. P. 202–209.

[9] Hu P., He T., Ji D., Wang M. A study of Chinese text summarization using adaptive clustering of paragraphs // Proc. of the 4th Intern. Conf. on Computer and Information Technology. Wuhan, China, 2004. P. 1159–1164.

[10] Jatowt A., Ishizuka M. Web page summarization using dynamic content // Proc. of the 13th Intern. World Wide Web Conf. New York, USA, 2004. P. 344–345.

[11] Kruengkrai C., Jaruskulchai C. Generic text summarization using local and global properties of sentences // Proc. of the IEEE/WIC Intern. Conf. on Web Intelligence. Halifax, Canada, 2003. P. 201–206.

[12] Mitra M., Singhal A., Buckley C. Automatic text summarization by paragraph extraction // Proc. of the ACL'97/EACL'97 Workshop on Intelligent Scalable Text Summarization. Madrid, Spain, 1997. P. 39–46.

[13] Shen D., Chen Z., Yang Q. et al. Web-page classification through summarization // Proc. of the 27th Annual Intern. Conf. on Research and Development in Information Retrieval. Sheffield, United Kingdom, 2004. P. 242–249.

[14] Yeh J.-Y., Ke H.-R., Yang W.-P., Meng I.-H. Text summarization using a trainable summarizer and latent semantic analysis // Information Processing and Management. 2005. Vol. 41, N 1. P. 75–95.

[15] Otterbacher J., Radev D., Kareem O. News to go: hierarchical text summarization for mobile devices // Proc. of the 29th Annual Intern. ACM SIGIR Conf. on Research and Development in Information Retrieval. Seattle, USA, 2006. P. 589–596.

[16] Barzilay R., McKeown K. R., Elhadad M. Inferring strategies for sentence ordering in multidocument news summarization // J. of Artificial Intelligence Research. 2002. Vol. 17. P. 35–55.

[17] Diao Q., Shan J. A new web page summarization method // Proc. of the 29th Annual Intern. ACM SIGIR Conf. on Research and Development in Information Retrieval. Seattle, USA, 2006. P. 639–640.

[18] Han J., Kamber M. Data mining: concepts and techniques (2nd ed.). Morgan Kaufmann Publishers, 2006. 800 p.

[19] Lebanon G. Metric learning for text documents // IEEE Trans. on Pattern Analysis and Machine Intelligence. 2006. Vol. 28, N 4. P. 497–508.

[20] Kim S., Zhang B.T. Genetic mining of HTML structures for effective Web-document retrieval // Applied Intelligence. 2003. Vol. 18, N 3. P. 243–256.

[21] Alguliev R.M., Aliguliyev R.M. Fast genetic algorithm for solving of the clustering problem of text documents // Artificial Intelligence. 2005. N 3. P. 698–707 (in Russian).

Received for publication 25 October 2006