A Uighur Automatic Summarization Method Based on Sub-theme Division

Xiaodong Yan(✉)

China National Language Resource Monitoring & Research Center Minority Languages Branch, Minzu University of China, Zhongguancun Street 27#, Haidian District, Beijing 100081, China
yanxd3244@sina.com

Abstract. As a very important research focus of natural language processing, automatic summarization can be used in many fields whether in improving the quality of searching results on a search engine or as a means of public opinion analysis. A method for Uighur automatic summarization is proposed in this paper which is base on sub-theme division and weight value. And by experiments, we find that it can get good precision and recall rates.

Keywords: Automatic summarization · Sub-theme division · Weight calculation

1 Introduction

In recent years, with the improvement of people's living standard, almost all minorities in our country began to use Internet. They obtain effective information resource on study, life and work and so on. Consequently many organizations and individuals have created websites to provide information services which are described by their own ethnic languages (including Uighur). Due to the existing Google, Baidu and other major search engines are not suitable for unique characteristics of the text of Uighur or other minorities language, search results of these Search Engine have never meet our expectations (search for Uighur in Baidu is not supported and the correlation between the pages of Google search results and the content of User query is very low, and Uighur text always is Confused with the other Arabic text). So they all cannot meet the majority of minority network users on information needs. Therefore searching and obtaining Uighur information quickly accurately, comprehensively and conveniently is the request of the information age.

Recently as the research hot spot, Internet public opinion analysis is widespread in concern. In particular minority language network public opinion analysis is valued by the national government. Automatic Summarization is also an indispensable method and way of public opinion analysis.

Y. Lu et al. (Eds.): ISCTCS 2014, CCIS 520, pp. 8–15, 2015.
DOI: 10.1007/978-3-662-47401-3_2

2 Related Researches

Automatic summarization research started early and has obtained a great amount of research results. In 1958, the United States, IBM's Luhn [1] in "The Automatic Creation of Literature Abstracts" first proposed automatic summarization. Early 1970s, Edmundson in University of Maryland proposed four weighting method [2, 3], Consolidating the words' weights in sentence, regarding their sum as the weights of this sentence, picking sentences as abstracts according to the weight. In 1989, U.S. GE Research Center Lisa F. Rauet developed a SCISOR system, the system generate the appropriate conceptual framework by analyzing the document theme and syntactic structure [4]. In 1995 a theme of "Summarizing Text" is published in special issue in international journals Information Processing & Management [5]. In 1991, Morris and Hirst presented the first computable model of vocabulary chain, it is word series consist of a set of adjacent words of a subject, provides important clues for dividing text structure and themes [6]. Then, Barzilay et have made other WordNet-based Lexical chain calculation method [7, 8]. Lexical chain can clearly represent the semantic relationships between words, providing an important basis for dividing text structure and analyzing themes. In 2004, the University of Michigan's Gunes Erkan etc. Put forward LexRank algorithm [9], it is a method calculating the weight of the sentence under graphical representation structure of text. Gunes Erkan achieved a abstracting system by using this method, and evaluate the system by using DUC2004 data sets, experimental results show that the system ranked first in the number of evaluation.

In Uighur automatic summarization a Uighur website automatic summary extraction method is proposed by Jepati which is based on statistics [10]. We learned from the past, all kinds of language automatic summarization methods and made a Uighur automatic summarization method based on the theme of division. In the following, we will describe it in detail.

3 A Uighur Automatic Summarization Method Based on Division of Sub-theme

3.1 Overview of the Method

A method for Uighur automatic summarization based on the division of sub-theme is presented in this paper. The main steps of the method are as following: text preprocessing, subtopics division, sentence weight calculation, redundant processing and summary generation.

3.1.1 Text Preprocessing

In preprocessing, first, clause and word segmentation is done. The stop words, rare words and modified word which is no practical significance in the text are all removed.

3.1.2 Sub-theme Division

In the division of sub-theme, first, vector space model is constructed by the unit of sentence, and the cosine of the angle between the vector is calculated as the element

values of text similarity matrix. Then according to the similarity matrix construct undirected weighted graph of the text, and the corresponding maximum spanning tree is obtained. Finally by using a modified K means clustering algorithm on maximum spanning tree, the clustering is completed and each sub-class we got represents a sub-theme.

3.1.3 Sentence Weight Calculation

When the weight of the sentence is calculated, each sub-theme will be represented as a graph structure and the LexRank Score of each sentence is calculated on sub-themes based on LexRank algorithm. Then the FeatureScored of each sentence is calculated. Finally, the weight of sentence is measured by the combination of these two parts.

3.1.4 Summary Generation

After the division of the sub-themes and the calculating weights of the sentence, Sub-themes are sorted from high to low according to the degree of importance. Then according to the order the highest weight of each sub-theme sentences is collected into the candidate Digest sentence set. In order to avoid repeat extracting a higher similar sentence next time, after extracting summary sentences from a sub-theme, the remaining weight of the sentence in this sub-theme is re-calculated.

The System Framework Figure of the method of Uighur document summarization based on the division of sub-theme which is mentioned in this paper is as shown in Fig. 1, and overall flow chart of it is shown in Fig. 2.

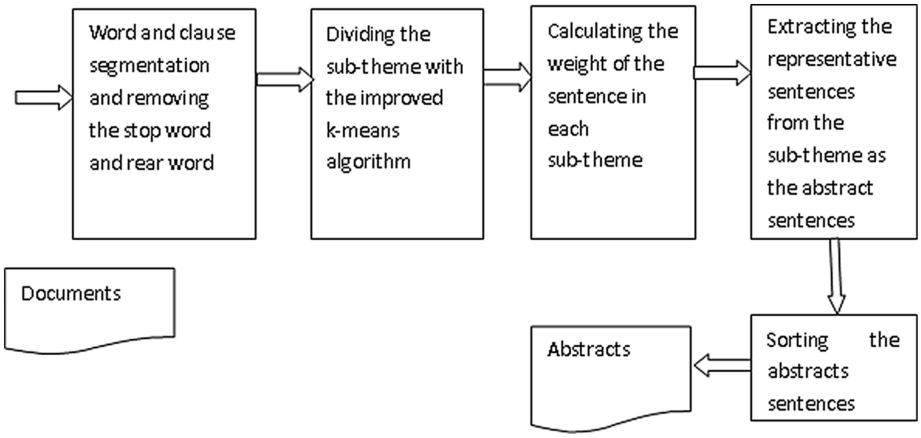

Fig. 1. System Framework Figure of the method of Uighur document summarization based on the division of sub-theme

3.2 Text Preprocessing

In the pretreatment process of the document, the main methods are speech tagging, filtering stop word and stemming. In addition to the above there are some other useful operations including deleting useless words, dictionary generation, text compression and stemming technology and so on.

- Speech tagging: nouns, verbs, adjectives, etc. can more accurately express a complete meaning and some parts of speech on expressing the central meaning of the article is not much useful. Such as conjunctions, Modal, function words, almost no meaning. So it is necessary to give nouns, verbs and other types of words weights. Of course, the introduction of part of speech tagging also has some disadvantages, not only increasing the execution time of the algorithm, but also not ensure that the chosen words can maximize express the central meaning of the article. However, after making part of speech tagging the amount of remaining keyword is further reduced. For the subsequent operations of feature extraction and the time and space to calculate weights can be reduced accordingly. After the part of speech tagging the number of each sentence and the location of each word in the sentence will be record.
- Filtering stop word: If the frequency of occurrences of the word is high but it hardly express the meaning of the text, the word is called stop words (stop words).such as "ھەم ", "بىلەن ","يەن ". If these words are in the vector representation, the dimension of feature vectors would be increased. Thereby the complexity of the algorithm will be increased. So first we will create stop word list, when in operation we will filter the stop words appeared in the text one by one, but the meaningful words are retained to build text vector model.
- Stemming: In order to improve the efficiency and accuracy of feature extraction we must do stemming in the title and text. Same meaning words are merged into a stem, for example, the following few words in Uyghur are all expressed "China", "جۇڭگونى", "جۇڭگوغا", "جۇڭگونىڭ", "جۇڭگو", "جۇڭگودىن ", so the words above are unified as "جۇڭگو ". After stemming the number of each sentence and the location of each word in the sentence will be record.

3.3 Division of Sub-Theme

In the research of our automatic summarization in this paper, in order to improve theme-coverage of Abstracts and reduce generation of redundancy, there are three major steps to achieve the subtheme division in this paper: the construction of similarity matrix, the generation of the largest tree and the clustering division of maximum spanning tree.

For a document, there are many construction methods of inter-sentence similarity matrix, such as the Hamming distance method, the absolute value of the reciprocal method, Euclidean distance method, scalar product method, absolute value of the index, the correlation coefficient method, geometric mean minimum, maximum minimum method, cosine method, and the arithmetic average minimum method. In this paper we use the cosine method to construct similarity matrix, as follows:

The text is divided into a set of sentences, expressed as S = (S1, S2, S3, … Sn), and as a set of samples which is waiting for be classified. Each sentence is a sample;

Each sentence is expressed as a vector whose components are the weight of feature words, the weights are calculated by using TFIDF algorithm, then we suppose the representation of m-dimensional vector of any two sentences Si and Sj is:

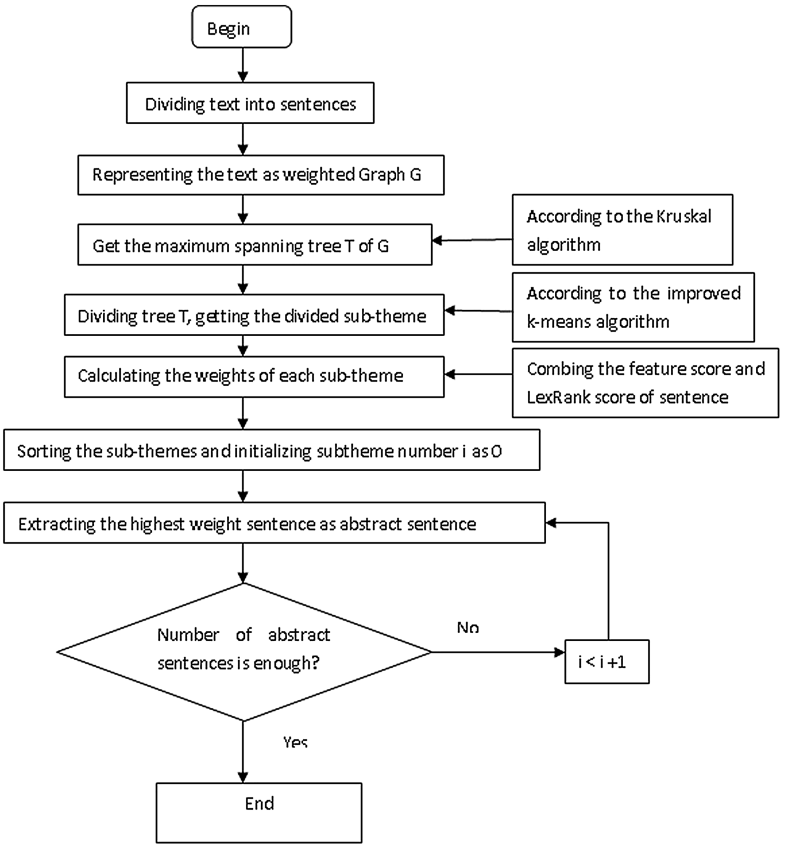

Fig. 2. Overall flow chart of a Uighur document summarization method based on the division of sub-theme

$$\mathrm{Si} = (t_{i1}, t_{i1}, \ldots, t_{im}),\ \mathrm{Sj} = (t_{j1}, t_{j2}, \ldots, t_{jm}),$$

Calculating the distance between two vectors according to the cosine equation, as the values of similarity matrix element, the formula is as follows:

$$r_{ij} = \frac{\sum_{k=1}^{m} x_{ik} x_{jk}}{\sqrt{\sum_{k=1}^{m} x_{ik}^2 \sum_{k=1}^{m} x_{ik}^2}} \tag{1}$$

Through the above steps, we can construct the similarity matrix of text, thus mading the foundation for the following generation and division of largest tree.

3.4 Calculation of Sentence Weight

After completion of the sub-theme division of the document, the next step is to determine the weight of sentences in each sub-theme and to provide the basis for extraction of abstract sentence. In our method, measurement of the weight of the sentence not only considers the sentence LexRankscore which is calculated by the LexRank algorithm, but also considers the importance of the sentence itself features, such as sentence length, location, sentence structure, clues word or title words and so on.

Next it will be described in detail. After completing sub-theme division of document, we can obtain sentences collection of each sub-theme. For each sub-theme, we can use the following procedure to calculate sentences LexRankscore based on LexRank algorithm:

① Each sentence within the sub-themes is expressed as vector space model whose component is the weights of feature words

② Calculating similarity between any two sentences in sub-theme by vector cosine;

$$sim(d_1, d_1) = \frac{\sum_{i=1}^{n} w_{1i} w_{2i},}{\sum_{i=1}^{m} w_{1i}^2 \sum_{i=1}^{n} w_{2i}^2} \tag{2}$$

③ Taking the sentences in subtheme as the vertices. Using the similarity between sentences to measure the edge weights between vertices. Constructing corresponding graph structure of subtopics.

④ According to the graph structure of the sub-theme, iteratively calculating the significant value of each vertex using LexRank algorithm within subtheme until variation of significant degree value of each vertex is less than a threshold value Threshold. The final significant value as an indicator of measuring sentence importance denoted LexScore. Calculated as follows:

$$p(u) = \frac{d}{N} + (1 - d) \sum_{v \in adj(u)} \frac{w(u, v)}{\sum_{z \in adj(v)} w(z,v)} p(v) \tag{3}$$

Wherein, p (u): significant value of vertex u;

Adj (u): a collection of vertices which adjacent to vertex u;
W (u, v): Edge weights between vertex u and vertex v, I.e., the similarity between corresponding sentences;
D: damping, located between [0.1, 0.2];
N: The total number of vertices which is in the graphic structure of sub-theme.

⑤ Repeating the above procedure, until sentences significant values of all subtheme are calculated.

Through the above steps, we can get LexRank algorithm based significant value sentence LexScore, the value is a measure of the overall importance of the sentence. It can be used as a important measure of the sentence weight.

3.5 Summary Generation

We can get sentences weights of all the sub-theme by subtheme division and the calculation method of sentences weight. Then according to the importance of sub-themes and the weight of each cub-topic sentences, we can complete the extraction of abstracts, which is a very critical step for automatic abstraction. Because extracting summary sentences generally have compression ratio (number of abstract sentence / number of sentence of the document), such as compression rate of 10 % or 20 %, the number of abstracts sentence is limited. In addition, each sub-theme of the document is not always important, therefore, when extracting summary sentences, in order to ensure that abstracts sentence can cover the important topics that descript in the document there will need to sort sub-themes before extraction according to the level of importance, so we can begin extracting from the most important sub-themes. Apparently, the importance of the sub-themes is decided mainly by the degree of importance of the sentences it contained. Therefore, in this article, we will take the sum of the weights of each sub-theme sentences as a measure of the importance of the sub-themes. Suppose there are n sentences in the sub-theme, the weight of sentences sf is represented by the form of (sf), then the importance of the sub-themes TopicScore can be calculated by the following formula:

$$TopicScore(T_k) = \sum_{i=1}^{n} w(s_i) \tag{4}$$

We can get the degree of importance of each sub-theme TopicScore by Eq. 4, and according this value we can sort the sub-theme in descending order. When the required abstraction sentence has a limited number, it can give priority to extract the highest weight sentences in the most important sub-theme. Thus ensuring the abstraction sentences can cover the important theme that expressed in the document. This paper presents a method of abstract generation, before extract abstract, all sub-themes are sorted in descending order according to the importance, then the sentence of each sub-theme are sorted in descending order by weight.

4 Experimental Results and Analysis

We compared the new abstract extraction method used in this system with Traditional abstract extraction method based on statistical [10]. In it we use the formula (5) to determine the precision and recall rate for comprehensive evaluation. As can be seen from Table 1, there is a good increase on the recall rate and precision rate in the new approach based on sub-theme division.

$$\text{Recall rate} = \frac{|s_a \cap s_r|}{s_r} \quad \text{precision rate} = \frac{|s_a \cap s_r|}{s_a} \tag{5}$$

Table 1. Comparisons of the experimental results based on traditional statistical method and on the sub-theme

results fields	Recall rate		Accuracy rate	
	Traditional approach	New approach	Traditional approach	New approach
Politics	0.56	0.72	0.45	0.68
News	0.50	0.55	0.46	0.51
Economy	0.51	0.63	0.52	0.58

Acknowledgement. The work in this paper is supported by the National Natural Science Foundation of China project "Research on Basic Theory and Key Technology of Cross Language Social Public Opinion Analysis"(61331013).

References

1. Luhn, H.P.: The automatic creation of literature abstracts. IBM J. Res. Dev. **2**(2), 159–165 (1958)
2. Edmundson, H.P.: New methods in automatic extracting. J. ACM **16**(2), 264–285 (1969)
3. Edmundson, H.P., Wyllys, R.E.: Automatic abstracting and indexing-survey and recommendations. Commun. ACM **4**(5), 226–234 (1961)
4. Ran, L.E., Jacobs, P.S., Zemik, U.: Information extracting and text summarizationusing lingusitic knowledge acquisition. Inf. Process. Manage. **25**(4), 419–428 (1989)
5. Jones, K.S., Brigitte, E.N.: Introduction:automatic summarizing. Inf. Process. Manage. **31** (5), 625–630 (1995)
6. Morris, J., Hirst, G.: Lexical Cohesion computed by thesaural relations as an indicator of the structure of text. Comput. Linguist. **17**, 21–48 (1991)
7. Elhadad, M.: Using lexical chains for text summarization. In: Proceedings of the Workshop on Intelligent Scalable Text Summarization, pp. 10−17. Madrid, Spain (1997)
8. Alam, H., Kumar, A., Nakamura, M.,et a1.: Structured and unstructured document summarization: design of a commercial summarizer using lexical chains. In: The 7th International Conference on Document Analysis and Recognition. pp. 1147−1152. UK, Edinburgh, Scotland (2003)
9. Gunes, E., Radev, D.R.: LexRank: graph-based centrality as salience in text summarization. J. Artif. Intell. Res. **22**(12), 457–479 (2004)
10. A Jepati Corneille mention, Venera - Mu Shajiang: Research of statistics-based Uighur website Automatic Extraction of summary. Artificial Intelligence and Recognition Technology, 7(1): 185–188 (2011)

http://www.springer.com/978-3-662-47400-6

Trustworthy Computing and Services
International Conference, ISCTCS 2014, Beijing, China,
November 28-29, 2014, Revised Selected papers
Yueming, L.; Xu, W.; Zhang, X. (Eds.)
2015, XII, 414 p. 206 illus., Softcover
ISBN: 978-3-662-47400-6