Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 24 (2013) 274 – 279

Procedia Computer Science

17th Asia Pacific Symposium on Intelligent and Evolutionary Systems, IES2013

Learning to Predict the Need of Summarization on News Articles

Ji Eun Lee, Hyun Soo Park, Kyung Joong Kim* , Jae Chun No

Dept. of Computer Engineering, Sejong Univ., Seoul, Republic of Korea

Abstract

Recently, we live with a huge amount of data. For example, we have great amount of news articles everyday. But there are small amount of useful information in the articles and it is hard to extract useful information manually. As a result, there are lots of news articles but, it is hard to read all of articles and find informative news manually. One of solutions on this problem is to summarize texts in the article. There are many studies on the text summarization techniques, but small number of studies to predict whether the article should be summarized or not. If we don't know about that, it is likely to waste computing resources to summarize unnecessary articles. In this paper, we propose a method to model the pattern of user's summarization needs on news articles. We perform experiments using news articles and apply data mining techniques (C4.5 and Naïve Bayes) to model common preprocessor to execute the automatic summarization. Finally, we can get some meaningful results on the "desire to summarize" prediction.

Selection and peer-review under responsibility of the Program Committee of IES2013

Keywords: article summarization; natural language processing; data mining; decision tree, naïve bayes

1. Introduction

Recently, many people and devices generate massive data and its size is continuously growing exponentially. One representative case is news articles. There are a lot of media companies and they produce news competitively everyday. People face with an overflow of news articles in a day. Therefore, people often fail to acquire useful

* Corresponding author.
E-mail address: apple173@naver.com (Ji Eun Lee), hspark@sju.ac.kr (Hyunsoo Park), kimkj@sejong.ac.kr (Kyung Joong Kim), jano@sejong.ac.kr(Jae Chun No).

1877-0509
Selection and peer-review under responsibility of the Program Committee of IES2013
doi:10.1016/j.procs.2013.10.051

information easily and even it is hard to distinguish useful news from unnecessary articles. So, it has been a significant problem to acquire information you want from massive news articles.

Automatic summarization is helpful to solve this problem. A summary of something is a short account of it, which gives the main points but not the details. Studies about summarization have been a popular topic to enable user access information effectively by discarding irrelevant parts. While there are many studies on the summarization methods, but we hardly see specific studies on the prediction of summarization needs [1]. If we can estimate what articles need to be summarized, there is a big advantage to save computing resources on large data. Many text summarization techniques are relatively complex and require a computational/data intensive job. Because of this, if we want to apply the text summarization on huge amount of data, probably we need a lot of computational resources. But if we can predict what article should be summarized in a selective manner and vice versa, we can expect saving of unnecessary computational resource.

Intuitively, we can guess that if an article is too long, people want to get summary. But we do not know the appropriate length of articles to be summarized and what is the good metric in order to measure the article length. In order to investigate on the issue, we collect data from users. In the experiment, we collect readers' desire (this article should be summarized/this article should not be summarized) and learn prediction models using data mining tool WEKA [2]. As a result, we can generate a prediction model which predicts with precision almost 90%.

2. Related works

Table 1. Summary of related works

Author	Year	Description
R. Barzilay, M. Elhada [3]	1997	Topic, representation (Lexical Chain)
C. Lin, E. Hovy [4]	2000	Topic, representation (Topic Signature)
O. Buyukkokten, H. Garcia-Molina, A. Paepcke [5]	2001	Summary for mobile device
J. M. Conroy [6]	2001	Selection (Hidden Markov Models)
Y. Gong, X. Liu [7]	2001	Topic, ranking method (Latent Semantic Analysis)
G. Erkan, D. R. Radev [8]	2001	Representation (Stochastic Graph)
L. Antiqueira, O. N. Oliveira Jr., *et al* [9]	2009	Representation (Complex Network)
M. A. Fattah, F. Ren [10]	2009	Selection (various Machine Learning techniques)
R. M. Aliguliyev [11]	2009	Sentence similarity measure

Text summarization is the one of the most important topics in natural language processing. Sometimes, we can regard the text summarization system as a text-to-text system [1]. This system outputs text shorter than imputed text. This system consists of three parts (1) transform inputted text to intermediate representation, (2) score each sentence and (3) select importance sentences. We can analyze many works in this frame. In most of cases, each work proposes new representation, new scoring method and/or new selection methods.

In the early days, many researchers find a topic in given text and score by its importance [3, 4, 7]. But, nowdays, many works use indicator instead topic [8-11]. In this approach, they compare the importance of each sentence directly, instead of searching for the topic or interpreting the sentences.

Table 1 is the summary of related works. This table summarized authors, published year and their main contribution.

3. The proposed method

For the experiment, we get experimental data from graduated students using news articles. There are four steps in our approach (Fig. 1). (1) We collect URL of an article selected by users for the summarization. If a student thinks this article should be summarized then presses the 'necessary' button on the screen while browsing web sites. (2) We

extract the attributes from the collected data. (3) We apply data mining techniques using data mining tool WEKA [2] and generate a decision model. (4) Finally, we measure the generalization ability of the model using unseen data.

Fig. 1. An overview of the proposed method

3.1. News article

News articles are typically made up of the subject and body. In the body, there are pictures, text and advertisements (ads). Ads should be excluded because usually people ignore that and it is not relevant to articles. However, it is difficult to remove ads completely. Because of the boundary between advertising and article body text is ambiguous, and each site has a different form. So we collect data from only one Korean news portal (http://www.nate.com). There are many attributes in an article body text. Characters, words, sentences and paragraphs can be the attribute. Fig. 2 shows an example of an article and attributes.

(a) An example of an article

Attributes

Necessary (Class)	No
Gender	Male
# of character in title	8
# of word in title	3
# of character in text	873
# of word in text	203
# of sentence text	14
# of paragraph text	15

(b) An example of attributes

Fig. 2. An example of article and attributes

3.2. Attributes

We have a simple survey to find good attributes. We used google drive for the survey. 80 people participated in the survey. We summarize the information of the participants in Table 2. From the results of the survey, we choose the seven attributes shown in Fig. 2 (b). The chosen attributes were used in the experiment. However, the following

attributes gain small attention from the participants as the preprocessor of summarization. They are pictures and newspaper company, because it is difficult to be recognized automatically or less relevant with current articles.

Attributes are divided into three categories as follows. “Characters” counts the number of alphabets, numbers, special characters and white spaces. “Words” are separated by a space. “Sentence” counts the number of period. We assume that articles do not have other end marks-question mark, exclamation mark and so on. Finally, “paragraph” counts new line marks such as “\n”.

Table 2. The information of the survey participants

Gender	Age	Job
Male 50 (63%)	20 – 30 (76%)	Students 50 (63%)
Female 30 (38%)	Over 30 (24%)	Workers 30 (37%)

Total number of participants: 80

4. Experimental results

4.1. Data collection and preprocessing

We use three programs: data collection program, article collector and article preprocessor. (1) We use the data collection program to collect many URLs from people. The program has two buttons (necessary / unnecessary). Five graduate students participated in experiments and they push a button when they read news articles. The program records their choices and this record contains the label of each instance. Table 3 is the condition of this experiment.

Table 3. The information of the experimental condition

Test people	5 students
Experimental period	About 3 weeks
Button click timing	Immediately after browsing the page/prior to peruse the article

(2) The second program is an article collector to collect articles’ text from the URLs. There is a problem that it is hard to perfectly extract only article text. That is very difficult and each news web site has different forms. So we use one Korean news portal (http://www.nate.com). We use “Jericho” HTML parser [13] to process HTML format text. Jericho parser is open source and we can handle HTML files easily. Through this simple process, we can get the article text, except for the most advertising and tags. But, we can not remove advertisement completely. So, we refine data manually to know our approach’s maximum performance, and compare its results with automatic data refinements.

(3) Finally, we use a preprocessor program to get the attributes extracted from the article text. The attributes are based on characters, words, and so on. Through this process, we can extract important attributes for the data mining tool.

4.2. Machine learning with WEKA

WEKA is a popular suite of data mining software written in JAVA. It contains a collection of visualization tools and algorithms for data analysis and predictive modeling. We use this tool to model our experimental data.

Table 4 shows the number of data and its distribution. This data was made by 5 participants who selected each article and pressed the button. The total number of data is 433 instances, the number of “unnecessary” is 281 and the number of “necessary” is 152. Table 5 shows experimental results (Cross-validation, ten-folds). We use decision tree C4.5 (J48) and Naïve Bayes classifier algorithms. With the manual data refinement, accuracy is about 90% / 91%

(J48/NaiveBayes) and even without the manual data refinements, accuracy is about 88% / 89% (J48/Naïve Bayes). We can predict readers' desire to get "summarization" service with high precision, regardless of the imperfect preprocessing. And there are still rooms for additional performance improvement if we can remove improper contents more precisely.

In Table 4, there are only 35 % of articles labeled as "necessary" and we can predict with alomost 90%. It means that we can save more than 50% of orignal computing resource requeirement (in case of summarizing all articles). In the worst case, if the prediction model predicts incorrectly the "unnecessary" to "necessary" (approximately 10% wrong), it requires only 35 + 10 = 45% computing resources.

Table 4. The information of participants

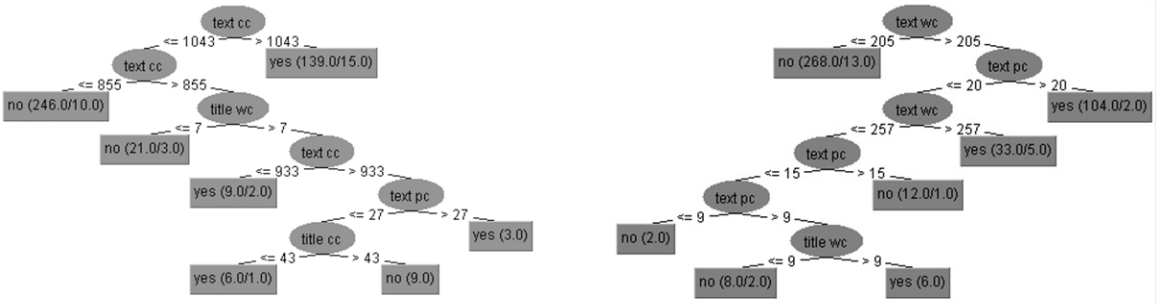
Data	# of instances
Necessary	152 (35%)
Unnecessary	281 (65%)
All	433

Five people collected the data.

Table 5. The result of accuracy

	Decision tree (C4.5)	Naïve Bayes
With manual data refinement	90%	91%
Without manual data refinement	88%	89%

Fig. 3 shows an example of learned decision trees with different preprocessing (without manual data refinement, with manual data refinement). In this decision tree, we identify # of characters in text is the most important in without manual refinement case and # of words in text is the most important in with manual data refinement case. Intuitively, # of characters in text = average # of characters in one word × # of words in text. So it might be almost the same results. These two decision trees show decision boundary of yes/no (necessary/unnecessary). In Fig. 3 (a) case, the most important decision boundary exists between # of characters is 855 and 1043. And in Fig. 3 (b) case, # of words is between 205 and 257.



(a) Without manual data refinement- C4.5 (b) With manual data refinement – C4.5

Fig. 3. Example of learned decision trees (text cc: # of characters in text, text wc: # of words in text, text pc: # of paragraphs in text, title cc: # of character in title, title wc: # of words count in title)

5. Conclusions and future works

In this paper, we present a method to predict the user's desire on the "summarization" service based on some training samples with readers' explicit labeling. Although there are many text summarization techniques, there are very small number of studies on the prediction issue. If we can realize what kind of article should be summarized for user's needs, it is possible to save computing resources.

We collect data from five graduate students using news articles. While browsing news articles, the participants (students) to get summarization service is recorded by pressing a button. After the data collection, we model their desire using data mining tool (WEKA). Using this method, we can model the pattern of service request and the model's precision is nearly 90%. However, the precision of model might be decreased because of the imperfect preprocessing (removal of the advertisement). So, we get the "ideal" maximum performance by removing the ads

manually. As a result, the learned model achieves over 90% accuracy. We can predict readers' desire easily with high precision. It means that we can save a lot of computing resource to summarize news articles.

However, there are some limitations of this work. First, we considered only one news portal. Web sites have different forms and placement of the advertisement and it is difficult to remove irrelevant contents. To expand this work, we can use multiple data source to get better conclusion. Secondly, we have attributes that are not considered because of the difficulty of processing. We choose most of attributes depending on the count of raw text. But there are also important attributes like word frequency, meaning of a text, importance of news articles and so on. If we can apply more complex attributes, it is possible to predict user's desire more precisely.

Acknowledgements

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIP) (2013R1A2A2A01016589, 2010-0018950).

References

[1] A. Nenkova, K, McKeown, 2012, A Survey of Text Summarization Techniques, *Mining Text Data, Springer US*, pp. 43-76.
[2] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, I. H. Witten, 2009, "The WEKA Data Mining Software: An Update," SIGKDD Explorations 11, pp. 10-18.
[3] R. Barzilay, M. Elhadad, 1997, "Using Lexical Chain for Text Summarization," In Proc. of the ACL Workshop on Intelligence Scalable Text Summarization, pp. 10-17.
[4] C. Lin, E. Hovy, 2000, "The Automated Acquisition of Topic Signatures for Text Summarization," In Proc. of the 18th Conf. on Computational Linguistics 1, pp. 495-501.
[5] O. Buyukkokten, H. Garcia-Molina, A. Paepcke, 2001, "Seeing the Whole in Parts: Text Summarization for Web Browsing on Handheld Devices," In Proc. Of the 10th International conf. On World Wide, Web, pp. 652-662.
[6] J. M. Conroy, 2001, "Text Summarization via Hidden Markov Models," In Proc. Of the 24th Annual International ACM SIGIR Conf. on Research and Development in Information Retrival, pp. 406-407.
[7] Y. Gong, X. Liu, 2001, "Generic Text Summarization using Relevance Measure and Latent Semantic Analysis," In Proc. of the 24th Annual International ACM SIGIR Conf. on Research and Developement in Information Retrival, pp. 19-25.
[8] G. Erkan, D. R. Radev, 2004,"LexRank: Graph-based Lexical Centrality as Salience in Text Summarization," Journal of Artificial Intelligence Research 22, pp. 457-479.
[9] L. Antiqueira, O. N. Oliveira Jr., L. da F. Costa, M. das G. V. Nunes, 2009, "A Complex Network Approach to Text Summarization," Information Science 179, pp. 584-599.
[10] M. A. Fattah, F. Ren, 2009, "GA, MR, FFNN, PNN and GMM based Models for Automatic Text Summarization," Computer Speech and Language 23, pp. 126-144.
[11] R. M. Aliguliyev, 2009, "A New Sentence Similarity Measure and Sentence based Extractive Technique for Automatic Text Summarization," Expert systems with Application 36, pp. 7764-7772.
[12] V. Gupta, G. S. Lehal, 2010, "A Survey of Text Summarization Extractive Techniques," Journal of Emerging Technologies in Web Intelligence 2, pp. 258-268.
[13] Jericho Parser, http://jericho.htmlparser.net/doc/index.html