The Best Feature Parameter and HMM for Text Summarization

Mohammad Golam Sohrab[1], Mohamed Abdel Fattah[1,2], Fuji Ren[1,3]

1 Faculty of Engineering, University of Tokushima
2-1 Minamijosanjima
Tokushima, Japan 770-8506
2 FIE, Helwan University, Cairo, Egypt
3 School of Information Engineering, Beijing University of Posts & Telecommunications
Beijing, 100088, China
(sohrab, mohafi, ren) @is.tokushima-u.ac.jp

Abstract. This work investigates different text features to select the best one and proposes an approach to address automatic text summarization. This approach is a trainable summarizer, which takes into account several features, including sentence position, sentence centrality, sentence resemblance to the title, sentence inclusion of name entity, sentence inclusion of numerical data and sentence relative length for each sentence to generate summaries. First we investigate the effect of each sentence feature on the summarization task. Then we use all features score function to train Hidden Markov Model (HMM) in order to construct a text summarizer model. The proposed approach performance is measured at several compression rates on a data corpus composed of 50 English articles from the domain of politics.

1. Introduction

With the huge amount of information available electronically, there is an increasing demand for automatic text summarization systems. Text summarization is the process of automatically creating a compressed version of a given text that provides useful information for the user. Text summarization addresses both the problem of selecting the most important portions of text and the problem of generating coherent summaries. There are two types of summarization: extractive and abstractive. Extractive summarization methods simplify the problem of summarization into the problem of selecting a representative subset of the sentences in the original documents. Abstractive summarization may compose novel sentences, unseen in the original sources. However, abstractive approaches require deep natural language processing such as semantic representation, inference and natural language generation, which have yet to reach a mature stage nowadays.

The process of text summarization can be decomposed into three phases: analysis, transformation, and synthesis. The analysis phase analyzes the input text and selects a few salient features. The transformation phase transforms the results of analysis into a summary representation. Finally, the synthesis phase takes the summary representation, and produces an appropriate summary corresponding to users' needs. In the

© G. Sidorov, B. Cruz, M. Martínez, S. Torres. (Eds.)
Advances in Computer Science and Engineering.
Research in Computing Science 34, 2008, pp. 153-161

Received 21/03/08
Accepted 26/04/08
Final version 30/04/08

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. As the compression rate decreases, the summary will be more concise; however, more information is lost. While the compression rate increases, the summary will be more copious; relatively, more insignificant information is contained.

Recently many experiments have been conducted for text summarization task. Some were about evaluation of summarization using relevance prediction [1], ROUGEeval package [2], SUMMAC, NTCIR, and DUC [3] and voted regression model [4]. Others were about single- and multiple-sentence compression using "parse and trim" approach and a statistical noisy-channel approach [5] and conditional random fields [6]. Some other researches were about multi-document summarization [7], [8] and summarization for specific domains [9] - [11].

In this work, sentences of each document are modeled as vectors of features extracted from the text. The summarization task can be seen as a two-class classification problem, where a sentence is labeled as "correct" if it belongs to the extractive reference summary, or as "incorrect" otherwise. We may give the "correct" class a value '1' and the "incorrect" class a value '0'. In testing mode, each sentence is given an analog value between '0' and '1'. Therefore, we can extract the appropriate number of sentences according to the compression rate. The trainable summarizer is expected to "learn" the patterns which lead to the summaries, by identifying relevant feature values which are most correlated with the classes "correct" or "incorrect". When a new document is given to the system, the "learned" patterns are used to classify each sentence of that document into either a "correct" or "incorrect" sentence and give it a certain score value between '0' and '1', producing an extractive summary.

2. Text Features

1- f1 = Sentence Position.

We assume that the first sentences of a paragraph are the most important. Therefore, we rank a paragraph sentence according to their position. For instance, the first sentence in a paragraph has a score value of 1, the second sentence has a score *(n-1)/n*, since *n* is the total number of sentences in the document under consideration.

2- f2 = Sentence Centrality (similarity with other sentences).

Sentence centrality is the vocabulary overlap between this sentence and other sentences in the document. It is calculated as follows:

$$Score_{f_2}(s) = \left| \frac{Keywords\ in\ s \cap Keywords\ in\ other\ sentences}{Keywords\ in\ s \cup Keywords\ in\ other\ sentences} \right| \quad (1)$$

Where, *S* is the sentence under consideration.

3- f3 =Sentence Resemblance to the title.

Sentence resemblance to the title is the vocabulary overlap between this sentence and the document title. It is calculated as follows:

$$Score_{f_3}(s) = \left| \frac{Keywords\ in\ s \cap Keywords\ in\ title}{Keywords\ in\ s \cup Keywords\ in\ title} \right| \quad (2)$$

4- f4 = sentence inclusion of name entity (proper noun).

Usually the sentence that contains more proper nouns is an important one and it is most probably included in the document summary. The score of f6 is calculated as follows:

$$Score_{f_4}(s) = \frac{\#(proper\ nouns\ in\ s)}{Length(s)} \quad (3)$$

5- f5 = sentence inclusion of numerical data.

Usually the sentence that contains numerical data is an important one and it is most probably included in the document summary. The score of f7 is calculated as follows:

$$Score_{f_5}(s) = \frac{\#(numerical\ data\ in\ s)}{Length(s)} \quad (4)$$

6- f6 = sentence relative length.

This feature is employed to penalize sentences that are too short, since these sentences are not expected to belong to the summary. We use the relative length of the sentence, which is calculated as follows:

$$Score_{f_6}(s) = \frac{Length(s) * \#(article\ sentences)}{Length(article)} \quad (5)$$

3. The Proposed Automatic Summarization Model

Figure 1 shows the proposed automatic summarization model. We have two modes of operations:

1- Training mode where features are extracted from 50 manually summarized English documents and used to train the HMM.

2- Testing mode where features are extracted from 50 English documents (These documents are different from that were used for training) and go through the HMM to be summarized.

The use of Hidden Markov model as a classification tool is motivated by the fact that it is very rich in mathematical structure and hence can form the theoretical basis for use in a wide range of applications [15].

Any sentence in a certain document may be classified as one of two types; summary sentence or no summary sentence. Sentence type prediction is about trying to guess what the next sentence type is based on the observations of the sentence's feature parameters. The statistical model for sentence type prediction is constructed. We collect statistics on what the sentence S_n type is like (summary or non summary sentence) depending on what the previous sentences (S_{n-1}, S_{n-2} ...) were. Therefore, the following conditional probability is considered:

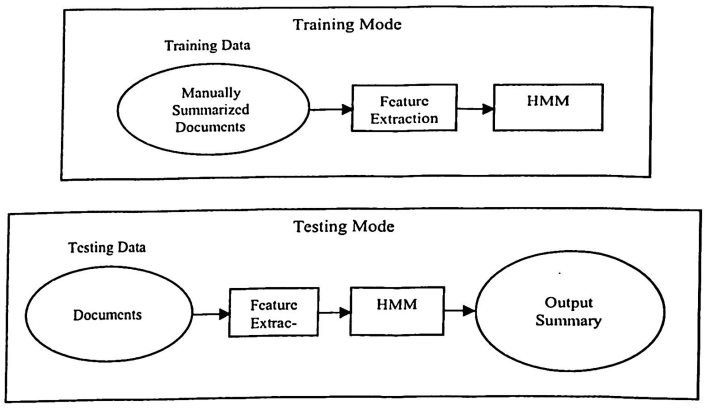

Fig. 1. The proposed automatic summarization model

$$P(S_n \mid S_{n-1}, S_{n-2}, \ldots, S_1) \tag{6}$$

Using Markov assumption to simplify the above probability:

$$P(S_n \mid S_{n-1}, S_{n-2}, \ldots, S_1) = P(S_n \mid S_{n-1}) \tag{7}$$

The probability of a certain sequence $\{S_1, S_2, \ldots, S_n\}$ may also be expressed using Markov assumption as follows:

$$P(S_1, S_2, \ldots, S_n) = \prod_{i=1}^{n} P(S_i \mid S_{i-1}) \tag{8}$$

Such an automaton would look like shown in figure 2.

Equation 8 is the Markov model. The only information of a certain sentence is the six feature parameters mentioned in section 2. These feature parameters are observable while the actual sentence type is hidden. Finding the probability of a certain sentence type $S_i \in \{Summary, NonSummary\}$ can only be based on the observation F_i. This conditional probability $P(S_i \mid F_i)$ can be written according to Bayes' rule:

$$P(S_i \mid F_i) = \frac{P(F_i \mid S_i)P(S_i)}{P(F_i)} \tag{9}$$

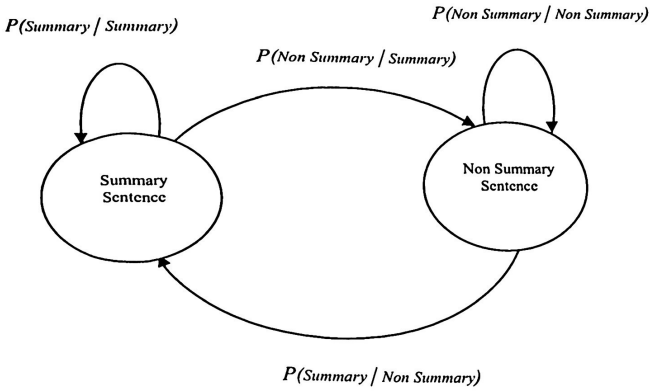

Fig. 2. Markov model for the summary non summary with state transition probabilities

Or for n sentences $\{S_1,...,S_n\}$, as well as n feature sequence $\{F_1,...,F_n\}$, This conditional probability can be:

$$P(S_1,...,S_n \mid F_1,...,F_n) = \frac{P(F_1,...,F_n \mid S_1,...,S_n)P(S_1,...,S_n)}{P(F_1,...,F_n)} \tag{10}$$

The probability $P(F_1,...,F_n \mid S_1,...,S_n)$ can be estimated as $\prod_{i=1}^{n} P(F_i \mid S_i)$, if we assume that, for all i, the S_i, F_i are independent of all F_j and S_j, for all $j \neq i$.

We want to draw conclusions from our observations about the sentence type. We can therefore omit the probability $P(F_1,...,F_n)$. We get a measure for the probability, which is proportional to the likelihood L as follows:

$$P(S_1,...,S_n \mid F_1,...,F_n) \alpha L(S_1,...,S_n \mid F_1,...,F_n) = P(F_1,...,F_n \mid S_1,...,S_n).P(S_1,...,S_n) \tag{11}$$

With the Markov assumption it turns to:

$$L(S_1,...,S_n \mid F_1,...,F_n) = \prod_{i=1}^{n} P(F_i \mid S_i).\prod_{i=1}^{n} P(S_i \mid S_{i-1}) \tag{12}$$

The classification of a sentence type is based on the above likelihood value.

4. Experimental Results

4.1. The English Data

100 English articles in the domain of politics were collected from the Internet archive. 50 English articles were manually summarized (by two subjects' agreement) using compression rate of 30%. These manually summarized articles were used to train the previously mentioned model (HMM). The other 50 English articles were used for testing (these articles were also manually summarized). The average number of sentences per English articles is 31.6.

We use the intrinsic evaluation to judge the quality of a summary based on the coverage between it and the manual summary. We measure the system performance in terms of recall from the following formula:

$$R = \frac{S \cap T}{T} \tag{13}$$

Where, R is the recall, T is the manual summary and S is the machine-generated summary.

4.2. Simple Lead Approach

The lead method is known to be effective for document summarization of newspapers in lower compression ratio. This approach is a simple one that is based on extracting a set of the first sentences in the document based on the compression ratio.

Table 1 shows the summarization recall associated with the lead approach for different compression rates.

4.3. The Effect of each Feature on Summarization Performance

In this section, we investigate the effect of each feature parameter on summarization by using the individual feature parameter scores. For instance, to investigate the effect of the first feature (sentence position) on summarization performance, we use the following equation:

$$Score(s) = Score_{f_1}(s) \tag{14}$$

Table 2 shows the summarization recall associated with each feature for different compression rates for the English documents.

Table 1. The lead approach performance evaluation based on recall

Compression rate (CR)	10%	20%	30%
Recall (R)	0.1750	0.2726	0.3846

Table 2. The summarization recall associated with each feature for different compression rates

Compression rate (CR)	10%	20%	30%
R(f1)	0.1783	0.2709	0.4066
R(f2)	0.3802	0.4213	0.4910
R(f3)	0.3874	0.4269	0.5012
R(f4)	0.1933	0.2942	0.4429
R(f5)	0.1800	0.2819	0.4164
R(f6)	0.2266	0.3740	0.4432

4.4. The Results using the Sum of all Normalized Feature Parameters

In this section, we take the summation of all normalized feature parameters associated with the sentence under consideration to calculate its score value from the following formula:

$$Score(s) = Score_{f_{1\_n}}(s) + Score_{f_{2\_n}}(s) + Score_{f_{3\_n}}(s) + Score_{f_{4\_n}}(s) + Score_{f_{5\_n}}(s) + Score_{f_{6\_n}}(s) \quad (15)$$

Since the six items of the above equation are the normalized feature parameters. Table 3 shows the summarization recall for different compression rates.

4.5. The Results of Hidden Markov Model (HMM)

The system is extracting features from the sentences of 50 English manually summarized documents and uses them to construct Hidden Markov model for each category (we have 2 categories). Use the other 50 English documents as a testing set. Then apply the sentences of these documents as inputs to the Hidden Markov model after feature extraction step as follows:

1- Extract features from the sentences of the document.
2- Construct the feature vector F.
3- Use this feature vector as an input of the HMM.
4- Save the output of the HMM for each sentence.
5- Rank all document sentences based on their scores then arrange them in a descending order.
6- Chronologically select the set of sentences of highest scores based on the required compression rate.

Table 4 shows the results of HMM for the 50 English articles.

4.6. Discussion

Although it is well known that the simple lead approach gives good results in case of newswire, it gives reasonable results for political documents as shown in table 1.

Usually, the document title conveys the main topic of this document. Therefore, f3 (sentence resemblance to the title) which is the vocabulary overlap between this sentence and the document title gives the best results over all other feature parameter results. It is reasonable, since the sentence that has a maximum overlap with the document title should convey the most important part in the article. It is also clear from table 2 that f2 (centrality) also gives good results since it conveys the vocabulary overlap between this sentence and other sentences in the document. The lowest results are associated with f1 (sentence position). f5 (sentence inclusion of numerical data) does not give high recall since most of political articles do not contain many numerical data. Therefore, the system arranges the article sentences that do not contain numerical data chronologically (based on sentence position).

The sum of all normalized feature parameters approach results outperforms the simple lead approach results as shown in tables 1 and 3.

Table 3. The sum of all normalized feature parameters approach performance evaluation based on recall

Compression rate (CR)	10%	20%	30%
Recall (R)	0.2950	0.3807	0.4843

Table 4. The HMM approach performance evaluation based on recall

Compression rate (CR)	10%	20%	30%
Precision (P)	0.3957	0.4368	0.5634

The results of HMM approach outperforms the results of lead and sum of all normalized feature parameters approaches.

In general the recall value is directly proportional to the compression ratio as illustrated in all the tables of this work. The recall significantly decreases with the decrease of the compression ratio for the lead, feature 1, feature 2 and feature 5 approaches while it slightly decreases with the decrease of compression ratio for the HMM, feature 2 and feature 3 approaches.

5. Conclusions and Future Work

In this work, we have investigated the use of Hidden Markov Model (HMM) for automatic text summarization task. We have applied our new approach on a sample of 50 English political articles. Our approach results outperform the baseline approach results. Our approach has been used the feature extraction criteria which gives researchers opportunity to use many varieties of these features based on the used language and the text type. Some text features are language independent.

In the future work, we will investigate the effect of the output summary from this system on information retrieval and cross language information retrieval systems.

Acknowledgment
This research has been partially supported by the Japan Society for the Promotion of Science (JSPS), Grant No. 07077 and the Ministry of Education, Science, Sports and Culture, Grant-in-Aid for Scientific Research (B), 19300029.

References

1. Hobson, S., Dorr, B., Monz, C., & Schwartz, R.: Task-based evaluation of text summarization using Relevance Prediction Information Processing & Management, 43(6), (2007), 1482-1499.
2. Sjöbergh, J.: Older versions of the ROUGEeval summarization evaluation system were easier to fool. Information Processing & Management, 43(6), (2007), 1500-1505.
3. Over, P., Dang, H., & Harman, D.: DUC in context. Information Processing & Management, 43(6), 1506-1520.
4. Hirao, T., Okumura, M., Yasuda, N., & Isozaki, H. (2007). Supervised automatic evaluation for summarization with voted regression model. Information Processing & Management, 43(6), (2007), 1521-1535.
5. Zajic, D., Dorr, B., Lin, J., & Schwartz, R.: Multi-candidate reduction: Sentence compression as a tool for document summarization tasks. Information Processing & Management, 43(6), (2007), 1549-1570.
6. Nomoto, T.: Discriminative sentence compression with conditional random fields. Information Processing & Management, 43(6), (2007), 1571-1587.
7. Vanderwende, L., Suzuki, H., Brockett, C., & Nenkova, A.: Beyond SumBasic: Task-focused summarization with sentence simplification and lexical expansion. Information Processing & Management, 43(6), (2007), 1606-1618.
8. Harabagiu, S., Hickl, A., & Lacatusu, F.: Satisfying information needs with multi-document summaries. Information Processing & Management, 43(6), (2007), 1619-1642.
9. Moens, M.: Summarizing court decisions. Information Processing & Management, 43(6), (2007), 1748-1764.
10. Reeve, L., Han, H., & Brooks, A.: The use of domain-specific concepts in biomedical text summarization. Information Processing & Management, 43(6), (2007), 1765-1776.
11. Ling, X., Jiang, J., He, X., Mei, Q., Zhai, C., & Schatz, B.: Generating gene summaries from biomedical literature: A study of semi-structured summarization. Information Processing & Management, 43(6), (2007), 1777-1791.
12. Russell, S. J., & Norvig, P.: Artificial intelligence: a modern approach. Englewood Cliffs, NJ: Prentice-Hall International Inc (1995).
13. Yeh, J., Ke, H., Yang, W., & Meng. I.: Text summarization using a trainable summarizer and latent semantic analysis. Information Processing & Management, 41(1), (2005), 75-95.
14. Jann, B.: Making regression tables from stored estimates. Stata Journal 5, (2005), 288–308.
15. Rabiner, L.: A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. Proceddings of the IEEE, 77(2), (1989), 257–286.