2011 Eighth International Conference on Information Technology: New Generations

Two-Step Sentence Extraction for Summarization of Meeting Minutes

Jae-Kul Lee
Computer Science and Engineering
Kyungpook National University
Daegu, Korea
Email: jklee@sejong.knu.ac.kr

Hyun-Je Song
Computer Science and Engineering
Kyungpook National University
Daegu, Korea
Email: hjsong@sejong.knu.ac.kr

Seong-Bae Park
Computer Science and Engineering
Kyungpook National University
Daegu, Korea
Email: sbpark@sejong.knu.ac.kr

***Abstract*—These days a number of meeting minutes of various organizations are publicly available and the interest in these documents by people is increasing. However it is time-consuming and tedious to read and understand whole documents even if the documents can be accessed easily. In addition, what most people want from meeting minutes is to catch the main issues of the meeting and to understand its contexts rather than to know whole discussions of the meetings. Existing text summarization techniques applied to this problem often fail because they are developed without considering the characteristics of the meeting minutes. In order to improve the performance of summarization of meeting minutes, this paper proposes a novel method for summarizing documents based-on two-step sentence extraction. It first extracts the sentences which are addressing the main issues. For each issue expressed in the extracted sentences, the sentences related with the issue are then extracted in the second step. Then, by transforming the extracted sentences into a tree-structure form, the results of the proposed method can be understood better than existing methods. In the experiments, the proposed method shows remarkable improvement in performance and this result implies that the proposed method is plausible for summarizing meeting minutes.**

***Keywords*-Text mining, Document summarization**

I. Introduction

These days many government organizations including congress open to the public their minutes, and a numerous of people are concerned with these documents due to various reasons. A meeting minute is a document that records everything related with the meeting such as a procedure, contents, results and etc. A large part of a minute is the utterances of meeting attendants. Although these documents can be easily accessed since they are recently digitalized, it is a very time consuming task to read whole contents of document. This is because most minutes are very long and even contain unnecessary utterances. Therefore, many or most readers want to grasp the important point of a meeting or the opinion of each member. For instance, the readers want to know who is positive or negative about a specific issue. In order to satisfy needs, the research for automatic summarization of meeting minutes is required.

Automatic document summarization has been of interest for a long time to researchers in natural language processing, and the summarization of meeting minutes can be regarded as a specific case of document summarization. However, meeting minutes have several characteristics which are different from those of the documents in other domain. First, there is a moderator who operates the meeting. Typically, she controls the procedure of the meeting and mentions main issues directly. Therefore, by analyzing the utterances of a moderator, the essential points of the meeting can be extracted simply.

Second, there are several words which play an important role in the meeting. These words can be used for grasping the essential points efficiently and helping to extract salience sentences more accurately. Third, there exists a dependency among utterances since the recorded utterances are normally a conversation among the meeting attendants. In addition, even if a meeting has several issues, an utterance has a dependency with just one of the issues. If it is possible to analyze the dependencies among the utterances and between an utterance and a main issue, it can be useful for understanding the flow of a meeting.

This paper proposes a novel summarization method for meeting minutes. By considering characteristics of meeting minutes, the proposed method consists of two extraction phases and a dependency analysis phase. For catching main issues of a meeting, a text ranking which is based on TextRank algorithm [1] is suggested in the first step. This step extracts only the sentences that are regarded as the representation of main issues. The second step collects the sentences that are relevant to each main issue based on a similarity between a sentence and a main issue. In the final step, dependency relations among the extracted sentences are analyzed and a tree-like representation of summarized text is constructed for better understanding of the relations.

The rest of this paper is organized as follows. In Section 2, previous research for document summarization is explained. Section 3 describes the proposed method for summarizing minutes in detail. Section 4 shows the experimental setup for proposed method and interprets results. Finally, Section 5 draws a conclusion and mentions future work.

II. Related Work

Document summarization is a task for reducing the complexity of texts while keeping the fundamentals of document

978-0-7695-4367-3/11 $26.00 © 2011 IEEE
DOI 10.1109/ITNG.2011.210

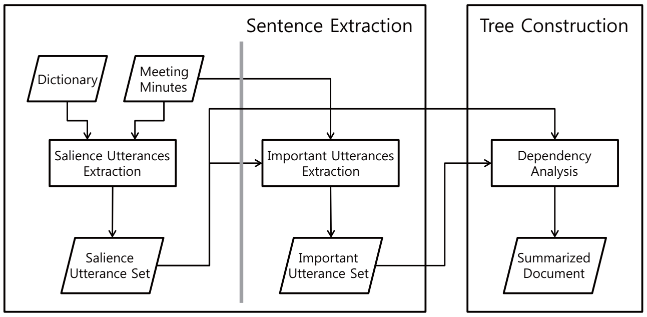

Figure 1. Procedure of proposed summarization model

contents [2]. This is a traditional task in natural language processing and information retrieval. Many researchers have been concerned with this task. Normally, document summarization is divided into two categories: an extraction summarization and an abstraction summarization [3]. Extraction summarization approach only extracts the essential segments of a document such as phrases, sentences, or paragraphs, which are regarded to contain the fundamentals of the document. This approach can be classified into two subcategories again according to the methods used: a discourse-level approach and a surface-level approach.

Abstraction summarization requires an additional processing of paraphrasing to generate a final result. This paraphrasing helps to write out more understandable results. However, it requires many difficult natural language processing techniques to paraphrase the extracted sentences, and these techniques are not perfect yet. As a result, the extraction summarization is more general than the abstraction summarization.

Recently, Mihalcea [1] proposed TextRank algorithm that is a variant of PageRank algorithm [4]. This algorithm is to determine ranks of sentences based on graph-structure. In this algorithm, a graph of documents is constructed by shared words in each sentence. The important sentences are then extracted by applying PageRank algorithm to this graph. TextRank does not require any sentences for training since it is an unsupervised learning method. It also has a good point that there is no limitation on languages. Thus, it has been applied to a number of researches. Especially, a recent research on Korean document summarization reported high performance [5].

Query-based text summarization is a user-centric method. This method extracts the sentences, which are relevant to queries of users [6], [7]. Sanderson proposed a user-directed summarization technique using local context analysis based on the research retrieval system, INQUERY. However, this method failed in outperforming other methods that do not use any query expansion [6]. Thus, for improving the performance of this approach, Goldstein proposed a query expansion technique by adding new features like as the title of a document, opinion feedback, and so on [7].

III. Method for Summarizing Meeting Minutes

The summarization of meeting minutes proposed in this paper is done as shown in Figure 1. The document summarization can be regarded as a ranking problem in viewpoint of machine learning. Thus, the first-step extracts a set of salience sentences spoken by a moderator based on TextRank, a well-known ranking method. These sentences are used as representative texts of main issues discussed in the meeting. In the second-step, the important sentences of attendants of the meeting are collected based on similarities between a sentence and the set of salience sentences, which is extracted in the first-step. The third step is a dependency analysis step. In this step, the dependency analysis among the extracted sentences is done. As its result, a tree-like representation of the summarized sentences is made.

A. Extraction of Salience Sentences

The first step is to extract the representative texts of main issues. In this step, the utterances of a moderator are first extracted, and then a set of salience sentences is made from the extracted utterances. For this task, the ranking algorithm proposed in this paper computes scores of each sentence. For training of ranking algorithms, a number of training data are needed. The meeting minutes must be manually labeled, since there are no standard data for meeting minutes. However, it requires very high human efforts to collect the annotated data, which is impossible to label large volume of data in most cases. Therefore, to reduce human efforts, this paper adopts an unsupervised learning method, TextRank algorithm, in this step. TextRank algorithm shows fine performance, although it does not require any data for training. TextRank is a ranking algorithm based on a graph-structure. Therefore, in order to apply it to text summarization, sentences must be represented as a formal graph, $G = (V, W)$. This graph, G, is a weighted undirected graph with a set of vertices V and a set of weighted edges W. For representing a document as a graph, each sentence is regarded as a vertex, v_i. A weighted edge between v_i and v_j, w_{ij}, is created if v_i and v_j share common terms. Based on this graph, G, TextRank computes the score of each node, $TR(v_i)$, as follows:

$$TR(v_i) = (1-d) + d \times \sum_{v_j \in in(v_i)} \frac{w_{ji}}{\sum_{v_k \in out(v_j)} w_{jk}} TR(v_j)$$

where d is a damping factor and w_{ji} is a weighting value of an edge between i-th and j-th nodes. $in(v_i)$ is a set of vertices that point to v_i, and $out(v_i)$ is a set of vertices that are pointed from v_i. It is not easy to estimate the best value for damping factor d. Thus it is selected empirically.

The weighting value of an edge, w_{ji}, is defined as similarity between two sentences. In this paper, the similarity between two sentences is defined by the ratio of common

Algorithm 1 The salience sentences extraction

```
Input:
  • $V_m$: set of utterances of the moderator
  • $D$: significant words dictionary
  • $W$: set of weighted edge
  • $r$: arbitrary initial value
  • $p$: significant word weight
  • $x$: extraction proportion
for each $v_i \in V_m$ do
  $TR\,(v_i) = r$
end for
while until $TR$ converge do
  for each $v_i \in V_m$ do
    $U = in\,(v_i)$
    $sum = 0$
    for each $v_j \in U$ do
      $sum += TR\,(v_j)$
    end for
    $TR\,(v_i) = sum$
  end for
end while
for each $v_i \in V_m$ do
  $T =$ the term set of $v_i$
  for each $t_i \in T$ do
    if $t_i \in D$ then
      $TR\,(v_i) = (1 + p) \times TR\,(v_i)$
    end if
  end for
end for
$V'_m = extract\,(V_m; x)$ // Extract high ranked utterances.
$S = combine\,(V'_m)$ // Combine utterances between paragraphs.
return S
```

terms, n_k, between two sentences and it is computed as follows:

$$Similarity\,(v_i, v_j) = \frac{|\{n_k | n_k \in v_i \wedge n_k \in v_j\}|}{log\,(|v_i|) + log\,(|v_j|)}$$

Terms in a sentence can be defined as various kinds of elements such as a character, a syllable, a word, and so on. In this paper, only nouns are selected as terms since the research on a graph-based Korean document summarization achieved best performance with noun-noun similarities [5], [8]. An arbitrary value is first assigned to each node, and it is updated iteratively until its convergence.

Normally, some words have more significance than other general words in summarization. For reflecting the significance of these words into TextRank, a dictionary of significant words is built. If a sentence has some words within the dictionary, its score is then determined as follows:

$$Score(v_i) = \begin{cases} (1+p) \times TR(v_i) & \text{if } n_k \in D \wedge n_k \in v_i \\ TR(v_i) & \text{otherwise} \end{cases}$$

where D is a set of significant words and p is a weighting parameter to represent the significant degree of a selected word, n_k. This parameter is set between 0 and 1. Then, the sentences with high score are selected as salience sentences.

In a meeting, it is often for an attendant to speak several utterances at once, even if these utterances are related with a single issue. Thus, the salience sentences in the same issue should be clustered into a single unit. Since it is believed that the sentences from a paragraph have the same issue, they are clustered into a set. That is, a set S_k of the salience sentences is made for each k-th paragraph. A sentence, v_i is contained in the same set S_k with other sentence, v_j, if v_i and v_j belong to a same paragraph. That is,

$$S_k = \{v_i | \forall j, paragraph\,(v_i) = paragraph\,(v_j)\}$$

Finally, a set of paragraphs, $S = \{S_i\}$, is created as a result. The overall procedure of the first extraction step is given in Algorithm 1.

B. Extraction of Important Sentences

The second step extracts important utterances which are relevant to the salience sentences of a moderator. In this step, the score value of each sentence is defined by similarity between the sentence and each set of salience sentences, S_i, which are selected in previous step. There could be various similarities between two texts, but this paper adopts Bag-of-Words (BoW) model and cosine similarity. As a feature set for representing a document with BoW model, only nouns are used. The similarity between each set of salience sentences, S_i, and each utterance, v_j, of attendants except the moderator is defined as follows:

$$Sim\,(S_i, v_j) = \frac{S_i \cdot v_j}{\|S_i\|\,\|v_j\|}$$

where $\|\cdot\|$ is the norm of a vector.

This paper assumes that an utterance of each member is related only with its preceding moderator's utterance. Thus, the computation of the similarity is done only between an utterance and a set of salience sentences preceding it. This step extracts the important sentences according to the similarity in a similar way of the first step. Algorithm 2 shows a detailed procedure for extracting important utterances for each salience sentence.

C. Utterances Dependency Analysis

Dependency analysis among extracted important sentences is the final phase of the proposed method. In this step, all paragraphs of a moderator and other attendants are transformed into a tree representation form. In order to find

Algorithm 2 Important sentences extraction

```
Input:
  • $S$: paragraph set of salience sentences
  • $V'$: set of utterances of attendants except the moderator
  • $x$: extraction proportion
for each $S_i \in S$ do
  $SV_i = \{v_i | v_i \in V'$ and $loc(S_i) < loc(v_i) < loc(S_{i+1})\}$ // $loc(\cdot)$ returns spatial index of a sentence or a paragraph.
  for each $v_i \in SV_i$ do
    $score(v_i) = Sim(S_i, v_i)$
  end for
end for
$V'' = extract(V'; x)$
$M = combine(V'')$ // $M = \{M_i\}$
return M
```

Algorithm 3 Tree construction procedure

```
Input:
  • $S$: paragraph set of salience sentences
  • $M$: paragraph set of attendants utterances
$P : S \cup M$
$T$ : Tree
$root$ : root node of $T$
for each $P_i \in P$ do
  if $P_i \in S$ then
    $root.addChild(P_i)$
  else
    $A = ancestor(P_{i-1})$ // $ancestor(\cdot)$ returns a set of nodes on the path from the node to root.
    $node = \arg\max_{A_j} Sim(A_j, P_i)$
    $node.addChild(P_i)$
  end if
end for
return T
```

relationships among important sentences within a paragraph, a cosine similarity is used in this step.

For constructing a tree-structured form, S_i's, sets of salience sentences are treated as top-level nodes of a tree and all important sentences of each S_i are considered as child nodes of their top-level node. The depth of a child node is determined according to the similarity between it and its salience sentences or preceding important sentences. The detail dependency analysis is explained in Algorithm 3.

IV. Experiments and Evaluation

A. Data Set

For evaluation of the proposed method, a series of experiments are performed with a data set of meeting minutes. Knowledge Managements System of the National Assembly of Korea [9] is used for collecting the data set since there is no gold-standard data set for this task. In the experiments, 36 documents among from 276th to 291th assembly records of 18th Nation Assembly are used.

The average number of sentences in each document is 1,524 and the average number of words is about 10,932. One annotator summarizes all the documents manually, since Knowledge Managements System does not provide summarized documents that can be used as an answer set. These summarized texts contains about 20∼25% of sentences of original documents.

B. Evaluation Measure

In this paper, three kinds of performance measurements are selected. The first measurement is ROUGE-N (Recall-Oriented Understudy for Gisting Evaluation) which is used as a standard measurement in document summarization area [10]. ROUGE-N is defined as

$$ROUGE-N = \frac{\sum_{S \in A} \sum_{gram_n \in S} count_{match}(gram_n)}{\sum_{S \in A} \sum_{gram_n \in S} count(gram_n)}$$

where A is a set of answer sentences and $count_{match}(gram_n)$ is a function which returns the number of common n-gram between the results of a summarized method and the answers. Among various size of ROUGE method, this paper uses only ROUGE-1 method.

The second measurement is a F-measure based on a sentence unit because the proposed method is a sentence-based method. As a final measurement, a new method for evaluating performance of tree-like representation is adopted. It measures the ratio of shared common paths between two trees, answer tree and constructed tree by proposed method. The accuracy of a result tree, $T_R = (N_R, E_R)$, is defined as follows:

$$accuracy = \frac{\sum_{n \in \{N_R \cap N_A\}} I(P(n, T_R), P(n, T_A))}{|N_R \cap N_A|}$$

where N_R is the set of nodes which belong to the result tree and N_A is a node set of an answer tree. I is an indicator function and $P(n, \cdot)$ is a function which returns a path from a root node of each corresponding tree to a target node, n.

C. Experimental Results

In order to show the improvements of the proposed method, the proposed method is compared with two existing summarization methods. The first method is TextRank algorithm. During the comparison with TextRank, only TextRank is used in summarization without any outer information. The second method is a combination model of TextRank algorithm and a weighting method. This second method performs a summarization task by only using method, which is explained in section 3.1. In experimental setup, a common

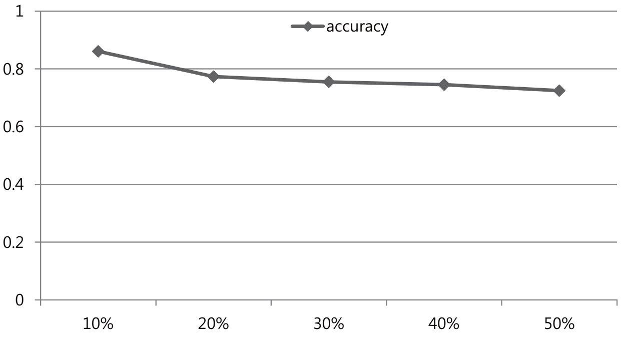

Figure 4. Accuracy of three-structure

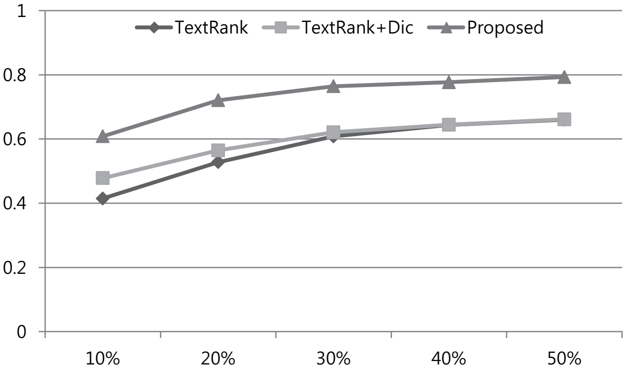

Figure 2. ROUGE-1 values of three methods

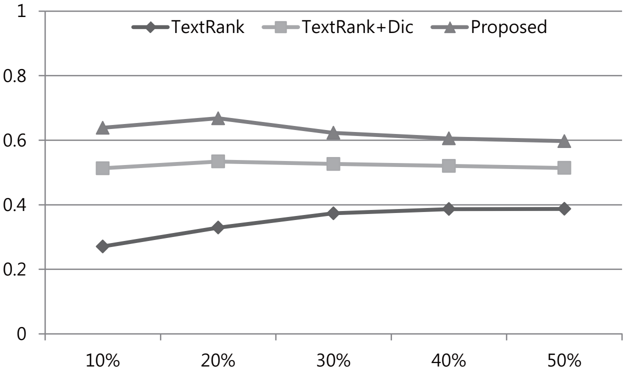

Figure 3. F_1-measure values of three methods

parameter, damping factor d, of three methods is set to be the same value, 0.85.

A number of experiments are done with various summarization ratios. In most cases, about 50% sentences are a zero score in second step. Thus, if the extraction ratio is over than 50%, there is no way to choose important sentences except random selection. Thus results are shown with lower than 50% of extraction ratio.

Figure 2 shows the ROUGE-1 values of three methods. The proposed method achieves the highest performance among all three methods regardless of the ratio. A sub-part of the proposed approach, the combination of TextRank and weighting, also shows better performance than TextRank algorithm, although the difference of performances is not large when the ratio is over 30%.

Figure 3 represents F-measure values of each method. Even in this figure, the proposed method shows the highest F-measure regardless of the ratio. The proposed method shows around two times higher F-measure than TextRank algorithm. Moreover, unlikely as in ROUGE-1, the gap between TextRank and the combination model is also quite large. These experimental results imply that the proposed method is plausible for summarizing meeting minutes.

Finally, Figure 4 shows accuracy of the tree structured results. According to this figure, the ratio of summarization increases, the accuracy decreases. The new measurement for evaluating a structure considers the shared ratio of exactly matched paths-pair. Thus, if sentences which are not contained in an answer set exist in the tree, they cause significant negative impact on this measurement. However, although the accuracy decreases, it achieves more than 70% of accuracy regardless summarization ratio and it is quite a plausible result.

V. Conclusion and Future work

This paper proposed a novel method for summarizing meeting minutes. To achieve high performance of summarizing meeting minutes, the proposed method reflected their several characteristics. First, the proposed method has a two-step extraction procedure and higher performance is achieved than the existing methods by the procedure. According to the experiment results, the proposed method shows remarkable improvement in performance, and this result proves that the proposed method is plausible for summarizing meeting minutes. In addition, the proposed method offers more readable results by transforming the extracted sentences into a tree-structured form.

In this paper, we also showed that some words aid to select important sentences more accurately. However, it is a very difficult, time-consuming, and expensive task to manually construct set of useful words for all domains. Thus, as a future work, we will consider an automatic dictionary construction technique. In addition, the performance of tree construction method is not robust about increasing summarization ratio, even if the tree construction method showed a plausible result. In order to overcome this weak point, we will investigate a new tree construction method.

Acknowledgment

This work was supported by converging research center program through the converging research headquarter for human, cognition and environment, funded by the Ministry of Education, Science and Technology (2010K001130).

References

[1] R. Mihalcea and P. Tarau, “TextRank: Bringing Order into Texts,” in *Conference on Empirical Methods in Natural Language Processing*, Barcelona, Spain, 2004. [Online]. Available: http://acl.ldc.upenn.edu/acl2004/emnlp/pdf/Mihalcea.pdf

[2] I. Mani, *Automatic Summarization*, ser. Natural Language Processing. John Benjamins Publishing Company, 2001.

[3] O. Sornil and K. Gree-ut, “An automatic text summarization approach using content-based and graph-based characteristics,” in *Cybernetics and Intelligent Systems, 2006 IEEE Conference on*, Jun. 2006, pp. 1–6.

[4] S. Brin and L. Page, “The anatomy of a large-scale hypertextual web search engine,” *Comput. Netw. ISDN Syst.*, vol. 30, pp. 107–117, April 1998. [Online]. Available: http://dx.doi.org/10.1016/S0169-7552(98)00110-X

[5] J. pyo Hong and J. won Cha, “A korean important sentence extraction using textrank algorithms,” in *Proceedings of KISSE*, 2009.

[6] M. Sanderson, “Accurate user directed summarization from existing tools,” in *Proceedings of the seventh international conference on Information and knowledge management*, ser. CIKM ’98. New York, NY, USA: ACM, 1998, pp. 45–51. [Online]. Available: http://doi.acm.org/10.1145/288627.288640

[7] J. Goldstein, M. Kantrowitz, V. Mittal, and J. Carbonell, “Summarizing text documents: sentence selection and evaluation metrics,” in *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, ser. SIGIR ’99. New York, NY, USA: ACM, 1999, pp. 121–128. [Online]. Available: http://doi.acm.org/10.1145/312624.312665

[8] W. Song, Y. Kim, E. Kim, and M. Kim, “A document summarization system using dynamic connection graph,” *Journal of KISSE : software and applications*, vol. 36, no. 1, pp. 62–69, 2009.

[9] Knowledge managements system of the national assembly of the korea. [Online]. Available: http://likms.assembly.go.kr/record/index.html

[10] C.-Y. Lin and F. J. Och, “Automatic evaluation of machine translation quality using longest common subsequence and skip-bigram statistics,” in *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, ser. ACL ’04. Stroudsburg, PA, USA: Association for Computational Linguistics, 2004. [Online]. Available: http://dx.doi.org/10.3115/1218955.1219032