Multi-Document Summarization Using Graph-Based Iterative Ranking Algorithms and Information Theoretical Distortion Measures

Borhan Samei
Department of Computer Science
University of Memphis
Memphis, TN 38120 bsamei@memphis.edu

Marzieh Eshtiagh
Department of Computer Eng.
Shiraz University
Shiraz, Iran

Fazel Keshtkar
Department of Computer Science
Southeast Missouri State University
MO, USA

Sattar Hashemi
Department of Computer Eng.
Shiraz University
Shiraz, Iran

Abstract

Text summarization is an important field in the area of natural language processing and text mining. This paper proposes an extraction-based model which uses graph-based and information theoretic concepts for multi-document summarization. Our method constructs a directed weighted graph from the original text by adding a vertex for each sentence, and compute a weighted edge between sentences which is based on distortion measures. In this paper we proposed a combination of these two models by representing the input as a graph, using distortion measures as the weight function and a ranking algorithm. Finally, a ranking algorithm is applied to identify the most important sentences to be included in the summary. By defining a proper distortion measure and ranking algorithm, this model gains promising results on the DUC2002 which is a well known real world data set. The results and ROUGE-1 scores of our model is fairly close to other successful models.

Introduction

Automatic summarization is the process of reducing a text document or a larger corpus of multiple documents into a short set of sentences expressing the main meaning of the text. With today's massive growth of information and the enormous amount of text in the Internet, representing any topic, the phenomenon of information overload has lead to importance of access to coherent and correctly-developed summaries. These needs are a motivation to conduct research on this field and develop various summarization techniques.

An example of the use of summarization technology is search engines such as Google. One might want to check the latest news about a particular subject in a short time which can be done using a web-based news summarizer. There are other applications of text summarizers, such as inXight (LinguistX) (SAP) by which users can see the summaries when they move their mouse over a hypertext link to a document that has been previously summarized. Automated text summarization techniques has also been used for summarizing source code (Haiduc 2010).

Summarization approaches are often divided into two categories: *text abstraction* and *text extraction*.

- **Text abstraction** is to parse the original text in a deep linguistic way, interpret the text semantically into a formal representation, find new more concise concepts to describe the text and then generate a new shorten text, an abstract, with the same information content. Parsing and interpretation of a text is an old research area in which we have a wide spectrum of techniques and methods ranging from word by word parsing to rhetorical discourse parsing as well as more statistical methods or a mixture of all. This approach is however more challenging than the extraction based techniques.
- **Text extraction** means to identify the most relevant passages in one or more documents. The important parts are often retrieved, using standard statistically based information techniques augmented with more or less shallow natural language processing and heuristics methods (Luhn 1958). More advanced techniques consider the rhetorical structure (Marcu 1997) and semantic relationships (Gong and Liu 2001) and there are also some machine learning models (Kupiec, Pedersen, and Chen 1995; Ye et al. 2007). One of the disadvantages in above techniques is that they seem to ignore the redundancy and coverage in summarization.

Martin Hassel (Hassel 2007) proposes a model which considered avoiding redundancy using Random Indexing method (M.Sahlgren 2005). Moreover, cluster-based (Zha 2002) and centroid-based techniques (D. R. Radev and Tam 2004) have been investigated in recent years. There are also some models that use graph based algorithms (Mihalcea and Tarau 2005) or information theoretical techniques (Wan and Ma 2010). Our proposed model is an extraction based summarization technique, in which the original text is represented by a graph and by applying an iterative ranking algorithm based on information theoretical distortion measures our goal is to retrieve the most important parts of the text.

The resulting summaries vary depending on the ranking algorithm parameters, distortion measures and some defined thresholds. The proposed model is a multi-document summarization. Therefore, as the input we have clusters of documents (multiple documents related to a particular subject). Each cluster-document consists of approximately five articles then a summary is generated for each cluster. We test our model on DUC2002 (DUC 2002) data sets and the resulting summaries are evaluated by ROUGE-1.5.5 (Lin and E.Hovy 2003) toolkit.

The rest of this paper is organized as follows: Section two describes the related works on document summarization. Next Section introduces proposed approach and in Section four we show the experimental results and a comparison with others and finally, last Section concludes the paper and future works.

Related Work

Automatic text summarization reduces a text document or a larger corpus of multiple documents into a short set of sentences which expresses the main meaning of the text. By massive growth of information and the enormous amount of text on the Internet , researchers in NLP are more interested to explore new models for summarization and investigating a variety of approaches to come up with accurate summarization.

As we mentioned above, there are two categories defined for text summarization approaches, text abstraction and text extraction. In this section we explore some of these techniques and mention why our research is needed to fill the gap in text extraction techniques to avoide redundancy.

Random Indexing technique which was introduced by (M.Sahlgren 2005) is used statistical properties of the words, such as word's frequency to form a semantic representation of sentences which can be applied in extraction-based approaches to compare the summary sentences to avoid redundancy. Also cluster based models such as the model introduced by (Zha 2002) handled redundancy in several ways. Other techniques such centroid-based techniques (D. R. Radev and Tam 2004) that deal with redundancy were also investigated in recent research. (Mihalcea and Tarau 2005) model used graph based algorithms (Mihalcea and Tarau 2005) and specific problem formulations to cover the properties of the text. Since, extractive summarizers are more applicable with today's models, recent researches are mostly focused on this category.

In Hassel (2007), they divided the summarization procedure into three major steps: preprocessing, processing and generating the summary. Preprocessing step consists of stemming the text, omitting stop words, etc. After the preprocessing, the ranking algorithms are applied to rank the sentences based on their relevance or coverage of the main idea of text, and the final step is to generate a summary consisting the most important sentences while avoiding redundancy by Random Indexing (M.Sahlgren 2005) methods. Recent extractive approaches are more likely to avoid redundancy and maintain the relevance of the summary using cluster based models (Zha 2002; Radev and Tam 2004) or by certain formulations of the problem.

Wan and Ma (Wan and Ma 2010) presented a model for multi-document summarization based on information theoretical concepts and more or less of clustering techniques. This model considers document summarization as a transmission system assuming that the best summary should have the minimum distortion. Some popular distortion measures are used to cluster the sentences and determine their similarity. The proposed model in this paper is an extraction based summarization. In this model a graph was constructed based on the input sentences and the distortions between each two sentences were calculated. Regarding this structure, a ranking algorithm is applied with some predefined thresholds and constraints, to identufy the most important sentences based on their corresponding rank.

Proposed Approach

The proposed approach in this paper is a graph-based extractive summarization. In this approach the sentences were splited based on the punctuation marks that represent the end of a sentence (e.g. period, semicolon, etc). We removed stop-words to be excluded in the ranking procedure. The input documents were transferred into a directed weighted graph by adding a vertex for each sentence. Each two sentences were then examined by a distortion measure representing the semantic relation between them, and an edge was added between two sentences if the distortion was below a predefined threshold. This distortion measure is used to represent the semantic distance between nodes as the wight of the edges. The distortion measure used in our model is based on "Squared Error" which is a stasticial way of quantifying the difference between values which was introduced as a loss function by Friedrich Gauss(Lehmann and Casella 1998). The squared error is calculated by the equation: $SquaredError(x, y) = (y - x)^2$
Where x and y are both sentences and each sentence is represented as a bag of words. For words that appear in the sentence the values are set to their freauency in the whole document and 0 otherwise.

As mentioned earlier the stop-words are not considered in the whole process of summarization since such words are usually not relevant to the semantics of the sentences. Another important step in our summarization algorithm is the stemming of words which means to identify the words that has the same root such as "create" and "creation". Stemming makes the values assigned to each word based on its frequency more accurate, since the words from the same root counted as the same word. In our model we used Porter Stemmer (Porter 1980).

In order to compute the distortion of two sentences, each sentence is considered as a bag of words (excluding stop words) . Then a score is assigned to each word based on its frequency and the position of the sentence in the whole text. The distortion of two sentences is calculated based on the score of their words. The algorithm can be interpreted as follows:

1. Check each word in *sentence1* to see if it exists in *sentence2*. If the word X of *sentence1* does not exist in *sentence2*, square the score of word X and add to the sum

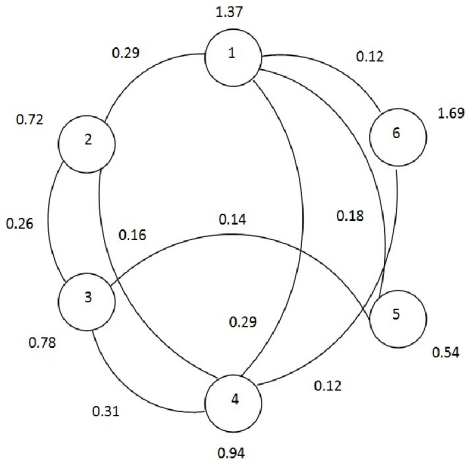

Figure 1: A graph of six sentences with edges representing the distortion. Values assigned to each node is the result of ranking algorithm.

and increase the number of not-common words by one.

2. In case the word X is common between *sentence1* and *sentence2*, calculate its frequency in *sentence2* and subtract it from the score of word X, then square and add to sum.
3. Then check the *sentence2* to find its not-common words with *sentence1*, in case the word Y is not in *sentence1*, square the score of word Y and add to sum and increase the number of not-common words by one.
4. At the end, calcualte the distortion between *sentence1* and *sentence2* by dividing sum by the number of not-common words.

Figure 1 illustrates a sample graph based on six sentences, each having an score showing its rank. The weight assigned to each edge is the squared error calculated by the above algorithm and the sentence are enumerated as follows:

1. I have a friend named Ken in England.
2. We often write to each other.
3. My letters are very short.
4. It is still hard for me to write in English.
5. I received a letter from Ken yesterday.
6. In his letter he mentioned that is waiting to visit me in England.

After the graph is built, the most important sentences are to be chosen to generate the summary. In order to find which sentences are more vital, an iterative ranking algorithm is applied to the graph. According to iterative ranking algorithm, each sentences is ranked based on its coverage of the whole content. In this research the pagerank algorithmPage and Brin (1998) which is mostly used for ranking webpages in search engines is adapted for the case of sentence ranking in an elegant way. Considering the desired length of the summary, then the vital sentences are chosen and added to the output summary. Pagerank is one of the most popular ranking algorithms, and was designed as a method for Web link analysis. Unlike other graph ranking algorithms, this algorithm integrates the impact of both incoming and outgoing links into one single node, then it produces a set of scores by the following equation:

$$PR(V_i) = (1-d)x \sum_{V_j \in I} \frac{PR(V_j)}{Out(V_j)} \quad (1)$$

Where d is a parameter set between 0 and 1. This parameter is used to add weight to the impact that adjacents of a node have on its rank. In our model the edges represent the distortion of sentences. More precisely, since we only add edges between sentences with a distortion below a predefined threshold, the more edges a sentence have the more likley it is to cover a major part of the text. On the other hand, the sentences with less or no edges may also be vital to the coverage of the summary as they may contain important information which was said only once or in a few sentences. A sentence with less number of edges is probably about a concept which is not overlaping with other sentences. The parameter d is used to make balance between these two kinds of sentences. Thus it is important to have a proper parameter d to balance the ranks and increase the coverage of the summary.

Starting from arbitrary values assigned to the rank of each node in the graph, the computation iterates until convergence below a given threshold is achieved. After running the algorithm, a score is given to each vertex, which represents the "importance" or "power" of that vertex within the graph. After ranking, the summary must be created from the top ranked sentences.

This algorithm is adjusted and used for multi-document summarization. A number of documents about the same topic are given as the input and the output is a summary of the given documents. Using the proposed algorithm, a single summary for each document is generated then a summary of summaries is created as the output summary. These steps are shown in Figure 2.

In the proposed approach, we attampt to make balanace between coverage and relevance of the summary by considering the weight of the edges are as distortion measures. Our graph-based approach is similar to Mihalcea & Tarau's (Mihalcea and Tarau 2005) approach. They also constructed constructed a graph by adding a vertex for each sentence in the text. Unlike our approach, the edges between vertices were established using sentence inter-connections and a link was drawn between any two sentences that share common content. The overlap of two sentences is determined as the number of common tokens between the lexical representations of two sentences, or it can be run through syntactic filters, which only count words of a certain syntactic category. We attemped to improve the coverage of summaries by considering the semantic difference between two sentences in a way that yeilds for coverage of the summary.

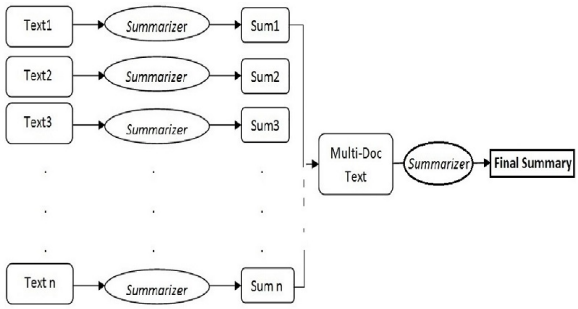

Figure 2: The process of generating summary for multi-documents. A single summary for each article is generated, then these summaries are aggregated and a text is created by adding these summaries one by one in their original order. The final summary is a summary of summaries and is created based on the first step.

Experimental Results

Datasets

We used DUC2002 dataset to evaluate the summarizer (DUC 2002). Document Understanding Conference (DUC) has organized yearly evaluation of document summarization. In DUC 2002, 59 document sets of approximately 10 documents each were provided and generic summaries of each document set with lengths of approximately 100 words or less were required to be created. Each document set consists of several articles written by various authors about a particular subject.

Evaluation Measure:

We use the ROUGE evaluation toolkit (Ye et al. 2007), which is adopted by DUC for automatic summarization evaluation. It measures summary quality by counting overlapping units such as the n-gram, word sequences and word pairs between the candidate summary and the reference summary. ROUGE-N is an n-gram recall measure computed as follows:

$$ROUG{-}N = \frac{\sum_{S \in RefSum} \sum_{ngram \in S} CountMatch(ngram)}{\sum_{S \in RefSum} \sum_{ngram \in S} Count(ngram)} \quad (2)$$

Where n stands for the length of the n-gram, and $Countmatch(ngram)$ is the maxi-mum number of *ngrams* co-occurring in a candidate summary and a set of reference summaries. *Count(ngram)* is the number of *ngrams* in the reference summaries. Among the evaluation methods implemented in ROUGE, ROUGE-N *(N=1, 2)* is relatively simple and works well in most cases. In our work we employ ROUGE-1 to score the summaries.

Results and Analysis

All the proposed models were evaluated on the DUC2002 dataset . Two variations of our model is tested and evaluated. Table 1 illustrates the evaluation of our model. The in-put documents are transferred into a directed weighted graph by adding a vertex for each sentence. Each two sentences are examined and the difference between them is calculated by distortion measures, and an edge is added between two sentences if the distortion of them is less than a predefined threshold. In the first model the same threshold is used to construct the graphs in both single and multi document summarization phase and it is set to infinity, however in the second model the threshold for the second phase is set to the average of sentences' distortions. As it is illustrated in Table I, the second approach gains better results and it is fairly close to best DUC2002 models. Table 2 illustrates the results of the PagerankW-U (Mihalcea and Tarau 2005), Minimum distortion (Wan and Ma 2010), and other top DUC2002 models. The scores are cited from their papers. According to ROUGE-1 scores, our model stands among the best DUC2002 models. The graph based algorithm used in this work was examined in several ways, and with a deeper look into the results it is found to be good in practice since the weight function is based on the difference of sentences' meaning, we see less redundancy and the coverage of the summaries is fairly enough. This model is also applicable for language independent summarizers and with a graph based modeling it opens up new developments by use of graph based popular algorithms such as shortest path or searches which are well known for their time and space complexities.

Another merit of this model, according to conducted results, is its ability to generate summaries with different lengths. The ranking algorithm sorts the sentences based on their importance so the system could choose a number of sentences for the summary based on the desired length. In the graph construction step-and-edge is added between two sentences if the distortion of them is below a predefined threshold. This helps us to avoid very large graphs and improves the ranking algorithm results, since in the ranking

Table 1: The ROUGE-1 results for mentioned models and top DUC 2002 systems. Model 1: Threshold set to infinity in both steps. Model 2: Threshold set to average distortion in second step.

Model	ROUGE-1
Evaluation of Model 1	0.3224
Evaluation of Model 2	0.3547

Table 2: The ROUGE-1 results for mentioned models and top DUC 2002 systems

Model	ROUGE-1
Minimum Distorsion	0.3588
PagerankW-U	0.3552
Team26	0.3515
Team19	0.3450
Team28	0.3436
Information Distance	0.2922

algorithm the number of out-going and incoming edges of a sentence is an important factor. In the meta-summary generation phase, this threshold is set to infinity, but in the final step where the meta-summaries are summarized this is set to the average distortion of the whole sentences. This technique improves the coverage and decreases the redundancy of the output summary since the rankings are based on the edges and with these thresholds the sentences which are similar in meaning are less likely to have similar ranks. There are several features still to consider which will probably lead to better results, and this model appears to be a good base line to apply more features and improvements depending on specific problem properties.

Conclusion and Future Work

Text summarization is one of the hot topics in NLP. Extractive based summarization approaches are mainly based on statistical analysis of the text, however researches have shown that by modeling the problem appropriately and with proper formulation, text-summarization could be handled by popular algorithms such as graph ranking or minimum distortion. In this paper we proposed a combination of these two models by representing the input as a graph, using distortion measures as the weight function and a ranking algorithm. The results and ROUGE-1 scores of our models is fairly close to other successful models.

This study opens up new research directions; first, extension of the distortion measures to define more proper functions with respect to the problem formulation. Second, incorporating semantic into the model and considering more features to rank the sentences and selection process. These models could be improved by considering other sides of the problem. For example, we could try to come up with a better function as the distortion measure specifically for this problem, instead of using the popular distortion measures. In these models we did not focused on the semantic analysis of the sentences. The results could be much better if the algorithms paid more attention to the meaning of the sentences and tried to understand the meanings.

In the meantime the distortion measures seem to potentially improvable. By considering more and more features for the summary and changing the distortion measures the results would change. Ranking algorithms are to be examined accurately. Perhaps we could design our own ranking algorithm which is more efficient for text summarization.

In future works, we plan to improve our model to be able to generate summaries of more complicated documents with a range of different sources. For example, summarizing the news about a particular topic from different news broadcasting services. With focus on different components of our models we plan to apply this approach in different areas of application.

References

D. R. Radev, H. Jingand, M. S., and Tam, D. 2004. Centroid-based summarization of multiple documents. In *Information Processing and Management*.

2002. *DUC. 2002. Document Understanding Conference*.

Gong, Y., and Liu, X. 2001. Generic text summarization using relevance measure and latent semantic analysis. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, 19–25.

Haiduc, S. 2010. On the use of automated text summarization techniques for summarizing source code. In *Reverse Engineering (WCRE)*.

Hassel, M. 2007. *Resource Lean and Portable Autamatic Text Summarization*. Ph.D. Dissertation, KTH School of Computer Science and Communication.

Kupiec, J.; Pedersen, J.; and Chen, F. 1995. A trainable document summarizer. In *SIGIR*.

Lehmann, E. L., and Casella, G. 1998. *Theory of Point Estimation*. Springer.

Lin, C.-Y., and E.Hovy. 2003. Automatic evaluation of summaries using n-gram cooccurrence statistics. In *HLT-NAACL (North American Chapter of the Association for Computational Linguistics)*.

Luhn, H. 1958. The automatic creation of literature abstracts. In *IBM Journal of Research and Development*.

Marcu, D. 1997. From discourse structures to text saummaries. In *ACL97/EACL97 Workshop on Intelligent scalable Text Summarization*.

Mihalcea, R., and Tarau, P. 2005. A language independent algorithm for single and multiple document summarization. In *Proceedings of IJCNLP*.

M.Sahlgren. 2005. An introduction to random indexing, methods and applications of semantic indexing. In *Workshop at the 7th international conference on Terminology and Knowledge Engineering TKE*.

Page, L., and Brin, S. 1998. The anatomy of a large-scale hypertextual web search engine. In *Computer Networks and ISDN systems*.

Porter, M. 1980. An algorithm for suffi stripping. In *Program: electronic library and information systems 14*.

Sap.

Wan, X., and Ma, T. 2010. Multi-document summarization using minimum distortion. In *IEEE International Conference on Data Mining*.

Ye, S.; Chua, T.-S.; M-Y.Kan; and Qiu, L. 2007. Document concept lattice for text understanding and summarization. In *Information Processing and Management*.

Zha, H. 2002. Generic summarization and keyphrase extraction using mutual reinforcement principle and sentence clustering. In *SIGIR02*.