Generic Text Summarization Using Relevance Measure and Latent Semantic Analysis

Yihong Gong
NEC USA, C & C Research Laboratories
110 Rio Robles Drive
San Jose, CA 95134
ygong@ccrl.sj.nec.com

Xin Liu
NEC USA, C & C Research Laboratories
110 Rio Robles Drive
San Jose, CA 95134
xliu@ccrl.sj.nec.com

ABSTRACT

In this paper, we propose two generic text summarization methods that create text summaries by ranking and extracting sentences from the original documents. The first method uses standard IR methods to rank sentence relevances, while the second method uses the latent semantic analysis technique to identify semantically important sentences, for summary creations. Both methods strive to select sentences that are highly ranked and different from each other. This is an attempt to create a summary with a wider coverage of the document's main content and less redundancy. Performance evaluations on the two summarization methods are conducted by comparing their summarization outputs with the manual summaries generated by three independent human evaluators. The evaluations also study the influence of different VSM weighting schemes on the text summarization performances. Finally, the causes of the large disparities in the evaluators' manual summarization results are investigated, and discussions on human text summarization patterns are presented.

Keywords

Generic Text Summarization, Relevance Measure, Latent Semantic Analysis

1. INTRODUCTION

The explosive growth of the world-wide web has dramatically increased the speed and the scale of information dissemination. With a vast sea of accessible text documents on the Internet, conventional IR technologies have become more and more insufficient for finding relevant information effectively. Nowadays, it is quite common that a keyword-based search on the Internet returns hundreds, or even thousands of hits, by which the user is often overwhelmed. Therefore, there is an increasing need for new technologies that can help the user to sift through vast volumes of information, and to quickly identify the most relevant documents.

With a large volume of text documents, presenting the user with a summary of each document greatly facilitates the task of finding the desired documents. Text search and summarization are the two essential technologies that complement each other. Text search engines return a set of documents that seem to be relevant to the user's query, and text summarizers produce document summaries that enable quick examinations through the returned documents. Text search engines serve as information filters that sift out an initial set of relevant documents, while text summarizers serve as information spotters that help users to spot the final set of desired documents.

Text summaries can be either query-relevant summaries or generic summaries. A query-relevant summary presents the contents of the document that are closely related to the initial search query. Creating a query-relevant summary is essentially a process of retrieving the query relevant sentences/passages from the document, which has a strong analogy with the text retrieval process. Therefore, query-relevant summarization is often achieved by extending conventional IR technologies, and to date, a large number of text summarizers in the literature fall into this category. On the other hand, a generic summary provides an overall sense of the document's contents. A good generic summary should contain the main topics of the document while keeping redundancy to a minimum. As no query nor topic will be provided to the summarization process, it is challenging to develop a high quality generic summarization method, and is even more challenging to objectively evaluate the method.

In this paper, we propose two generic text summarization methods that create text summaries by ranking and extracting sentences from the original documents. The first method uses standard IR methods to measure sentence relevances, while the second method uses the latent semantic analysis technique to identify semantically important sentences, for summary creations. Both methods strive to select sentences that are highly ranked and different from each other. This is an attempt to create a summary with a wider coverage of the document's main content and less redundancy. Performance evaluations on the two summarization methods are conducted by comparing their summarization outputs with the manual summaries generated by three independent human evaluators. The evaluations also study the influence of different VSM weighting schemes on the text summariza-

SIGIR'01, September 9-12,2001,New Orleans, Louisiana, USA.

tion performances. Finally, the causes of the large disparities in the evaluators' manual summarization results are investigated, and discussions on human text summarization patterns are presented.

The remainder of the paper consists of four sections: Section 2 describes related research studies, Section 3 describes the two proposed text summarization methods, Section 4 presents the performance evaluations, and Section 5 summarizes the paper.

2. RELATED WORK

Text summarization has been actively researched in recent years. A majority of the research studies in the literature have been focused on creating query-relevant text summaries. M. Sanderson proposed a query-relevant summarizer that divides the document into equally sized overlapping passages, and uses the INQUERY text search engine to obtain the passage that best matches the user's query. This best passage is then used as a summary of the document [1]. A query expansion technique called Local Context Analysis (LCA, which is also from INQUERY) is used before the best passage retrieval. Given a topic and a document collection, the LCA retrieves top-ranked documents from the collection, examines the context surrounding the topic terms in each retrieved document, and then selects and adds the words/phrases that are frequent in this context to the query. B. Baldwin and T.S. Morton developed a summarizer that selects sentences from the document until all the phrases in the query are covered. A sentence in the document is considered to cover a phrase in the query if they co-refer to the same individual, organization, event, etc [2]. R. Barzilay and M. Elhadad developed a method that creates text summaries by finding lexical chains from the document [3]. The Cornell/Sabir system uses the document ranking and passage retrieval capabilities of the SMART text search engine to effectively identify relevant passages in a document [4]. The text summarizer from CGI/CMU uses a technique called Maximal Marginal Relevance (MMR) which measures the relevance of each sentence in the document to the user provided query, as well as to the sentences that have been selected and added into the summary [5]. The text summary is created by selecting the sentences that are highly relevant to the user's query, but are different from each other. The SUMMARIST text summarizer from the University of Southern California strives to create text summaries based on the equation: *summarization=topic identification+ interpretation+generation*. The identification stage filters the input document to determine the most important, central topics. The interpretation stage clusters words and abstracts them into some encompassing concepts. Finally, the generation stage generates summaries either by outputting some portions of the input, or by creating new sentences based on the interpretation of the document concepts [6]. However, this generation function was not realized in the paper. The Knowledge Management (KM) system from SRA International, Inc. extracts summarization features using morphological analysis, name tagging and co-reference resolution. They used a machine learning technique to determine the optimal combination of these features in combination with statistical information from the corpus to identify the best sentences to include in a summary [7].

3. CREATING GENERIC SUMMARIES

Query-relevant text summaries are useful for answering such questions as whether a given document is relevant to the user's query, and if relevant, which part(s) of the document is relevant. As query-relevant summaries are query biased, they do not provide an overall sense of the document content, and hence, are not appropriate for content overview. For answering such questions as which category the document belongs to, and what are the key points of the document, generic text summaries must be created and presented to the reader.

A document usually consists of several topics. Some topics are described intensively by many sentences, and hence form the major content of the document. Other topics may just be briefly mentioned to supplement the major topics, or to make the whole story more complete. A good generic summary should cover the major topics of the document as much as possible, and at the same time, keep redundancy to a minimum.

In this section, we propose two methods that create generic summaries by selecting sentences based on the relevance measure and the latent semantic analysis. Both methods need to first decompose the document into individual sentences, and to create a weighted term-frequency vector for each of the sentences. Let $T_i = [t_{1i}\ t_{2i}\ \ldots\ t_{ni}]^T$ be the term-frequency vector of passage i, where element t_{ji} denotes the frequency in which term j occurs in passage i. Here passage i could be a phrase, a sentence, a paragraph of the document, or could be the whole document itself. The weighted term-frequency vector $A_i = [a_{1i}\ a_{2i}\ \ldots\ a_{ni}]^T$ of passage i is defined as:

$$a_{ji} = L(t_{ji}) \cdot G(t_{ji}) \tag{1}$$

where $L(t_{ji})$ is the local weighting for term j in passage i, and $G(t_{ji})$ is the global weighting for term j in the whole document. When the weighted term-frequency vector A_i is created, we further have the choice of using A_i with its original form, or normalizing it by its length $|A_i|$. There are many possible weighting schemes. In Section 4.3, we inspect several major weighting schemes and disclose how these weighting schemes affect the summarization performances.

In the following subsections, the two text summarization methods are described in details.

3.1 Summarization by Relevance Measure

After the given document is decomposed into individual sentences, we compute the relevance score of each sentence with the whole document. We then select the sentence k that has the highest relevance score, and add it to the summary. Once the sentence k has been added to the summary, it is eliminated from the candidate sentence set, and all the terms contained in k are eliminated from the original document. For the remaining sentences, we repeat the steps of relevance measure, sentence selection, and term elimination until the number of selected sentences has reached the predefined value. The operation flow is as follows:

1. Decompose the document into individual sentences, and use these sentences to form the candidate sentence set S.

2. Create the weighted term-frequency vector A_i for each sentence $i \in S$, and the weighted term-frequency vector D for the whole document.

3. For each sentence $i \in S$, Compute the relevance score between A_i and D, which is the inner product between A_i and D.

4. Select sentence k that has the highest relevance score, and add it to the summary.

5. Delete k from S, and eliminate all the terms contained in k from the document. Recompute the weighted term-frequency vector D for the document.

6. If the number of sentences in the summary reaches the predefined value, terminate the operation; otherwise, go to Step 3.

In Step 4 of the above operations, sentence k that has the highest relevance score with the document is the one that best represents the major content of the document. Selecting sentences based on their relevance scores ensures that the summary covers the major topics of the document. On the other hand, eliminating all the terms contained in k from the document in Step 5 ensures that the subsequent sentence selection will pick the sentences with a minimum overlap with k. This leads to the creation of a summary that contains little redundancy.

3.2 Summarization By Latent Semantic Analysis

Inspired by the latent semantic indexing, we applied the singular value decomposition (SVD) to generic text summarization. The process starts with the creation of a terms by sentences matrix $\mathbf{A} = [A_1\ A_2\ \ldots\ A_n]$ with each column vector A_i representing the weighted term-frequency vector of sentence i in the document under consideration. If there are a total of m terms and n sentences in the document, then we will have an $m \times n$ matrix A for the document. Since every word does not normally appear in each sentence, the matrix A is usually sparse.

Given an $m \times n$ matrix $\mathbf{A}$, where without loss of generality $m \geq n$, the SVD of $\mathbf{A}$ is defined as [8]:

$$\mathbf{A} = \mathbf{U}\Sigma\mathbf{V}^T \tag{2}$$

where $\mathbf{U} = [u_{ij}]$ is an $m \times n$ column-orthonormal matrix whose columns are called left singular vectors; $\Sigma = \mathrm{diag}(\sigma_1, \sigma_2, \ldots, \sigma_n)$ is an $n \times n$ diagonal matrix whose diagonal elements are non-negative singular values sorted in descending order, and $\mathbf{V} = [v_{ij}]$ is an $n \times n$ orthonormal matrix whose columns are called right singular vectors. If rank($\mathbf{A}$)=r, then Σ satisfies

$$\sigma_1 \geq \sigma_2 \cdots \geq \sigma_r > \sigma_{r+1} = \cdots = \sigma_n = 0. \tag{3}$$

The interpretation of applying the SVD to the terms by sentences matrix $\mathbf{A}$ can be made from two different viewpoints. From transformation point of view, the SVD derives a mapping between the m-dimensional space spanned by the weighted term-frequency vectors and the r-dimensional singular vector space with all of its axes linearly-independent. This mapping projects each column vector i in matrix $\mathbf{A}$, which represents the weighted term-frequency vector of sentence i, to column vector $\psi_i = [v_{i1}\ v_{i2}\ \cdots\ v_{ir}]^T$ of matrix $\mathbf{V}^T$, and maps each row vector j in matrix $\mathbf{A}$, which tells the occurrence count of the term j in each of the documents, to row vector $\varphi_j = [u_{j1}\ u_{j2}\ \cdots\ u_{jr}]$ of matrix $\mathbf{U}$. Here each element v_{ix} of ψ_i, u_{jy} of φ_j is called the index with the x'th, y'th singular vectors, respectively.

From semantic point of view, the SVD derives the latent semantic structure from the document represented by matrix $\mathbf{A}$ [9]. This operation reflects a breakdown of the original document into r linearly-independent base vectors or concepts. Each term and sentence from the document is jointly indexed by these base vectors/concepts. A unique SVD feature which is lacking in conventional IR technologies is that the SVD is capable of capturing and modeling interrelationships among terms so that it can semantically cluster terms and sentences. Consider the words *doctor*, *physician*, *hospital*, *medicine*, and *nurse*. The words *doctor* and *physician* are synonyms, and *hospital*, *medicine*, *nurse* are the closely related concepts. The two synonyms *doctor* and *physician* generally appear in similar contexts that share many related words such as *hospital*, *medicine*, *nurse*, etc. Because of these similar patterns of word combinations, the words *doctor* and *physician* will be mapped near to each other in the r-dimensional singular vector space. Furthermore, as demonstrated in [10], if a word combination pattern is salient and recurring in a document, this pattern will be captured and represented by one of the singular vectors. The magnitude of the corresponding singular value indicates the importance degree of this pattern within the document. Any sentences containing this word combination pattern will be projected along this singular vector, and the sentence that best represents this pattern will have the largest index value with this vector. As each particular word combination pattern describes a certain topic/concept in the document, the facts described above naturally lead to the hypothesis that each singular vector represents a salient topic/concept of the document, and the magnitude of its corresponding singular value represents the degree of importance of the salient topic/concept.

Based on the above discussion, we propose the following SVD-based document summarization method.

1. Decompose the document D into individual sentences, and use these sentences to form the candidate sentence set S, and set $k = 1$.

2. Construct the terms by sentences matrix A for the document D.

3. Perform the SVD on A to obtain the singular value matrix Σ, and the right singular vector matrix $\mathbf{V}^T$. In the singular vector space, each sentence i is represented by the column vector $\psi_i = [v_{i1}\ v_{i2}\ \cdots\ v_{ir}]^T$ of $\mathbf{V}^T$.

4. Select the k'th right singular vector from matrix $\mathbf{V}^T$.

5. Select the sentence which has the largest index value with the k'th right singular vector, and include it in the summary.

6. If k reaches the predefined number, terminate the operation; otherwise, increment k by one, and go to Step 4.

Table 1: Particulars of the Evaluation Database

Document Attributes	Values
number of docs	549
number of docs with more than 10 sentences	243
avg sentences/doc	21
min sentences/doc	3
max sentences/doc	105

Table 2: Statistics of the Manual Summarization Results

Summarization Attributes	Values	Average Sentences/Doc
Total number of sentences	7053	29.0
Sentences selected by 1 person	1283	5.3
Sentences selected by 2 persons	604	2.5
Sentences selected by 3 persons	290	1.2
Total number of selected sentences	2177	9.0

In Step 5 of the above operation, finding the sentence that has the largest index value with the k'th right singular vector is equivalent to finding the column vector ψ_i whose k'th element v_{ik} is the largest. By the hypothesis, this operation is equivalent to finding the best sentence describing the salient concept/topic represented by the k'th singular vector. Since the singular vectors are sorted in descending order of their corresponding singular values, the k'th singular vector represents the k'th important concept/topic. Because all the singular vectors are independent of each other, the sentences selected by this method contain the minimum redundancy.

4. PERFORMANCE EVALUATION

In this section, we describe the data corpus constructed for performance evaluations, present various evaluation results, and make in-depth observations on some aspects of the evaluation outcomes.

4.1 Data Corpus

Our evaluations on the two proposed summarization methods have been conducted using a database of two months of the CNN Worldview news programs. Excluding commercial advertisements, a one day broadcast of the CNN Worldview program lasts for about 22 minutes, and consists of 15 individual news stories on average. The evaluation database consists of closed captions of 549 news stories whose lengths are in the range of 3 to 105 sentences. As summarizing short articles does not make much sense in real applications, for our evaluations we eliminated all the short stories with less than ten sentences, resulting in 243 documents. Table 1 provides the particulars of the evaluation database.

Three independent human evaluators were employed to conduct manual summarization on the 243 documents contained in the evaluation database. For each document, each evaluator was requested to select exactly five sentences which he/she deemed the most important for summarizing the story. Because of the disparities in the evaluators' sentence selections, each document can have between 5 to 15 sentences selected by at least one of the evaluators. Table 2 shows the statistics of the manual summarization results. As evidenced by the table, the disagreements among the three evaluators were much more than expected: each document has an average of 9.0 sentences selected by at least one evaluator, and among these 9.0 selected sentences, only 1.2 sentences receive a unanimous vote from all three evaluators. Even when the sentence selection is determined by a majority vote, we still get a lower than expected overlapping rate: an average of 2.5 sentences per document. The disparities became even larger with longer documents. These statistics suggest that for many documents in the database, their manual summarization determined by a majority vote could be very short (2.5 sentence per document), and this summary length is below the best fixed summary length (three to five sentences) suggested in [5]. For this reason, we decided to evaluate our two summarization methods using each of the three individual manual summarization results, as well as the combined result determined by a majority vote.

4.2 Performance Evaluations

We used the recall (R), precision (P), along with F to measure the performances of the two summarization methods. Let S_{man}, S_{sum} be the set of sentences selected by the human evaluator(s), and the summarizer, respectively. The standard definitions of R, P, and F are defined as follows:

$$R = \frac{|S_{man} \cap S_{sum}|}{|S_{man}|} \qquad P = \frac{|S_{man} \cap S_{sum}|}{|S_{sum}|} \qquad F = \frac{2RP}{R+P}$$

For our evaluations, we set the length of the machine generated text summaries to the length of the corresponding manual summaries. When the evaluation is performed using each individual manual summarization result, both $|S_{man}|$ and $|S_{sum}|$ are equal to five. When the evaluation is performed using the combined result determined by a majority vote, $|S_{man}|$ becomes variable, and $|S_{sum}|$ is set to the value of $|S_{man}|$.

The evaluation results are shown in Table 3. These results are generated using the BNN weighting scheme (see

Table 3: Evaluation Results

Test Data	First Summarizer			Second Summarizer		
	R	**P**	**F**	**R**	**P**	**F**
Assessor 1	0.57	0.60	0.58	0.60	0.62	0.61
Assessor 2	0.48	0.52	0.50	0.49	0.53	0.51
Assessor 3	0.55	0.68	0.61	0.55	0.68	0.61
Majority Vote	0.52	0.59	0.55	0.53	0.61	0.57

Section 4.3 for the detailed descriptions). As evidenced by the results, despite the very different approaches taken by the two summarizers, their performance measures are quite compatible. This fact suggests that the two approaches interpret each other. The first summarizer (the one using the relevance measure) takes the sentence that has the highest relevance score with the document as the most important sentence, while the second summarizer (the one based on the latent semantic analysis) identifies the most important sentence as the one that has the largest index value with the most important singular vector. On the other hand, the first summarizer eliminates redundancies by removing all the terms contained in the selected sentences from the original document, while the second summarizer suppresses redundancies by using the k'th singular vector for the k'th round of sentence selection. The first method is straightforward, and it is relatively easy for us to give it a semantic interpretation. As for the second method, there has been a long history of arguments about what essentially each of the singular vectors represents when a collection of text (which could be sentences, paragraphs, documents, etc) are projected into the singular vector space. Surprisingly, the two different methods bring to us very similar summarization outputs. This mutual resemblance enhances our belief that each important singular vector does capture a major topic/concept of a document, and two different singular vectors do capture two semantically independent topics/concepts that have the minimum overlap.

4.3 Weighting Schemes

In our performance evaluations, we studied the influence of different weighting schemes on the summarization performances as well. As shown by Eq.(1), given a term i, its weighting scheme is defined by two parts: local weighting $L(i)$ and global weighting $G(i)$. Local weighting $L(i)$ has the following four possible alternatives:

1. No weight: $L(i) = tf(i)$ where $tf(i)$ is the number of times term i occurs in the sentence.
2. Binary weight: $L(i) = 1$ if term i appears at least once in the sentence; otherwise, $L(i) = 0$.
3. Augmented weight: $L(i) = 0.5 + 0.5 \cdot (tf(i)/tf(max))$ where $tf(max)$ is the frequency of the most frequently occurring term in the sentence.
4. Logarithm weight: $L(i) = \log(1 + tf(i))$.

Possible global weighting $G(i)$ can be:

1. No weight: $G(i) = 1$ for any term i.
2. Inverse document frequency: $G(i) = \log(N/n(i))$ where N is the total number of sentences in the document, and $n(i)$ is the number of sentences that contain term i.

When the weighted term-frequency vector A_k of a sentence k is created using one of the above local and global weighting schemes, we further have the choice of

1. Normalization: which normalizes A_k by its length $|A_k|$.
2. No normalization: which uses A_k with its original form.

Therefore, for creating vector A_k of a sentence k, we have a total of $4 \times 2 \times 2 = 16$ combinations of the possible weighting schemes. In our experimental evaluations, we have studied nine common weighting schemes, and their performances are shown in Figure 1. As seen from the figure, summarizer 1 is less sensitive than summarizer 2 to the changes of weighting schemes. Any of the three local weighting schemes (i.e. Binary, Augmented, logarithm) produces quite similar performance readings. Adding a global weighing and/or the vector normalization deteriorates the performance of summarizer 1 by 2 to 3% in average. In contrast, summarizer 2 reaches the best performance with the binary local weighting, no global weighing and no normalization (denoted as BNN) for most of the cases, while its performance drops a bit by adding the global weighing, and deteriorates dramatically by adding the normalization into the formula.

4.4 Further Observations

Generic text summarization and its evaluation are very challenging. Because no query nor topic are provided to the summarization task, summarization outputs and performance judgments tend to lack consensus. In our experiments, we have seen the large degree of disparities in the sentence selections among the three independent evaluators, resulting in lower than expected scores (F=0.55 for summarizer 1, F=0.57 for summarizer 2) from the performance evaluation by a majority vote. The disparities became even larger with longer documents, and because of this, we adopted CNN worldview news reports, which have manageable text lengths, in our performance evaluations.

On the other hand, for query-relevant text summarization, the most common approach for performance evaluations, as showcased by the TIPSTER SUMMAC initiative [11], is that human evaluators use the automatically generated summary to judge the relevance of the original document to the user's query. These human evaluators' judgments are compared with some grand-truth judgments obtained beforehand, and the accuracy of the human evaluator's judgments are then used as the performance measures of the text summarizer. A document or a summary is judged relevant if at least one sentence within it is regarded as relevant to the query. As it is highly probable that a text summarizer can extract at least one query-relevant sentence from the original document, this simplistic evaluation method is likely to produce good performance scores.

It is observed from Table 3 that the two proposed summarizers both receive better scores when they are evaluated using the manual summarization results from evaluator 1 and 3. However, when evaluated using evaluator 2's results,

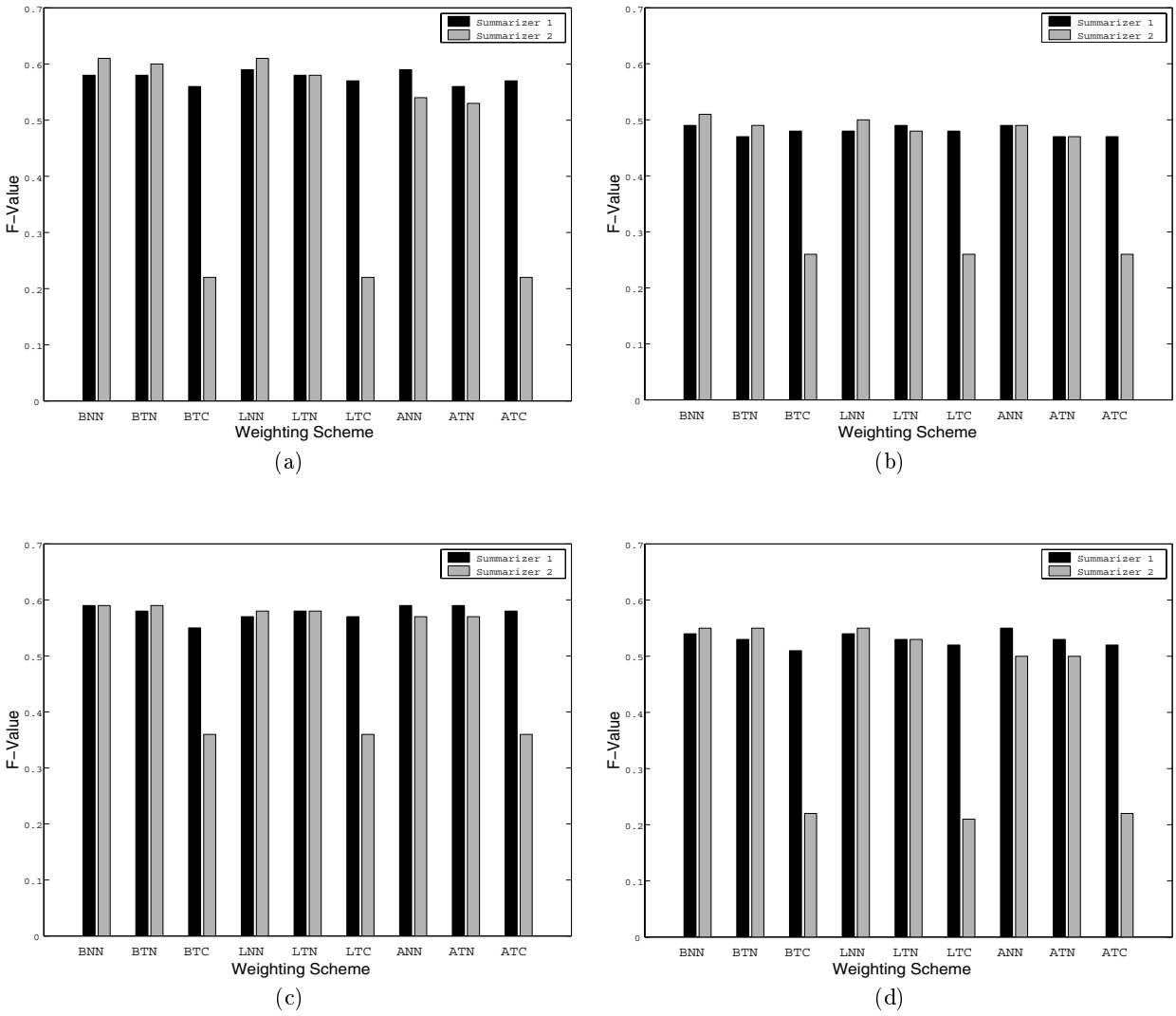

Figure 1: The influence of different weighting schemes on the summarization performances. (a),(b),(c),(d): Evaluation using the manual summarization result from evaluator 1, 2, 3, and the one determined by a majority vote, respectively. The notation of weighting schemes is the same as the one from the SMART system. Each weighting scheme is denoted by three letters. The first, second, and the third letters represent the local weighting, the global weighing, and the vector normalization, respectively. The meaning of the letters are as follows: N: No weight, B: Binary, L: Logarithm, A: Augmented, T:Inverse document frequency, C: Vector normalization.

the performance scores drop by 10% in average, dramatically dragging down the performance scores for the evaluation by a majority vote. An in-depth analysis of the cause of this large difference has revealed different manual summarization patterns among the three evaluators. Consider the following passage taken from a CNN news story reporting the recent Israeli-Palestinian conflicts, political efforts for restoring the calm in the region, and hostile sentiments among Palestinian people:

> (1) .....IN RECENT VIOLENCE MORE THAN 90 PEOPLE HAVE BEEN KILLED, THOUSANDS MORE INJURED, THE OVERWHELMING MAJORITY OF THOSE ARE PALESTINIANS.
> .....
> (2) NOW AFTER A BRIEF LULL IN THE VIOLENCE , NEW FIGHTING, NEW CLASHES ERUPTED THURSDAY, AND TONIGHT MORE GUNFIRE REPORTED WITH MORE INJURIES OF PALESTINIAN.
> .....
> (3) IN THE NORTHERN WEST BANK TOWN NABLUS , ISRAELI TANKS EXCHANGED FIRE WITH PALESTINIAN GUNMEN, KILLED AT LEAST 3 OF THEM ON WEDNESDAY.

The above three sentences all cover the topic of Israeli-Palestinian conflicts. Our two summarizers both selected sentence (1), and discarded (2) and (3) because of their similarities to sentence (1). On the other hand, both evaluator 1 and 3 selected sentence (1), while evaluator 2 picked all the three sentences for summarizing the topic. This example represents a typical pattern that happens repeatedly in the whole evaluation process. The fact suggested by this phenomenon is that, to summarize a document, some people strive to select sentences that maximize the coverage of the document's main content, while others tend to first determine the most important topic of the document, and then collect only the sentences that are relevant to this topic. Evidently, when it comes to the evaluation of our two proposed summarization methods, the former type of evaluators generates a higher accuracy score than the latter.

5. SUMMARIES

This paper presented two text summarization methods that create generic text summaries by ranking and extracting sentences from the original documents. The first method uses standard IR methods to rank sentence relevances, while the second method uses the latent semantic analysis technique to identify semantically important sentences, for summary creations. Both methods strive to select sentences that are highly ranked and different from each other. This is an attempt to create a summary with a wider coverage of the document's content and a less redundancy. For experimental evaluations, a database consisting of two months of the CNN Worldview news programs was constructed, and performances of the two summarization methods were evaluated by comparing the machine generated summaries with the manual summaries created by three independent human evaluators. Despite the very different approaches taken by the two summarizers, they both produced quite compatible performance scores. This fact suggests that the two approaches interpret each other. The evaluations also included the study of the influence of different VSM weighting schemes on the text summarization performances. Finally, the causes of the large disparities in the evaluators' manual summarization results were investigated, and discussions on human text summarization patterns were provided.

In future work, we plan to investigate machine learning techniques to incorporate additional features for the improvement of generic text summarization quality. The additional features we are currently considering include linguistic features such as discourse structure, anaphoric chains, etc, semantic features such as name entities, time, location information, etc. As part of the large-scale video content summarization project, we also plan to investigate how image and audio acoustic features extracted from video programs can help to improve the text summarization quality, and vice versa.

6. REFERENCES

[1] M. Sanderson, "Accurate user directed summarization from existing tools," in *Proceedings of the 7'th International Conference on Information and Knowledge Management (CIKM98)*, 1998.

[2] B. Baldwin and T. Morton, "Dynamic coreference-based summarization," in *Proceedings of the Third Conference on Empirical Methods in Natural Language Processing (EMNLP3)*, (Granada, Spain), June 1998.

[3] R. Barzilay and M. Elhadad, "Using lexical chains for text summarization," in *Proceedings of the Workshop on Intelligent Scalable Text Summarization*, (Madrid, Spain), Aug. 1997.

[4] C. Buckley and et al., "The smart/empire tipster ir system," in *Proceedings of TIPSTER Phase III Workshop*, 1999.

[5] J. Goldstain, M. kantrowitz, V. Mittal, and J. Carbonell, "Summarizing text documents: Sentence selection and evaluation metrics," in *Proceedings of ACM SIGIR'99*, (Berkeley, CA), Aug. 1999.

[6] E. Hovy and C. Lin, "Automated text summarization in summarist," in *Proceedings of the TIPSTER Workshop*, (Baltimore, MD), 1998.

[7] *http://www.SRA.com.*

[8] W. Press and et al., *Numerical Recipes in C: The Art of Scientific Computing*. Cambridge, England: Cambridge University Press, 2 ed., 1992.

[9] S. Deerwester, S. Dumais, G. Furnas, T. Landauer, and R. Harshman, "Indexing by latent semantic analysis," *Journal of the American Society for Information Science*, vol. 41, pp. 391–407, 1990.

[10] M. Berry, S. Dumais, and G. O'Brien, "Using linear algebra for intelligent information retrieval," Tech. Rep. UT-CS-94-270, University of Tennessess, Computer Science Department, Dec. 1994.

[11] T. Firmin and B. Sundheim, "Tipster/summac summarization analysis participant results," in *TIPSTER Text Phase III Workshop*, 1998.