International Journal of Computational Intelligence and Applications

A PROBABILISTIC APPROACH TO MULTI-DOCUMENT SUMMARIZATION FOR GERERATING A TILED SUMMARY

M. SARAVANAN*
Research Scholar
Department of Computer Science & Engineering Department, IIT Madras
Chennai, Tamil Nadu 600 024, India.

Prof. S. Raman
Department of Computer Science & Engineering Department, IIT Madras
Chennai, Tamil Nadu 600 024, India.

Dr. B. Ravindran
Department of Computer Science & Engineering Department, IIT Madras
Chennai, Tamil Nadu 600 024, India.

Received (25th May 2006)
Revised (June 8, 2006)

Data availability is not a major issue at present times in view of the widespread use of Internet; however, information and knowledge availability are the issues. Due to data overload and time-critical nature of information need, automatic summarization of documents plays a significant role in information retrieval and text data mining. This paper discusses the design of a multi-document summarizer that uses Katz's K-mixture model for term distribution. The model helps in ranking the sentences by a modified term weight assignment. Highly ranked sentences are selected for the final summary. The sentences that are repetitive in nature are eliminated, and a tiled summary is produced. Our method avoids redundancy and produces a readable (even browsable) summary, which we refer to as an event-specific tiled summary. The system has been evaluated against the frequently occurring sentences in the summaries generated by a set of human subjects. Our system outperforms other auto-summarizers at different extraction levels of summarization with respect to the ideal summary, and is close to the ideal summary at 40% extraction level.

Keywords: Text Summarization; Probabilistic model; Tiled summary.

1. Introduction

Automatic text summarization has an important role in information industry, especially for the Internet community, due to the exponential growth of on-line documents in recent years. It is presumed that most part of the web is occupied by

*Working in Department of Information Technology, Jerusalem College of Engineering, Chennai-601 302, Tamil Nadu, India.

text-related data. The availability of fast search-engines enables the retrieval of colossal amount of information in the form of related documents quickly. However, the user community is benefited only if the retrieved documents are condensed and presented in the form of a readable or a browsable summary. Due to the increased diversity of document collections, the use of automatic summarization is guaranteed to remain as one of the important topics of research in statistical natural language processing and text data mining. In this paper, we describe the design of a multi-document summarizer and the use of an intrinsic method to evaluate the summaries.

In case the user does not use meaningful keywords in a query, an irrelevant summary gets generated, because lexical matching techniques are inaccurate [1]. Due to the ambiguities in natural language text, like synonymy, by which multiple words have the same meaning, and polysemy, by which the same word refers to different concepts, there will be many irrelevant matching of terms in the collection. The summarization methods discussed in the literature are mainly query-based [2][3][4]. Another study [5] presents an approach to query-based summaries in information retrieval that helps to customize summaries in a way, which reflect the information need expressed in a query. In order to decide the aspects of the documents which provide utility to the generation of a summary, title, headings, leading paragraph, and their overall structural organization were studied. Moreover, it can be a repetition of Edmundson's [6] work of abstract generation system, specifically for text summarization system. In this work, we consider automatic summarization that is not query-based.

Extraction of sentences in the generation of a summary at different percentage levels of text is one of the widely used methods in document summarization [2][3]. In addition to the extraction of sentences from documents, some algorithms automatically construct phrases that are added to the generated summary, to make it more intelligible [7]. One problem in this practice is that automatic construction of phrases is a difficult task, and wrongly included phrases will totally degrade the quality of the summary.

To circumvent the above problems associated with summarizers, we pursue a statistical approach that predicts summary-worthy sentences from the input documents. Statistics-based systems are empirical, retrainable systems that minimize human efforts [8]. The proportion of terms that are identified for summarization is closely related to the semantic content of the documents. Hence, we attempt to design a probabilistic model, which is a modified version of a term weighting scheme that improves the performance level of the summarizer. The pre-processed terms in the documents, represented by the vector-space model, are further processed by the term distribution model (K-mixture model) that identifies the hidden term patterns, and finally produces the ranked sentences. The block diagram of the system is shown in Fig 1.

Now we will discuss some related tools that play a support role in the summarization task. In the multi-document summarization procedure, there is a possibility

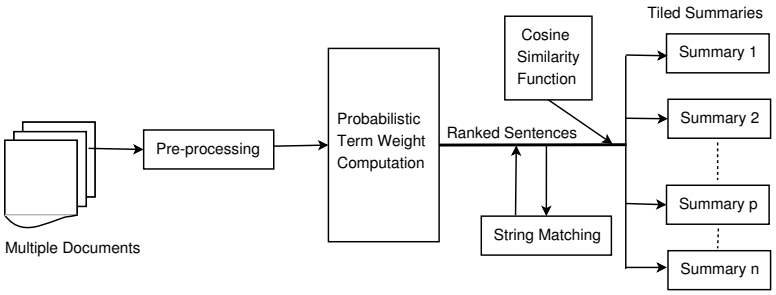


Fig. 1. System for summarization of multiple documents.

for redundancy during the extraction of sentences from documents belonging to different domains. We eliminate the sentences that are repetitive in nature by the application of string-matching algorithms. For presenting the summary in the form of a short caption for easy readability or browsability, a cosine similarity function is applied to the highly ranking sentences. This step produces a tiled summary. The layout of tiled summary is meant to reflect the pattern of subtopics contained in an expository text.

We also evaluate the usefulness of the proposed term distribution model for multi-document summarization. For evaluation, we have followed the intrinsic methods which are concerned with the quality of summary, produced by considering two different techniques. First, we compare the sentences generated by our method with the summaries generated by 10 humans (judges). The judges are from different levels of accomplishments in particular to the domain. In the second part, we compare our results with summary generated by 2 other auto-summarizers available on the web. To make evaluation more efficient, we have considered two different types of auto-summarizers for comparison. That is one is based on semantic analysis of the given text, and the other one is based on the statistical approaches. The results of the system-generated summary have been analyzed at different percentage levels like 15%, 20%, 25%, 30%, 40% ,50% and 60% with human generated summary, and are also compared with the results of the other auto-summarizers. The percentage levels mentioned above refers to the extraction levels of summary from the original documents. The degree of closeness of the system-generated summary to a human generated summary is taken as one of the measures of quality, and recall is another standard measure used for the evaluation. Recall measures show the proportion of relevant sentences that are retrieved. By evaluating the summary at different percentage levels of summarization, we have arrived at a threshold value at which the quality reached the maximum level.

This paper begins with the discussion of different approaches to text summarization and their drawbacks in Section 2. Section 3 introduces a term distribution model to illustrate how sentences are extracted from the document space. Section 4 discusses the evaluation part of our work. In Section 5, a string-matching algorithm

that helps to avoid the redundant sentences in the summary is discussed. The process of generating a tiled summary is also discussed in the same section. The results and discussion is covered in Section 6. The conclusions and suggestions for future work are listed in Section 7.

2. Related Work

Luhn points out that frequency data can be used to extract words and sentences for representing a document [9]. The relative frequency approach [6] used in automatic abstraction and indexing can also be extended to other information retrieval tasks. Salton & Buckley[10] has suggested a technique for automatic text retrieval based on term weighting approach. The different approaches used for text summarization on task-based evaluation have been compared and discussed in detail in [4]. None of these methods deal with the characterization of terms. In recent times, a query-based summarization approach [4] has emerged. It does not seem to make much headway because of its inability to get an exact string match between query words and the words in the document space. As an alternative to the term-weighting approach, a semantic analysis-based summarization [2] has also been considered. This produces summaries only for domain-specific documents, while rejecting the other documents. For multi-document summarization dealing with different types of subjects, an approach that is independent of domain knowledge is desirable.

The above-mentioned approaches are not based on the application of probabilistic models in information retrieval subsystems. One reason for this is that the earlier work in information retrieval is non-probabilistic. After 30 years later, some significant headway has been made with probabilistic methods. Automatic indexing based on a probabilistic model is discussed in Harters work [11], which uses the distribution of word tokens within a document. The MEAD summarizer uses the technique of centroid-based summarization [12], can be used for single and multiple documents.

The successes of information extraction research [12][13][14] have had a significant impact on the approaches to multi-document summarization task. The current literature on multi-document summarization does not place much emphasis on term distribution. In our work, we rely heavily on the assumption that the distribution of index terms throughout the collection, or within some subset of it, will indicate the likely relevance of the sentence in the given document collection. The TextTiling method proposed by Martin Hearst [15] is based on the algorithm of partitioning expository texts into multi-paragraph segments that reflect their subtopic structure. In our work, we have used a cosine similarity function [10] to arrive at a tiled summary from the ranked sentences that have been output from the term distribution model. In the next section, we describe the use of term distribution model for automatic extraction of sentences in the context of retrieval of the ranked sentences from the document space.

3. Probabilistic Models

Probabilistic models of term distribution in documents are getting renewed attention in both statistical NLP and in intelligent information retrieval [7]. In this section, we discuss the application of two theoretical models and the usage of K-mixture model for the projection of term patterns in the collection. The TF-IDF (Term Frequency Inverse Document Frequency) [13] used in most of the summarization tasks is an ad-hoc weighting scheme because it is not directly derived from a mathematical model of term distribution or relevancy. But it can be used in situations in which a rough measure of similarity between vectors of counts is needed. Term distribution models are used for deriving a probabilistically motivated term weighting scheme for the vector space model. It captures the regularities of word occurrence in subunits of a document collection.

Our mathematical model of term distribution makes use of two theoretical models, namely, the Poisson and the Negative Binomial distributions. Johnson and Kotz [16] have concluded in their work: "*The negative binomial is frequently used as a substitute for the Poisson when it is doubtful whether the strict requirements of the Poisson, particularly independence, will be satisfied*". Our experiments with negative binomial and Poisson distribution confirmed the findings of the earlier work, that Negative binomial is capable of providing good fit for distribution of content words as well. The computation of negative binomial involves large binomial coefficients and it is difficult to work with, in practice. Hence, we selected the computationally simpler version of negative binomial, known as K-mixture model [17] for our work. We found that K-mixture model yields a better approximation of the data compared to the Poisson model. It is described as the mixture of an infinite number of Poisson distributions [18] and its terms can be arrived at by varying the Poisson parameters between different set of observations. The formula used in the calculation of probability of the word w_i that appears k times in a document by using the K-mixture is given as:

$$P_i(k) = (1 - a_f)\delta_{k,0} + \frac{a_f}{e_t + 1} \quad \frac{(e_t)^k}{(e_t + 1)^k} \tag{1}$$

where $\delta_{k,0} = 1$ if and only if $k = 0$; and $\delta_{k,0} = 0$, otherwise, and af and et are parameters that can be fit using the observed mean (t) and observed inverse document frequency (IDF) as follows.

$$t = cf_i/N; IDF = log_2(N/df_i); e_t = t * 2^{IDF} - 1 = (cf_i - df_i)/df_i; a_f = t/e_t \tag{2}$$

Collection frequency (cf_i) refers to the total number of occurrences of terms in the collection, document frequency (df_i) refers to the number of documents in the collection in which the term occurs, and N is the number of documents on the collection. The parameter a_f used in the formula refers to the absolute frequency of the term, and e_t can be used to calculate the "extra terms" per document in which the term occurs (compared to the case where a term has only one occurrence per document). The most frequently occurring words in all selected documents are

removed by using the measure of IDF that is used to normalize the occurrence of words in the document. In addition to retrieval, a good understanding of distribution patterns is useful whenever we want to assess the likelihood of a certain number of occurrences of a specific word in a unit of text. This will be helpful in identifying key sentences based on the weights.

We understand that the content words are very important for identifying the concepts (topics) in the documents. Non-content words occur frequently in a document along with content words, but make a significant impact in the distribution of terms in the document space. In the K-mixture model, each occurrence of a content word in the text decreases the probability of finding an additional term, but the decrease becomes consecutively smaller. Also, the large number of occurrences of content words indicates the central concept of the document. After substituting different calculated weights to K-mixture formula, the following procedure is adopted to extract the most relevant sentences from the document space [19].

(1) Normalize the terms by using the term characterization based on the parameter e_t.
(2) Calculate the sentence weight by summing up the term probability values.
(3) Rank the sentences based on the sentence weights by controlling the variables identified during the pre-processing of documents.
(4) Output the sentences in decreasing order of rank.

In this extraction process, each sentence is assigned a score that represents the degree of appropriateness for inclusion in the summary. The results of our work are discussed in sections 4 and 6.

4. Text Summarization: Evaluation

Evaluation of summarization methods can generally be discussed in two ways [19]. The evaluation measure based on information retrieval task is termed as the *extrinsic* method, while the evaluation based on user judgements of system-generated summaries is called the *intrinsic* measure. We chose the latter, since we concentrated more on the quality of the summary.

Table 1. Recall measure at each extraction level of summary for different sets of documents.

Document Set	15% (1)	20% (2)	25% (3)	30% (4)	40% (5)	50% (6)	60% (7)
1	0.33	0.47	0.53	0.60	0.73	0.80	0.93
2	0.18	0.36	0.55	0.64	0.73	0.74	1.00
3	0.42	0.58	0.67	0.67	0.75	0.83	0.92
4	0.25	0.50	0.50	0.75	0.75	1.00	1.00
5	0.22	0.33	0.47	0.53	0.73	0.80	0.93
6	0.44	0.56	0.56	0.67	0.79	0.89	1.00

We used the newswire text, available on www.hinduonnet.com, www.rediff.com, www.indiatimes.com and www.cricinfo.com. We did not examine the documents based on the number of sentences or by any other domain-specific subjects. The corpus taken for this study is consisted of a total of 29 documents grouped into 6 sets with approximately 1400 sentences extracted from different newsgroups web sites which is available in [21]. The documents are grouped based on conceptual proximity along with the indexing of temporal dimensions related to concepts.

In order to evaluate the effectiveness of the application of term distribution model in summary extraction, sentences extracted by our model, semantic analysis based summarizer '*Sinope'* (System A) [22] and the summarizer based on the statistical model with linguistic approaches '*Copernic*' (System B) [23] are compared with the human-generated summaries. In the first part of the evaluation process, we compared our results with the frequently occurring sentences in the human-generated summary at different extraction levels of summarization.

The ideal summary comprises of the intersection of frequently occurring sentences in the human-generated summaries. As we already mentioned about recall in the introduction, it has been redefined as the ratio of the number of sentences that are common to both the system-generated summary and the ideal summary to the number of sentences in the ideal summary. Recall measure calculated at different extracted summary levels are shown in Table 1, and the average recall is displayed in Fig 2.

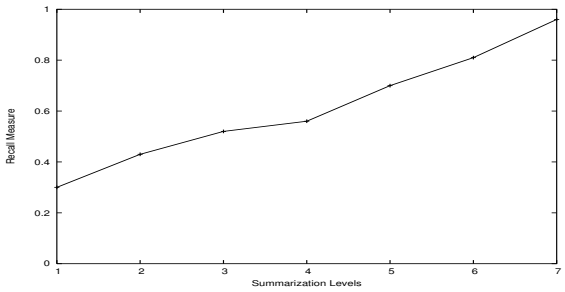


Fig. 2. Average recall measures at 7 different extraction levels.

Normally, the most relevant sentences should be retrieved in the summarization process, while the non-relevant ones must be rejected, to have a high recall value. From Table 1, it is clear that the system-generated summary contains almost all the sentences present in the ideal summary at 60% extraction level of summary. At 40% system extraction level, it contains close to 75% of the sentences in the ideal summary. The test results are based on the corpus consisting of 6 sets of documents with approximately 1400 sentences extracted from different newsgroup web sites.

5. Applying String-matching algorithm and Tilizing a summary

We now briefly explain how the string-matching algorithm has been used to remove the redundancy in the generated summary. Subsequently, we outline the process of tilization, which produces a captioned summary.

5.1. *String-matching algorithm*

To assemble the extracted sentences into a coherent summary by removing the redundant sentences, we propose a string-matching algorithm. Even though string matching is used as a basic component for most of the preprocessing tasks, the algorithm could be extended to analyze the sentences resulting from the term distribution model. Here, the intelligible stems in the sentences are compared with stems present in other sentences of the summary. To get the intelligible stems, we used a root word stemmer [24] that avoided the inflectional forms of words as well. The algorithm is explained in Fig 3.

1. The ranked sentences from the term distribution model considered as an input to this algorithm.
2. Use the stems from the preprocessing stage of selected sentences for matching.
3. If 75% of the stems in the two sentences compared are equal, then remove one of the sentences and supplement the list with the highest ranked next sentence.
4. Continue Steps 2 and 3 until all sentences are processed.
5. Submit the resulting list to the TextTiling procedure.

Fig. 3. Algorithm for String matching.

Usage of the string-matching algorithm during post processing stage guarantees the maintenance of quality and coherence in the final summary. The final summary is not structured into paragraphs and sections. Since a structured summary is desirable for readability, we followed the TextTiling algorithm proposed by Hearst [15] to produce event-based tiled summary.

5.2. *Tilizing a summary*

TextTiling is a method for partitioning full-length text documents into coherent multi-paragraph units, with each unit reflecting the subtopic structure of the text. Hearst designed an algorithm to partition expository texts into multi-paragraph segments that reflect their subtopic structure. The algorithm detects subtopic boundaries by analyzing the term repetition patterns within the text.

Hearst [15] observed that subtopics are often expressed by the interaction of multiple simultaneous themes while describing two fully implemented algorithms that use only term repetition information to determine the extents of the subtopics. The layout of the tiles is meant to reflect the pattern of subtopics contained in an expository text. In our work, the idea of the tilization is implemented based on cosine similarity [10][15] between highly ranked sentences, computed using the sentence weights obtained from the term distribution model. This is not purely a distance measure. It is developed from the cosine of the angle between the weight vectors representing the ranked sentences. The equation for the cosine similarity measure [9] is

$$F(X,Y) = \frac{\sum(x_i * y_i)}{\sqrt{\sum(x_i)^2} * \sqrt{\sum(y_i)^2}} \tag{3}$$

where i ranges over all the sentences in the document collection and x_i and y_i refer to the sentence weights in the document space. The angular measure of the cosine similarity $F(X,\ Y)$ is ranging from 1 for the highest to 0 for the lowest. We check how well the two sentences taken for comparison are correlated with each other by the above measure. Based on this similarity computation between sentences, tiles are exhibited in the summary. This sentence extraction-based summarization approach has been enhanced with tiled summary for better readability or browsability. The summary resembles the captioned first page of newsgroup web sites. The result is presented as a final summary containing distinct event-based tiled paragraphs. Formally, the final summary (FS) can be represented as:

$$FS = \sum_{j=1}^{no.ofevents} t_j \tag{4}$$

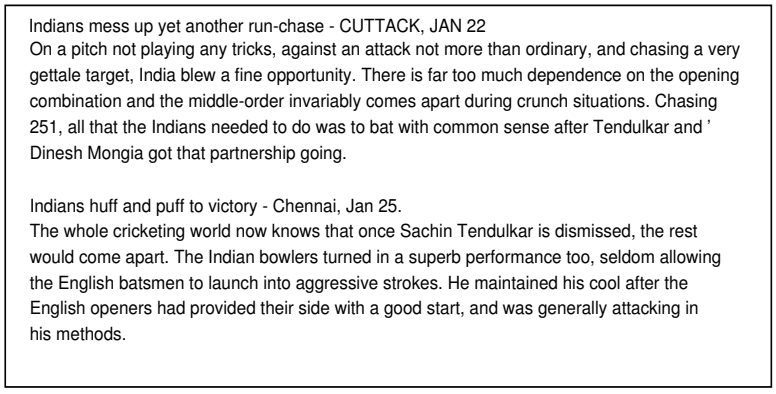
where each t_j represents the tiled paragraphs present in the final summary and $\sum$ represents the concatenation of paragraphs. The tilization can be seen in the sample summary of an input documents generated by our system in Fig 4, along with the summary produced by semantic analysis-based summarizer in Fig 5. The tiled summary exhibits the readability in Fig 4 compare to the full-text summary shown in Fig 5.

We observed only small variations in the human-generated summaries, as verified by the statistical test discussed in the next section. Comparison of the system summary with auto-summarizers on the basis of its closeness to the ideal summary is given in the subsequent section.

6. Results and Discussion

6.1. *Statistical Significance*

In the process of generating ideal summary, we asked human subjects to summarize our document sets approximately at 20% extraction level, since the degree of


Indians mess up yet another run-chase - CUTTACK, JAN 22
On a pitch not playing any tricks, against an attack not more than ordinary, and chasing a very gettale target, India blew a fine opportunity. There is far too much dependence on the opening combination and the middle-order invariably comes apart during crunch situations. Chasing 251, all that the Indians needed to do was to bat with common sense after Tendulkar and ' Dinesh Mongia got that partnership going.

Indians huff and puff to victory - Chennai, Jan 25.
The whole cricketing world now knows that once Sachin Tendulkar is dismissed, the rest would come apart. The Indian bowlers turned in a superb performance too, seldom allowing the English batsmen to launch into aggressive strokes. He maintained his cool after the English openers had provided their side with a good start, and was generally attacking in his methods.

Fig. 4. Example of 20% tiled summary produced by our system.

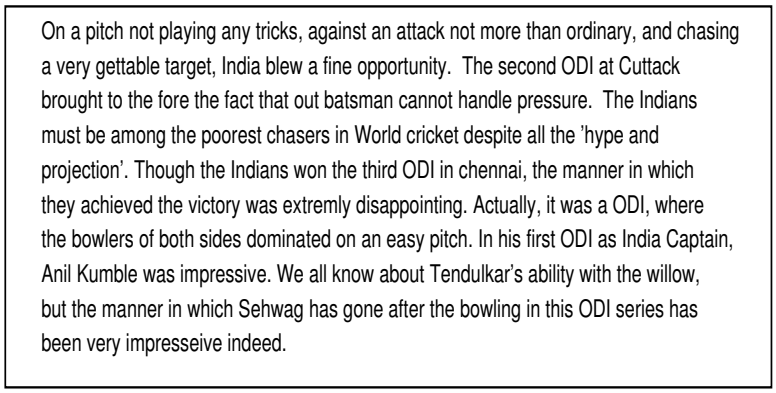

On a pitch not playing any tricks, against an attack not more than ordinary, and chasing a very gettable target, India blew a fine opportunity. The second ODI at Cuttack brought to the fore the fact that out batsman cannot handle pressure. The Indians must be among the poorest chasers in World cricket despite all the 'hype and projection'. Though the Indians won the third ODI in chennai, the manner in which they achieved the victory was extremly disappointing. Actually, it was a ODI, where the bowlers of both sides dominated on an easy pitch. In his first ODI as India Captain, Anil Kumble was impressive. We all know about Tendulkar's ability with the willow, but the manner in which Sehwag has gone after the bowling in this ODI series has been very impresseive indeed.

Fig. 5. Example of 20% tiled summary produced by System A.

agreement among human subjects tends to decrease as the length of summary increases [20]. The goal of the experiment was to understand the agreement among the human subjects and how it influenced the evaluation results. We performed the *Cochran's Q test* to evaluate the null hypothesis that the common sentences presented in the summaries generated by human subjects were randomly distributed.

We applied Cochran's test on the data from the subjects for analyzing the frequency of sentences present in the human-generated summary on the same lines as in [4][20]. Cochran Q test [20] for k related samples provide a method for testing whether three or more matched sets of frequencies or proportions differ significantly among themselves. *Cochran statistic Q* approximates the chi-square distribution with $j-1$ degrees of freedom, where j refers to the number of sentences in a document. We compared the sentences in all the six document sets having j matched sentences in each set generated by human subjects. Our results show that there is

significant agreement among the human-generated summaries. Hence the null hypothesis is rejected at 95% confidence level in all document sets. In other words, the probability that human subjects extract the same sentences is much higher than would be expected by chance.

We are interested in how these factors can affect the evaluation results. Summaries from human and automatic summarizers were compared at different extraction levels of summarization and the results are discussed in the next section.

6.2. *Comparison between generic and ideal summary*

To arrive at the threshold level of summarization, the ideal summary is compared with different extraction levels of system-generated summaries. We found that at 40% extraction level, the system summary contains close to 75% of the sentences in the ideal summary. We also observe a steady increase in the recall level after the level of extraction of sentences augmented from 20% to 40% . We further increased the extraction level by 20% and found that the generated summary contains almost all the sentences present in the ideal summary. In the next stage of evaluation, we compared the recall measure of our summarizer with that of publicly available summarizers (System A) and (System B). Recall measures show the relevancy level of the summary. Our system clearly outperformed the two auto-summarizers (System A and System B) on all document sets, where System A performs semantic analysis and System B uses statistical and linguistics algorithms to generate a summary. The results are shown in the form of bar diagram in Fig 6.

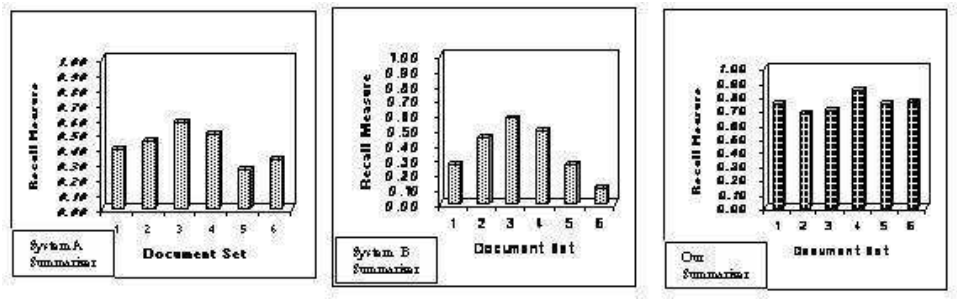


Fig. 6. Recall at 40% extration level of summarization by System A, System B and our system for six different document sets.

6.3. *Discussion*

The results of our summarizer agree with the hypothesis that there is a possibility of emergence of relevant sentences in probabilistically motivated summarization. The human judges produced summaries which had variation of 20 to 25%. The

intersection of these summaries was considered as the ideal summary. The system-generated summary was compared with the ideal summary at different percentage levels of summarization. We observed that there was 75% of sentences contained in the ideal summary are present in our system summary at 40% extraction level. At this level, the system outperforms the auto-summarizers as well. At 60% extraction level, it contains almost all the sentences present in the ideal summary. The result of our summarizer shown in Fig 6 also illustrates the uniform recall in our system irrespective of the size of the document sets.

7. Conclusion and Future work

The goal of Information retrieval research is to develop models and algorithms for retrieving information from document repositories, particularly text data. The suggested term distribution model in the summarization task increases the usefulness of statistical NLP in information retrieval applications. We have discussed the issues that must be considered during the non-query based summarization. The results of our system based on term distribution model show that the generated summary at 40% level is close to the ideal summary. The event-specific tiled summary without redundancy is an additional feature of our summarizer. In general, statistical systems are empirical, retrainable systems that minimize human efforts.

We plan to extend the testing of our system using different term distribution models over larger text collections. The application of different term models can eventually lead to better measures of content similarity. The most important aspect of research is to extend the functionality of the tool so that it can perform cross language information retrieval and multilingual, multi-document summarization. Except stemming, other NLP preprocessing tools used in our research could be extended to other natural language texts. More precisely, we can perform other text mining tasks like text categorization and automatic indexing from the generated summary instead of from the original documents so as to improve the speed and performance of such systems.

References

1. M. Berry, S. Dumais, T. Landauer, and G. O'Brian, Using linear algebra for intelligent information retrieval, *SIAM Review.* **37** (1995) 573-595.
2. R. Barzilay and M. Elhadad, Using Lexical chains for text summarization, *in Proc. of the ACL Workshop on Intelligent Scalable Text Summarization.* (1997) Madrid, Spain, 10-17.
3. Julian M. Kupiec, J. Pedersen and F. Chen, A Trainable Document Summarizer, *in Proc. of* 18^{th} *ACM-SIGIR conference on Research and Development in Information Retrieval.* July 1995, Seattle, USA, 68-73.
4. M. Hajime and O. Manabu, A comparison of summarization methods based on taks-based evaluation, 2^{nd} *International Conference on language resources and evaluation, LREC-2000.* (2000) Athens, Greece, 633-639.

5. A. Tombros and M. Sanderson, Advantages of query biased summaries in Information Retrieval, *in Proc. of SIGIR-98.* (1998) Melbourne, Australia, 2-10..
6. H. P. Edmundson, New methods in automatic abstracting, *Journal of the ACM.* **16** No.2 (1969) 264-285.
7. C. D. Paice, Constructing Literature abstracts by computer: Techniques and Prospects, *Information Processing & Management.* **26** No.1 (1990) 264-285.
8. C. D. Manning and H. Schutze, *Foundations of Statistical Natural Language Processing* (The MIT Press, London, England, 2001).
9. H.P. Luhn, The automatic creation of literature abstracts, *IBM Journal of Research and Development.***2** No.2 (1958) 159-165.
10. G. Salton and C. Buckley, Term weighting approaches in automatic text retrieval, *Information Processing & Management.* **24** (1988) 513-523.
11. S. Harter, A Probabilistic Approach to automatic keyword indexing, PhD Dissertation, University of Chicago, (1975).
12. D. R. Radev, H. Jing and M. Budzikowska, Centroid-based Summarization of multiple document: Sentence extraction, Utility-based evaluation, and user studies, *ANLP-NAACL workshop on summarization.* (2000) Washington, USA, 21-29.
13. Neto and Santos, Document Clustering and Text Summarization, *in Proc. of* 4^{th} *International Conference on Practical applicationsof knowledge discovery and data mining, PADD-2000.* (2000), London, 41-55.
14. C. Y. Lin and E. H. Hovy, The Automated Acquisition of Topic Signatures for Text Summarization, *in Proc. of the COLING Conference.* (2000) Strasbourg, France, 595-601.
15. M. A. Hearst and C. Plaunt, Subtopic structure for Full-length Document Access, *in Proc. of the Sixteenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval.* (1993) 59-68.
16. N. L. Johnson and S. Katz, *Discrete Distributions* (The Houghton Mifflin Press, Massachusetts, 1969).
17. S. M. Katz, Distribution of content words and phrases in text and language modeling, *Natural Language Engineering.* **2** No.1 (1995) 15-59.
18. K. W. Church and W. Gale, Poisson Mixtures, *Natural Language Engineering.* **1** No.2 (1995) 163-190.
19. M. Saravanan and S. Raman, The term distribution model for summarization of multiple docuements, *in Proc. of Indo-European Conference on Multilingual Communication Technologies (IEMCT)* (2002) CDAC, Pune, 182-192.
20. H. Jing, R. Barzilay, K. Mckeown and M. Elhadad, Summarization Evaluation Methods:Experiments and Analysis, *in Proc. of AAAI98 Spring Symposium on Intelligent Text Sumamrization*, (1998) 60-68.
21. http://iil.cs.iitm.ernet.in
22. http:// www.sinope.nl/en/sinope/home.html.
23. http://www.copernic.com/products/sumamrizer.
24. M. Saravanan, P. C. Reghu Raj, V. S. Murthy and S. Raman, Improved Porter's Algorithm for Root Word Stemming, *in Proc. of International Conference on Natural language Processing (ICON'02),* (2002) Mumbai, India, 21-30.