Multi-Document Automatic Text Summarization Using Entropy Estimates

G.Ravindra,N.Balakrishnan,K.R.Ramakrishnan K R
Supercomputer Education and Research Center
Indian Institute of Science, Bangalore-560012, INDIA
{ravi@mmsl.,balki@}serc.iisc.ernet.in and krr@ee.iisc.ernet.in

Abstract

This paper describes a sentence ranking technique using entropy measures, in a multi-document unstructured text summarization application. The method is topic specific and makes use of a simple language independent training framework to calculate entropies of symbol units. The document set is summarized by assigning entropy-based scores to a reduced set of sentences obtained using a graph representation for sentence similarity. The performance is seen to be better than some of the common statistical techniques, when applied on the same data set. Commonly used measures like precision, recall and f-score have been modified and used as a new set of measures for comparing the performance of summarizers. The rationale behind such a modification is also presented. Experimental results are presented to illustrate the relevance of this method in cases where it is difficult to have language specific dictionaries, translators and document-summary pairs for training.

1 Introduction

The fast moving Internet age has resulted in a huge amount of digital data that has redefined paradigms in the field of data storage, data mining, information retrieval and information assimilation. Automatic text summarization is seen as an important off-shoot of this digital era. It is envisaged that machine generated summaries are going to help people assimilate relevant information quickly. It also helps in quick search and retrieval of desired information form a summary database which is a reduced version of a larger database.

Automatic extraction-based text summarization consists of a set of algorithms and mathematical operations performed on sentences/phrases in a document so that the most difficult task of identifying **relevant sentences** can be performed. Definition of relevance is subjective and measures to determine the relevance of a sentence to a summary is mathematically difficult to express. The cognitive process that takes place when humans try to summarize information is not clearly understood, yet we try to model the process of summarization using tools like decision trees, graphs, wordnets and clustering algorithms. Although these pattern recognition techniques may not correctly represent the human cognitive process, these methods are popular and seem to produce reasonable if not perfect results. Techniques that attempt to understand natural language the way humans do, might help produce better summaries, but such systems might not work well for all languages. Many of the methods work well for specific sets of documents. Techniques that rely on features such as position, cue words, titles and headings perform poorly on unstructured text data. Methods that rely on word frequency counts obtained after analyzing the document collection to be summarized, perform well in cases such as multiple news reports on the same event, but fare badly when there is only one document to be summarized.

2 Related Work

In [1], Baldwin and Morton present a query-based summarizer wherein sentences are selected from the document such that all the phrases in the query are covered by these selected sentences. A sentence in the document is said to include a phrase present in the query, if the query and the sentence corefer to the same information unit. In [2], Carbonell and Goldstein introduce a new relevance measure called the Maximal Marginal Relevance(MMR). MMR is a linear combination of "novelty" and "relevance" measured independently. They have applied this measure for ranking documents in an IR environment. They have also applied it to passage extraction as a method for query-based text summarization and in multi-document text summarization. SUMMARIST [3], from the University of Southern California strives to create text summaries based on topic identification and document interpretation. The identification stage filters input documents to determine the most important topics. The interpretation stage clusters words and abstracts them into some encompassing concepts and this is followed by the generation stage. SUMMARIST uses traditional IR techniques augmented with semantic and statistical methods. WordNet has been used to identify synonyms and sense disambiguation algorithms have been used to select the proper word sense. Latent Semantic Analysis [4] has been used to solve the term dependency problem. Topic identification has been realized using a method called "optimal position policy (OPP)". This method tries to identify those sentences that are more likely to convey maximum information about the topic of discourse. Cue phrases such as "in conclusion", "importantly" etc have also been used for sentence selection. As reported in [5] use of location and cue phrases can give better summaries than just word frequency counts. The serious drawback with the OPP method is that location and cue phrase abstraction is dependent on the text genre.

Barzilay and Elhadad [6], present a method that creates text summaries by using lexical chains created from the document. Lexical chains are created by choosing candidate words [7] from the document , finding a related chain using relatedness criteria and introducing that word into the chain. Relatedness of words is determined in terms of the distance between their occurrences and shape of the path connecting them in the WordNet database. For summarizing the document, strong chains are selected from the lexical chain and sentences are chosen based on chain distribution.

Gong and Liu [8], present and compare sentence extraction methods using latent semantic analysis and relevance measures. The document is decomposed into sentences where each sentence is represented by a vector of words that it is composed of. The entire document itself is represented as a single vector of word frequencies. The word frequencies are weighted by local word weights and global word weights and these weighted vectors are used to determine the relevance. A sentence that has the highest relevance is selected from the document and included in the summary and then all the terms contained in this sentence are removed from the document vector. This process is continued till the number of sentences included into the summary has reached a pre-defined value. The latent semantic analysis approach uses singular value decomposition (SVD) to generate an index matrix which is used to select appropriate sentences to be included in the summary. The SVD operation is capable of capturing the interrelationships between words so that they can

be semantically clustered.
MEAD summarizer [9, 10], uses a centroid-based summarization scheme for multiple document summarization. The scheme creates centroids which are pseudo-documents consisting of words with a "word count x IDF" measure greater than a threshold. Sentences containing words from these centroids are more indicative of the topic of the cluster. Scores are computed by a weighted sum of proximity to the cluster centroid, length and position values.

Our work uses an entropy measure to rank sentences in order of relevance based on "past knowledge" in the particular domain which the documents to be summarized belong to. The method works well for single document as well as multiple-documents as redundancy in information is not the only criterion used for sentence selection. Document-summary pairs have not been used for feature extraction and other statistical calculations. The principle reason for taking this approach is governed by some of the problems encountered in the context of Indian languages. There are almost as many as 45 dialects and information in the form of news articles, magazines are available in many of them. It becomes a very difficult task to generate dictionaries, have document-summarizer pairs for training and have language translators. Hence a language independent framework that can be trained with only raw (untagged,unparsed etc.) data can be used. This framework when applied to documents in English language has shown encouraging results. Further the importance of "past/background knowledge" in improving the summarizer performance has been shown. The remaining part of the paper is organized as follows: section-3 explains the sentence selection procedure using entropy estimates. Section-4 describes the measures used for comparing summaries followed by experimental results in section-5. Section-6 concludes the paper with some observations.

3 Sentence Selection Technique

Any document set to be summarized is first classified as belonging to a particular domain. We use a database of documents clustered into various domains/topics. Using an IR technique the document set to be summarized is first classified as belonging to one of these domains. Once the domain/topic has been identified, an entropy model for the various words and collocations in the identified domain is generated. Documents available in this identified domain constitute the training data set. The calculated entropy values are applied to each of the sentences in the document set to be summarized and a sentence ranking formula is computed. In the remaining part of the paper we use summary and extract interchangeably.

3.1 Redundancy Removal

The document set to be summarized is subjected to a pre-processing step aimed at removing redundant information. The entropy-based sentence selection formula cannot be used to detect if two or more sentences are similar in word composition. We use a graph representation of sentences to detect and remove redundancy before applying the entropy based ranking formula.

Every sentence is represented as a node in a directed graph. A link is established from one node to another if at least 3 non-stop-words are common to them. If the parent node represents a longer sentence than what the child node represents, then the link weight is the ratio of number of words common to both the sentences to the length(number of non-stop words) of the child node. If not, the link weight is the ratio of common words to the length of the parent node. For every parent node, those child nodes which have a link weight greater than a particular threshold and which are shorter than the parent node are excluded from the sentence ranking process. Hence, sentences that have been repeated and sentences that are almost similar in word composition are thrown away.

As a second option for removing redundancy we use latent semantic indexing (LSI). The result of this step is a collection of candidate sentences for the entropy-based method to choose from. LSI uses singular value decomposition (SVD) on a weighted term x sentence matrix to produce the right and left singular vectors and the singular values in descending order of importance. LSI-based summarizers select sentences based on the magnitude of singular values. But in reality, singular values do not indicate the strength of a topic in terms of its importance. On the other hand LSI helps to efficiently remove redundant information which may be expected in a multi-document summarization case, even in the case of synonymy and polysemy. We exploit this capability of LSI to effectively remove redundancy and use entropy-scores to compensate for its inability to choose important sentences.

3.2 Entropy Measure for Sentence Selection

Let T= $[t_1,t_2,t_3.......,t_M]$ be a set of M sentences in a training set for a particular domain or topic. We define a window of length L within which word collocations are counted. We define the vocabulary $V = \{v_1,v_2,.....,v_{|V|},\}$ for this domain,as the set of all unique non-stop words that occur in the training data set. We represent any sentence t_i as a sequence of non-stop words $[w_1,w_2.........,w_n]$ where n is the length of the sentence t_i in terms of number of non-stop words. Different collocations starting with the word w_i discovered in a sentence can be written as $\{(w_i,w_{i+1}),(w_i,w_{i+2}),......,(w_i,w_{i+L})\}$, where L is the length of the collocation window. Let $C(v_i,v_j)$ be the number of collocations with the first word v_i and the second word v_j in the collocation pair $<v_i,v_j>$. The probability that the word v_j occurs as the second word given v_i as the first word is

$$p(v_j|v_i) = \frac{C(v_i,v_j)}{\sum_{j=1}^{j=|V_i|} C(v_i,v_j)} \tag{1}$$

where $|V_i|$ is the number of unique collocations discovered with V_i as the first word. The uncertainty in the occurrence of a word from the vocabulary in the second position of a collocation, given the first word, can be expressed using the conditional entropy relation as

$$H(v_i,x) = -\sum_{j=1}^{j=|V_J|} p(v_j|v_i)ln[p(v_j|v_i)] \tag{2}$$

where $|V_J|$ is the number of unique words from the vocabulary in the collocations with v_i as the first word. Instead of including the entire vocabulary as the word-set participating in the collocation, this method includes only that sub-set of the vocabulary which contributes non-zero values to the entropy score. The probability of occurrence of the collocation pair $<\boldsymbol{v_i},\boldsymbol{v_j}>$ is $p(v_i)p(v_j|v_i)$ and this is not the same as the probability for the pair $<\boldsymbol{v_j},\boldsymbol{v_i}>$ as the order has to be preserved in a collocation. Hence the uncertainty when the word v_i is encountered can be computed as

$$-\sum_{j=1}^{j=|V_J|} p(v_i)p(v_j|v_i)ln[p(v_i)p(v_j|v_i)] \tag{3}$$

which can be further expressed in terms of conditional entropy and word probability as

$$E_F(v_i) = -p(v_i)\left\{ln[p(v_i)] - H(v_i, x)\right\} \tag{4}$$

This we call the forward entropy equation. Similarly the uncertainty in the occurrence of the first word, given the second word of the collocation is given by

$$H(y, v_j) = -\sum_{i=1}^{i=|V_I|} p(v_j|v_i) ln[p(v_j|v_i)] \tag{5}$$

where $|V_I|$ is the number of unique words from the vocabulary in the collocations with v_j as the second word. This results in the backward entropy equation which can be expressed as

$$E_B(v_i) = -p(v_i)\left\{ln[p(v_i)] - H(y, v_j)\right\} \tag{6}$$

The forward and backward entropies for all the words in the training set is computed. As can be seen, the entropy computation procedure does not use language specific features and parsing techniques to determine measures for words. The basic philosophy is to consider meaningful text as just a sequence of symbols(words,phrases etc.) and see how to use such a frame work for sentence selection in summarization applications.

3.3 Sentence Ranking

Once the forward and backward entropies have been computed for a particular domain/topic the summarizer is ready to perform sentence selection tasks on the document to be summarized.

For every sentence $\boldsymbol{t_i}$ in the document set to be summarized, an entropy score is computed as

$$H(t_i) = \frac{\sum_{k=1}^{k=|t_i|} E_F(w_k) + E_B(w_k)}{|t_i|} \tag{7}$$

where $|\boldsymbol{t_i}|$ is the number of non-stop words in the sentence. Estimated entropy values are not substituted if the forward and backward entropies for a particular word cannot be determined form the training data. After entropy scores have been computed for all sentences in the document(s) to be summarized, sentences are ranked in the ascending order of entropy scores. The first $\boldsymbol{N}$ sorted sentences are selected as the candidate sentences to be included into an extract or for further processing.

4 Comparing Summaries

A common approach to judge the quality of summaries is to allow human experts to score them on a scale of 0-5 or 0-10. Human judges can vary in their opinions and assigning scores is based on the extent to which the expert knows about the subject. Getting experts and taking their opinion is a laborious task and is difficult to establish consistency. Other method followed is to count the number of sentences that are common between machine generated and human summaries and then compute precision and recall scores. Exact sentence matches do not take into account slight variations in word composition. Hence we use modified precision,recall and f-score measures to compare summaries. As we use a fuzzy set paradigm, these new measures are called as fuzzy precision, recall and f-score measures.

4.1 Membership Grades

Let $H=\{h_1,h_2,h_3,...,h_{|H|}\}$ be a human generated summary that consists $|H|$ sentences. Let $M=\{m_1,m_2,......,m_{|M|}\}$ be a machine generated extract consisting of $|M|$ sentences. Each sentence is represented by a unit normal vector of word collocations. The dimensionality of each of these vectors is the total number of unique collocations discovered in the machine and human generated summaries, using the collocation window which was mentioned in the previous section. Let R_H and R_M be the matrices whose columns are vectors representing the human summary and machine summary respectively. Let $F = \{f_1,f_2,f_3,...,f_{|H|}\}$ be a set of fuzzy sets corresponding to each of the sentences in H. The matrix product $\Phi = R_H^T \times R_M$ gives the matrix of membership values of each sentence belonging to M in F . For example, the i^{th} row of Φ gives the membership value of each sentence of M in the fuzzy set f_i. The human summary can be considered as union of the sets of F. If $\mu_{H_k}(m_j)$ is the membership grade in the set f_k then the membership grade in the set $H = \bigcup_{k=1}^{k=|H|} f_k$, is given by $\mu_H(m_j) = max_i[\Phi_{i,j}]$. This relation is used in computing the precision scores.

Similarly the membership grade of the sentence h_j in the machine summary becomes $\mu_M(h_j) = max_i[\Phi_{ji}]$ and this relation is used to compute the recall scores.

4.2 Fuzzy Precision, Recall and F-score Measures

Precision,recall and f-score have been used as one of the standard methods for evaluating summaries. Precision is defined as the ratio of number of sentences of the machine summary that has an exact match with the sentences in the human summary, to the number of sentences in the machine summary. Hence the fuzzy precision measure can be defined as

$$Fp = \frac{\sum_{j=1}^{j=|M|} \mu_H(m_j)}{|M|}$$

Recall is defined as the ratio of number of sentences of the human summary that has an exact match with the sentences in the machine summary, to the number of sentences in the human summary. Hence, fuzzy recall measure can be defined as

$$Fr = \frac{\sum_{j=1}^{j=|H|} \mu_M(h_j)}{|H|}$$

The fuzzy F-score becomes $\frac{1}{(\lambda \frac{1}{Fp} + (1-\lambda)\frac{1}{Fr})}$.

5 Results and discussion

5.1 Experimental Conditions

MEAD, LSI-based summarizer and the entropy-based summarizer were evaluated on a summary data set consisting of 13 document clusters and their corresponding summaries. These 13 clusters were chosen from the DUC2002 data set released by NIST. The reason for choosing particular document sets from this data set is, because relevant information about the topics that they discuss were available in our topic database. Each document cluster consists of two possible human summaries compressed to 400 words. Each cluster can have between 7 and 12 documents, all of them containing information about a common topic. The topics covered by these 13 document sets fall under the following categories:
hurricanes, earthquakes, explosions, personalities, floods, Berlin wall history, turmoils and scientific pursuits

The MEAD summarizer (version-3.05) was used with the default configuration file parameters. Compression basis was "sentences" and compression percentage was varied so that each document cluster was summarized to produce between 20 and 22 sentences. This translates to a compression varying between 6-13% for the original document set, in terms of sentence count. The LSI-based summarizer was implemented fully in C using matlab-library functions. Full rank SVD was computed on the word x sentence matrix for the cluster to be summarized. The columns of the right-singular vectors were sorted in the ascending order and the sentences corresponding to first 20 largest singular values were selected to be included into the summary. The 'word x sentence' vectors were normalized and $ln[N/c]$ was used as the IDF weighting formula before SVD computation.

Table 1: Number of sentences per cluster and the reduced rank after SVD

Sentences per cluster	Reduced Rank
243	134
118	80
219	133
212	127
309	172
177	112
120	84
236	146
334	194
332	191
199	124
111	76

For the method where LSI was used as a pre-processing step before entropy-based sentence selection was used, all the unique sentences corresponding to the largest index values were used as the reduced set of sentences. Table-1 shows the number of sentences in the original document set and the number of sentences selected as the output of the preprocessing step.

Word entropies were computed after stop-words were removed but word stemming was not performed. The window length for collocation discovery while computing entropy values was set to 4 non-stop words.

5.2 Experimental Results and Discussion

Precision, recall, f-score and their fuzzy variants were used as measures to evaluate the summaries produced by the summarizers. Hence forth we call the entropy-based summarizer using graph representation for redundancy removal as “E-G” or “Entropy” method and entropy-based summarizer with LSI for redundancy removal as “E-LSI” or “LSI+Entropy” method. The precision and recall values for E-G method, MEAD, LSI and E-LSI using exact sentence-match are shown in Fig.1. Fscores corresponding to the maximum scoring summary for each document, have been listed in Table-2.

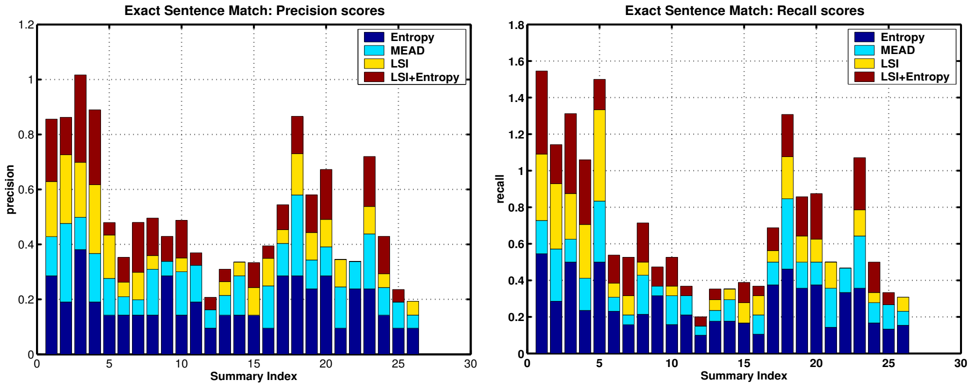

Figure 1: Precision and Recall scores using exact sentence match

There are some cases where the increase in these scores for E-G method is as high as 0.25 which translates to selecting 3-4 sentences more than the other methods. At the same time there are 2 cases where the other methods have out performed E-G method by a maximum difference of 0.1 showing that this method selected 1 sentence less than the other methods. Exact sentence-match shows that on an average, E-G method selects 2 sentences more than the other methods. Sentence redundancy removal using graph-based method resulted in the same quality of summary as LSI-based removal would give. But exact sentence match with the human summary shows 3 cases where E-LSI method produced a summary in which none of the sentences matched with the human summary. This can be attributed to inaccuracy of SVD when the matrices are sparse. Another reason is that when there are more than two sentences having the same index value only one of them is chosen. Table-3 shows the average increase in precision,recall,f-score and number of sentences selected when evaluation was done using exact sentence match.

Table 2: Exact Sentence Match: Fscore

Document Type	LSI	MEAD	LSI+Entropy	Entropy
earthquake	0.2941	0.2857	0.3030	0.3750
hurricane2	0.2703	0.1765	0.3684	0.4324
johnmajor	0.2400	0.1905	0.1143	0.2222
explosion	0.1026	0.1875	0.1951	0.1714
hurricane1	0.0513	0.1579	0.1463	0.3000
kashmir	0	0.1176	0.0488	0.2000
hubble	0.0541	0.1290	0.0513	0.1579
india	0.1053	0.1250	0.1000	0.1538
wall	0.1818	0.3333	0.1714	0.3529
flashflood	0.1176	0.1212	0.2105	0.3243
aquino	0.1176	0.1765	0	0.2778
sakharov	0.1176	0.2353	0.2222	0.2857
ship	0.0606	0.1111	0.0541	0.1176

This is for the case where E-G method was compared against the other three methods.

Table 3: Average improvement in scores with exact sentence match

Exact Match	Precision	Recall	F-score	Average increase in matching sentences
MEAD	0.0622	0.1196	0.0842	1.8
LSI	0.0979	0.1365	0.1131	2.15
E-LSI	0.0784	0.1078	0.0891	1.5

Exact sentence match may not be a good criterion especially if a machine has to judge a summary based on the above mentioned metrics. Sentences that are very similar in word composition but not having the same words and word sequence will not be considered as a match. For example the sentence "JAMMU-KASHMIR IS THE ONLY STATE WITH A MOSLEM MAJORITY IN PREDOMINANTLY HINDU INDIA" and the sentence "HINDU DOMINATED INDIA ACCUSES PAKISTAN OF ARMING AND TRAINING KASHMIRI MILITANTS FIGHTING FOR SECESSION OF JAMMU KASHMIR INDIA S ONLY STATE WITH A MOSLEM MAJORITY" would end up as sentences that are completely different in the case of an exact sentence match. But a fuzzy sentence match would give a partial match score which is the true picture. More over, by using word collocations as already mentioned, the above two sentences can be found similar with regard to the collocations <JAMMU KASHMIR> <HINDU INDIA> <MOSLEM MAJORITY>. By using word pairs we are imposing tighter bounds on information and by using the 4-word window there is flexibility in finding similarity when there are adjective words and other word modifiers. For example, the second sentence does not contain <HINDU INDIA> as consecutive words but within the window [HINDU DOMINATED INDIA ACCUSES] the word pair can be found. The Fuzzy precision,recall and fscore measures for the different summarizers is shown in Fig.2 and Table-4. The average increase in these scores over MEAD,LSI and E-LSI for E-G method is shown in Table-5. The table also shows that on an average, the E-G method chooses 3 sentences more than the other two methods.

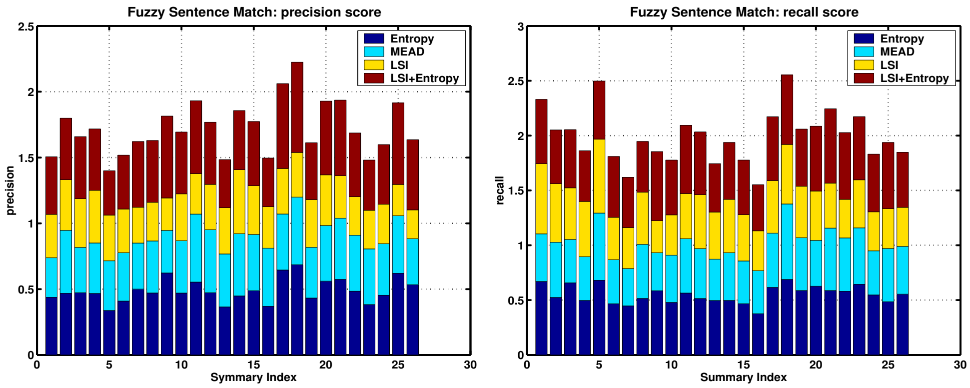

Figure 2: Precision and Recall scores using Fuzzy sentence match

6 Conclusion

As is evident from the graphs, the performance of the summarizers can be seen to be different based on the type of scoring metric used. In general entropy-based sentence selection scheme can be expected to perform better. If the background knowledge required to compute entropies is not sufficient many sentences may go un-scored. Hence no decision can be taken as to whether such sentences can be included into the summary or not. LSI on the other hand does not require any background knowledge but relies purely on

Table 4: Fuzzy Sentence Match: Fscore

Document Type	LSI	MEAD	LSI+Entropy	Entropy
earthquake	0.3839	0.3121	0.5287	0.5288
hurricane2	0.2793	0.2363	0.4629	0.5040
johnmajor	0.3262	0.2859	0.2163	0.3895
explosion	0.2119	0.3299	0.2969	0.3708
hurricane1	0.1906	0.2651	0.4421	0.5039
kashmir	0.1855	0.2423	0.2990	0.3848
hubble	0.3389	0.2339	0.1866	0.2767
india	0.2547	0.3024	0.2782	0.2507
wall	0.2547	0.4572	0.3161	0.4957
flashflood	0.2477	0.2567	0.3600	0.4499
aquino	0.2004	0.3452	0.1965	0.3658
sakharov	0.1931	0.3676	0.3853	0.4341
ship	0.1174	0.2709	0.1477	0.2719

Table 5: Average improvement in scores with Fuzzy sentence match

Fuzzy Match	Precision	Recall	F-score	Average increase in Matching Sentences
MEAD	0.0835	0.1465	0.1104	2.78
LSI	0.1423	0.1574	0.1521	3.19
E-LSI	0.0670	0.0752	0.0717	1.15

redundancy in terms of word frequencies. Hence LSI-based summarization cannot be used while trying to extract sentences from a single document. MEAD on the other hand uses centroid method which allows identification of the various topics that are in the document. This is more efficient than LSI. Hence sentences that are relevant to each of those topics are selected. As already mentioned all the topics may not be of importance when the final extract is created. There can be multiple sentences belonging to the same topic that are important. The entropy-based selection framework allows sentence extraction method suited for most languages where words are separated by definite markers. As information related to the document layout is not considered while scoring, entropy-method can be used more efficiently for summarizing un-structured text.

References

[1] Baldwin.B., Morton.T.: Dynamic co-reference based summarization. In: Proc. Third Conference on emperical Methods in Natural Language Processing. (1998) 630–632

[2] Carbonell.J.G, Goldstein.J: Use of mmr diversity-based re-ranking for recording documents and producing summaries. In: Proc.ACM (SIGIR'98). (1998)

[3] Hovy.E.H, Lin, C.Y.: 8. In: Automated Text Summarization in SUMMARIST. MIT. Press, cambridge Massachusetts, London,England (1999)

[4] Deerwester.S, D., et al.: Indexing by latent semantic analysis. American Society for Information Science **41** (1990) 391–407

[5] Paice, C.: Constructing literature abstracts by computer:techniques and prospects. Information Processing and Management **26** (1990) 171–186

[6] Barzilay.R, Elhadad.M: Using lexical chains for text summarization. In: Proc. Workshop on Intelligent Scalable Text Summarization(madrid-spain). (1997)

[7] Morris.J, Hirst.G: Lexical cohesion computed by thesaural relations as an indication of the structure of text. Computational Linguistics **17** (1991) 21–43

[8] Yihong Gong, X.L.: Generic text summarization using relevance measure and latent semantic analysis. In: Proc.ACM (SIGIR'01). (2001) 19–25

[9] Radev, D., Budzikowska.M: Centroid-based summarization of multiple documents: sentence extaction, utility-based evaluation and user studies. In: Proc.(ANLP/NAACL'00). (2000)

[10] Dragomir Radev, V.H., McKeowen, K.R.: A description of the cidr system as used for tdt-2. In: Proc.DARPA Broadcast News Workshop, Herndon. (1999)