Fuzzy Graph Documet Model Representation of Document Summaries Using Eigen Vector Approach

Sruthimol M P

Assistant Professor,

College of Engineering, Vadakara

Abstract— Automatic document summarization is the process of reducing a text document into a summary that retains the points highlighting the original document. Ranking sentences according to their importance of being part in the summary is the main task in summarization. This paper proposes an effective approach for document summarization by sentence ranking. Sentence ranking is done by vector space modeling and Eigen analysis methods. Vector space model is used for representing sentences as vectors in an n-dimensional space assigning tf-idf weighting system. The principal eigen vectors of the characteristic equation will rank the sentences according to their relevance. Experimental result using standard test collections of DUC2001 corpus and Rouge Evaluation system shows that the proposed sentence ranking based on eigen analysis scheme improves conventional Tf-Idf language model based schemes.

Index Terms— Text summarization, Tf-Idf weighting, Vector Space Model, Eigen Analysis

1 Introduction

From a purely applied perspective, Automatic summarization is one of the applications of NLP that aims to act as an interface between the information needs of persons and the huge amount of information that is publicly available to satisfy this information, specially on the World Wide Web. Text summarization has become essential to provide enhanced mechanisms to perceive and present effective textual information. In everyday life, we come across various forms of summaries without consciously recognizing them as such. For example, the news headlines which summarize the materials present in the news. The scoreboard which shows various statistics of a cricket match summarizes what is happening on the ground. The trailer of a movie. The demo of software which is a summary of what will appear in the actual software. The abstract in scholarly articles is a summary written by the authors. Regardless of the media, type and objective, the common thing is that all summaries are condensed representations of their source. Only information that captures the central meaning or theme of the input, or is deemed relevant to its user, is retained in the summary.

The text summarization is an area of research that has attracted the interest of many researchers through the years, because it can contribute to gain a better knowledge for the people who produce and understand language. Neural networks, decision trees, graph theory, regression models, ontology, latent semantic indexing, fuzzy logic and swarm intelligence are the various methods that employ summarization. The main problem for creating an automatic text summary is to extract the most relevant information in the source document. This paper proposes a domain independent document summarization approach by sentence extraction based on eigen analysis. The system uses a vector space model to extract the sentence specific features of the document. The features are represented as a weighted term frequency matrix. Then the eigen values and eigen vectors of the weighted term frequency correlation matrix are computed. The eigenvectors of the characteristic equation will give the information about the relevance of sentence which is to be included in the final summary.

The paper is organized as follows: Section II discusses the related work that has undergone in the area of text summarization. Section III explains the proposed methodology used for the extractive summarization system. In the section IV discusses the evaluation measures for the system. Section V reports the experimental results after testing the system

2 Literature Survey

2.1 Review Stage

Interest in automatic text summarization, arose as early as the fifties. An important paper by Luhn(1958) [1], suggested a method to weight the sentences of a text document as a function of high frequency words. Baxendale (1958) [2] used sentence position in the text as feature for extracting the important sentence of documents. Another method proposed by Edmundson (1969) [3] which used cue method, title method and location method, in addition to the standard keyword methods for determining the sentence weight. After 1990's exploration in the field of automatic summarization which mainly based on the natural language analysis. Yong et al. [4] worked on developing an automatic text summarization system by combining both the statistical approach and neural networks to train and learn the relevant features of sentences. To summarize one hundred English articles Mohamed Abdel Fattah & Fuji Ren [5] applied genetic algorithm (GA) and mathematical regression (MR) in order to obtain a suitable combination of feature weights sentences. Hamid et al. [6]

proposed a new method to optimize summarization process based on fuzzy logic by selecting a set of features such as sentence length, sentence position, titles similarity, keywords similarity, sentence-to-sentence correlation and occurrence of proper names.

The main trends that can be identified in summarization study are the usage of machine learning algorithms. Joel Larocca Neto et al. [7] applied trainable machine learning algorithms which employs a set of sentence features extracted directly from the original text to train the document. The trainable algorithms employed are Naive Bayes and C4.5. Vorgelegt von [8] discussed a new machine learning algorithm to extract concept, keyword and tag recommendation. Two-level learning hierarchy (TLLH) to extract concepts from tagged textual contents.

Jen-Yuan Yeh [9] proposes a novel graph-based ranking method, iSpreadRank to perform sentence extraction. iSpreadRank used a set of content related documents into a sentence similarity network. Based on such a sentence similarity network model, iSpreadRank utilize the spreading activation theory to formulate a general concept from social network analysis. Another graph based approach proposed by Gunes Erkan and Dragomir R. Radev [10] named LexRank, for computing relative importance of textual units. For calculating sentence importance based on the concept of eigenvector centrality in a graph representation of sentences.

Of the works devoted to summarization, most concentrate on term weighting. Song et al. [11] proposes a novel term weighting scheme based on discrimination power obtained from past retrieval results. Recently an approach different from other methods was presented by Ledeneva et al. [12]. In this paper the sentences are weighted by using the terms derived from the maximal frequent word sequences. Liu et al. [13] discusses a multi-document summarization based on BE vector clustering. In this strategy sentence are represented by BE vectors. BE is a triple representation of sentences and it is more precise as semantic unit than a word. An approach to assist the learners with reading difficulties described by K. Nandhini and S.R. Balasundaram [14]. This approach predicts the summary sentences that are important as well as readable to the target audience with good accuracy. Supervised machine learning algorithms were used for extracting sentences from educational text. Jagadeesh J et.al [15] discusses a new method based on identification and extraction of important sentences in the input document. A set of features are extracted and this features are represented as a vector space.

3 METHODOLOGY

The proposed system uses a vector space model where the index terms in documents are assigned with a non binary weight. The methodologies used to develop the system are discussed here. System pre-processes the document collection. Preprocessing of the text document is done to obtain a structured representation of the original text. Preprocessing include tokenization, stemming, stop word removal to identify the candidate terms in the document.

3.1 Tokenization

During the tokenization process each sentence of the document in the corpus is taken and split into words or co-occurrence patterns. The query given by the user is also tokenized.

3.2 Stemming

A stem is the portion of a word which is left after the removal of affixes (prefixes or suffixes). Stemming is the process of getting the stem for a given word. This process is used in information retrieval task as a recall enhancing approach. The stemming differs from lemmatization, as the stem generated may not necessarily be a lemma (syntactic root word).

3.3 TF-IDF Weghting scheme

The proposed system uses a vector space model where the index terms in documents are assigned with a non binary weight. Let k_i be an index term, d_j be a document, and $w_{i,j} \geq 0$ be a weight associated with the pair (k_i , d_j). Then the weight $w_{i,j}$, quantifies the relevance of the index term for describing the document contents. In vector model, the frequency of a term w_i , in the document d_j , is referred to as the tf factor. And the inverse of the frequency of a term wi in the document collection is referred to as inverse document frequency or idf factor.

$$TF(w_i) = \frac{freq_{i,j}}{max_l freq_{l,j}} \qquad (1)$$

Where, freqi,j is the frequency of term ki in the document dj , which is normalized by the maximum frequency computed over all terms in the document dj

$$IDF(w_i) = \log \frac{N}{n_i} \qquad (2)$$

Where, N is the total number of documents in the collection and ni is the number of documents in which the term ki appears. The tf-idf weighting scheme uses weights which is given by,

$$W_{i,j} = TF * IDF \qquad (3)$$

- *Sruthimol M P is currently working as Assistant professor in Department of Information technology at College of Engineering, Vadakara.. E-mail: sruthinarayan.mp@gmail.com*

This work represents each sentence as a vector of weighted terms. Let W(|W |=n) denote the set of terms in the document group. The vector of a sentence sj is specified by equation, where wi,j is the TF-IDF weight of term ti in sj.

$$s_j = (w_{1,j}, w_{2,j}, ...w_{n,j}) \qquad (4)$$

The degree of similarity between two sentences si and sj is measured by Eq. (5) as the cosine of the angle between the vectors si and sj .

$$sim(s_i, s_j) = \frac{\vec{s_i} \cdot \vec{s_j}}{|\vec{s_i}| \times |\vec{s_j}|} \qquad (5)$$

All the above features are normalized on a 0-1 scale. A weighted combination of all these features is used in calculating the score of sentence.

3.4 Sentence Ranking

The proposed sentence ranking algorithm, which is the major contribution of this work, is based on the concept of eigen analysis. The weighted term frequency vector S_i for each sentence is represented as an adjacency matrix for eigen vector computation. The adjacency matrix A, with rows and columns labeled by sentence nodes, and each entry $a_{i,j}$ is initialized by Eq(6).

$$a_{i,j} = \begin{cases} 0 & i = j \\ sim(s_i, s_j) & i \neq j \end{cases} \qquad (6)$$

3.5 Fuzzy Graph

The edges only go out from a sentence to one or more sentences following it. The forward DAG representation has been implemented ,tested and found that the algorithm seems to be biased and has been consistently ranking the sentences in the latter part of the document better than the starting portions. Hence going by the fact that 2 sentences are similar if one's contents are similar and one follows the other or vice versa we can use an undirected graph. This implementation seems to be giving positive and impressive results than its forward directed counter part. The following rules governing the graph structure would be adhered to : 1)There is no chronological differences between the sentences , only the contents carry importance. 2)There is also no self-edge, the similarity of every sentence to itself is considered to be 0.

4 Implimentation

The overall The overall system architecture of the proposed extraction based summarization system is as shown in Fig 1. The proposed system for Malayalam IR is implemented in two modules: the pre-processing module and sentence ranking module. The pre-processing module involves several stages, as described in Algorithm 1.

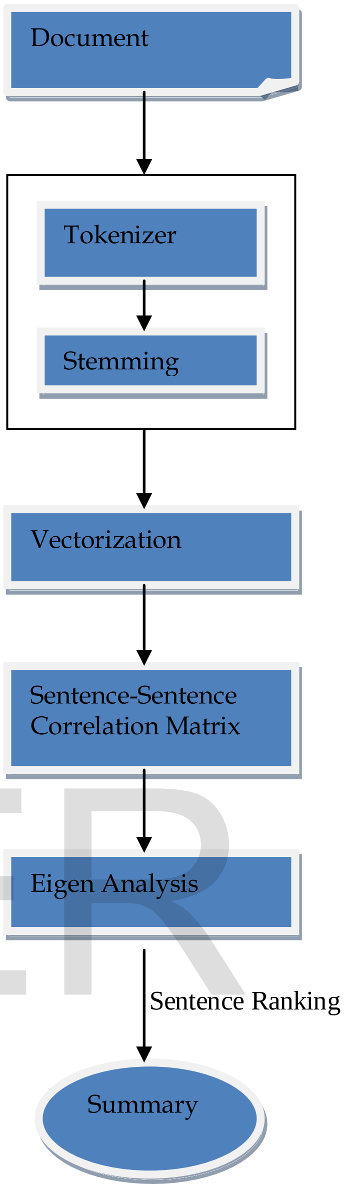

Fig. 1. Proposed System Architecture

Algorithm 1 : Algorithm for Preprocessing Document

Require: Document of type(.pdf,.html)

Ensure: Candidate terms of the document.

1. Read the document and convert it to text document.
2. Split the text document into sentences.
3. Tokenize the sentences into keywords.
4. Eliminate stop words and identify the candidate terms.

This sentence ranking procedure is briefly mentioned in algorithm 2.

```
Algorithm 2 : Algorithm for Sentence Ranking
Require: Candidate terms of the Document.
Ensure: Most important Sentences in the document.
  1. Create the weighted term-frequency vector $S_i$ for each
       sentence $i \varepsilon S$ using TF-IDF weighting.
  2. Initialize a n ×n similarity matrix A[i,j] = 0
  3. Fill the matrix A with similarity values given in eqn
     (5).
  4. Find the eigen values and eigen vectors of the matrix
     M
  5. Let E be the principal eigen vector and compute
      $arg\ maxE_i$
  6. Return corresponding sentence Si as the important
     sentences
```

4.1 Summary Generation

The eigen vector whose values are high are included in the final summary in the same order of their occurrence as in the original text document to keep the overall semantic meaning and readability. Sentences are selected to be included in the summary are arranged according to the ascending order of score value. If more than one sentence has same score then, sentence occurring earlier in the document is given preference over the other. The compression rate of summary is fixed to 20 percentage..

5 Evaluation measures

5.1 DUC corpus and Task

The DUC 2001 data set from DUC (Document Understanding Conferences) was used to examine the effectiveness of the proposed summarization method. DUC task1 is to produce fully automatic single-document summarization. Participants were required to create a generic 100-word summary for each document in a set of 30 topics in DUC 2001. Each set contained documents, per-document summaries (100 word summary), and multi-document summaries (50, 100, 200, 400 word summaries), with sets defined by different types of criteria such as event sets, opinion sets, etc. The manually generated summaries are treated as gold-standard summaries or peer summaries to evaluate the qualities of system generated summaries. Table I shows the data set used for evaluation.

TABLE I. Statistics of dataset

Number of Docs	100
Avg number of Sentences per docs	20
Summary as % of doc Length	20%

5.2 Evaluation Metrics

ROUGE is a widely used evaluation package for text Summarization [20]. It generally counts as a performance indicator the number of co-occurrences between machine- generated and ideal summaries in different word units, such as n-gram, word sequences and word pairs. ROUGE scores at the DUC 2001 are the 1-gram, 2-gram, 3-gram, 4-gram, and longest common substring scores [21]. ROUGE would compare human generated summaries with system generated summaries produced by the application. ROUGE score has been found to correlate very well with human judgments at a confidence level of 95%, based on various statistical metrics [22]. ROUGE will generates three scores for each summary precision, recall and the F-Score of applications as the output to measure the quality of summary.

6 Testing and Result

The data given in Table II mentions the ROUGE scores produced by the proposed system at the DUC 2001 data sets.

TABLE II. Performance Evalation Metrics

Duc 2001	E. M	ROUGE					
		1	2	3	4	L	SU4
d04a	P	0.4137	0.1852	0.3226	0.1495	0.2477	0.7053
	R	0.1200	0.1887	0.1087	0.5934	0.2523	0.7188
	F						
d05a	P	0.2670	0.2857	0.0575	0.1105	0.2045	0.0817
	R	0.4563	0.4902	0.0990	0.1172	0.3495	0.1412
	F						
d06a	P	0.7898	0.7977	0.5143	0.1004	0.1562	0.7777
	R	0.6176	0.2772	0.1800	0.2142	0.5392	0.2734
	F						
d08b	P	0.4137	0.1071	0.3704	0.2600	0.3793	0.1582
	R	0.1200	0.3030	0.1020	0.4698	0.1100	0.4281
	F						
d011b	P	0.1179	0.5797	0.0305	0.1995	0.3000	0.1163
	R	0.5922	0.4040	0.0362	0.0849	0.2100	0.0804
	F						
d012b	P	0.3507	0.2132	0.0038	0.5515	0.0967	0.0431
	R	0.4795	0.1078	0.1980	0.1752	0.4854	0.2209
	F						
d013c	P	0.4565	0.0902	0.0303	0.1773	0.2761	0.1180
	R	0.2187	0.1237	0.0416	0.1497	0.3775	0.1625
	F						
d014c	P	0.3507	0.1555	0.1136	0.3095	0.4347	0.2115
	R	0.4795	0.7368	0.5319	0.0899	0.2083	0.0982
	F						

E.M :Evaluation metric

P: Precision, R: Recall, F: F-measure

The performance of the summarization system is expressed as a function of summary size.

TABLE III. Accuracy Expressed in Average precision, recall, F-measure

Avg. Precision	0.6138
Avg. Recall	0.6773
Avg. F-measure	0.5687

The above table III signifies Avg. precision, Avg. Recall, Avg. F-measure of the proposed system at the 95% confidence level. It is evident that the performance of the model as got improvement in the case of precision, recall and F-measure for various level of compression rate of 20%, 30%, 40% respectively. Overall, the proposed summarization method is found to perform well with competitive results.

7 Conclusion

This paper proposes a novel extractive summarization system based on eigen analysis to rank the important sentence which is to be included in the summary. The proposed summarization method has several advantages over the other systems. First the method proposed here is domain-independent. Second the eigenvalue/eigenvector gives the important features of sentences. Thus the ranking of sentences indicating their relative importance is considered. As a future enhancement proposal we could have easily introduced more parameters like document readability factor. And also developments in summarizations involving multiple languages, hybrid sources and multimedia systems can also be suggested.

References

[1] Luhn P.H, The Automatic Creation of Literature Abstracts, IBM Journal of Research Development, Vol.2, No.2, pp.159-165,1958

[2] Baxendale P, A Machine-made Index for Technical Literature -An Experiment IBM Journal of Research Development, Vol. 2,No.4, pp. 354- 361, 158.

[3] Edmundson, H.P, New Methods in Automatic Extracting, Journal of ACM, Vol.16, No.2, pp. 264-185, 1969.

[4] Yong, S.P., Ahmad I.Z. Abidin and Chen Y.Y, A Neural Based Text Summarization System, 6th International Conference of DATA MINING, pp.45-50,2005.

[5] Mohamed Abdel Fattah and Fuji Ren, Automatic Text Summarization, International Journal of Computer Science, Vol.,No.1, pp.25-28, 2008

[6] Hamid Khosravi, Esfandiar Eslami, Farshad Kyoomarsi and Pooya, Optimizing Text Summarization Based on Fuzzy Logic, ISpringer-Verlag Computer and Information Science, SCI 131, pp.121-130, 2008.

[7] Joel Larocca Neto, Alex A. Freitas, Celso A. A. Kaestner, Automatic Text Summarization using a Machine Learning Approach, Intelligence and Security Informatics Lecture Notes in Computer Science Volume 2665, pp 1-12, 2003..

[8] Vorgelegt von, Machine Learning for Text Indexing Concept Extraction, Keyword Extraction and Tag Recommendation, Proceedings of the

[9] 2007 Conference on Empirical Methods in Natural Language Processing (EMNLP-07).

[10] Jen-Yuan Yeh, Hao-Ren Ke, Wei-Pang Yang, iSpreadRank: Ranking sentences for extraction-based summarization using feature weight prop- agation in the sentence similarity network , Expert Systems with Applications 35, 14511462,2008.

[11] Gunes Erkan, Dragomir R. Radev, LexRank: Graph-based Lexical Centrality as Salience in Text Summarization, Journal of Artificial Intelligence Research 22, 457-479,2004.

[12] Sa-kwang Song,Sung Hyon Myaeng , A novel term weighting scheme based on discrimination power obtained from past retrieval results Information Processing and Management 48, 919930,2012.

[13] Yulia Ledeneva, Effect of Preprocessing on Extractive Summarization with Maximal Frequent Sequences, , MICAI '08 Proceedings of the 7th Mexican International Conference on Artificial Intelligence: Advances in Artificial Intelligence Pages 123-132.

[14] Dexi Liu, Yanxiang, Donghong, Hua Yang, Multi-document summariza- tion based on BE-Vector clustering, CICLing'06 Proceedings of the7th international conference on Computational Linguistics and Intelligent Text Processing Pages 470-479

[15] K. Nandhini, S.R. Balasundaram, Improving readability through extrac- tive summarization for learners with reading difficulties , Egyptian Informatics Journal, 14, 195204, 2013.

[16] Jagadeesh J, Prasad Pingali, Vasudeva Varma, Sentence Extraction Based Single Document Summarization, Workshop on Document Summarization, 19th and 20th March, IIIT Allahabad ,2005.

[17] Divya S., Dr. P. C. Reghuraj, News Summarization Based on Sen- tence Clustering and Sentence Ranking, International Conference of Mathematical Modeling in Computer Management and Medical Sci- ences(ICMCMM),2013

[18] S. Brin and L. Page, The 25,000,000,000 Eigenvector: The Linear Algebra behind The Anatomy of a Large-Scale Hypertextual Web Search Engine, Computer Networks and ISDN systems, 30, pp. 107117, 2005

[19] Taher H. Haveliwala and Sepandar D. Kamvar, The Second Eigenvalue of the Google Matrix ACM Transactions on Internet Technology (TOIT) TOIT Homepage archive Volume 6 Issue 3, August Pages 282-301, 2006

[20] Terry copeck, Diana inkpen, anna kazantseva, Alistair kennedy, Darren kipp, Stan szpakowic, Catch What You Can, School of Information Technology and Engineering University of Ottawa.

[21] Chin-Yew Lin, Looking for a Few Good Metrics: Automatic Summarization Evaluation How Many Samples Are Enough?, In Proceedings of NTCIR Workshop 4, Tokyo, Japan, June 2-4, 2004.

[22] Lin, C.-Y., and Hovy, E. , NeATS in DUC 2002 In Proceedings of the DUC 2002 workshop on text summarization. Philadelphia, PA, USA2002.

[23] http: //en.wikipedia.orgwiki //Automatic summarization

[24] http://www.berouge.com/Pages/default.aspx)