Journal of Theoretical and Applied Information Technology
31st December 2012. Vol. 46 No.2

ISSN: 1992-8645www.jatit.org E-ISSN: 1817-3195

MULTI DOCUMENT SUMMARIZATION USING CLUSTERING

[1]R.C. BALABANTARAY, [2]D. K. SAHOO, [3]M. SWAIN, [4]B. SAHOO
[1]Asst Prof., CLIA Lab, Department of Computer Science, IIIT, Bhubaneswar, Odisha, India
[2]Research Project Fellow, CLIA Lab,IIIT, Bhubaneswar, Odisha, India
[3]Research Project Fellow, CLIA Lab,IIIT, Bhubaneswar, Odisha, India
[4]Research Project Fellow, CLIA Lab,IIIT, Bhubaneswar, Odisha, India
E-mail: [1]rakeshbray@gmail.com, [2]deepsahoo@gmail.com, [3]monalisaswain1988@gmail.com, [4]s.bibhuprasad@gmail.com

ABSTRACT

There are two types of situations in which multi-document summarization would be useful: (1) the user is faced with a collection of dis-similar documents and wishes to assess the information landscape contained in the collection, or (2) there is a collection of topically related documents, extracted from a larger more diverse collection as the result of a query, or a topically-cohesive cluster. In this paper we present the development of an automatic multi- document summarizer based on clustering and sentence extraction. Based on reference document provided by the user, similar type of documents is extracted from a group of documents. We create an n x n similarity matrix among the entire sentence in the similar type of documents which represent sentence level similarity in all sentences in all extracted documents. Then we make clusters of similar sentences using Markov clustering principle. Then in each cluster each sentence is assigned three weights 1.chronological weight (Document level) 2.Position weight (position of sentence in document) 3. Sentence weight (Stastical weight based on term weight). Then we extract best sentences from each cluster. We have tested the system with document of different domain documents and the result is satisfactory.

Keywords:*Multi Document Summarizer, Markov Clustering Principle, Term Weight, Positional Weight, Chronological Weight, Vector Space Model.*

1. INTRODUCTION

A summary is a shorter usually not longer than half of original text. The main aim of summarization is to identify the most salient parts of a text. Usually the salient parts are determined on the following assumptions [1] [2]:

- they contain words that are used frequently;
- they contain words that are used in the title and headings;
- they are located at the beginning or end of sections;
- they use key phrases which emphasize the importance in text;
- they are the most highly connected with the other parts of text;

Following is a list of requirements for multi-document summarization: [10]

Clustering: The ability to cluster similar documents and passages to find related information.
Coverage: The ability to find and extract the main points across documents.
Anti-redundancy: The ability to minimize redundancy between passages in the summary.
Summary cohesion criteria: The ability to combine text passages in a useful manner for the reader.-Thismay include:

Document ordering: All text segments of highest ranking document, then all segments from the next highest ranking document, etc.
News-story principle (rank ordering): present the most relevant and diverse information first so that the reader gets the maximal information content even if they stop reading the summary.
Topic-cohesion: Group together the passages by topic clustering using passage similarity criteriaand present

ISSN: 1992-8645www.jatit.org E-ISSN: 1817-3195

the information by the cluster" centroid passage rank.

Time line ordering: Text passages ordered based on the occurrence of events in time.

Identification of source inconsistencies: Articles often have errors (such as billion reported as million, etc.); multi-document summarization must be able to recognize and report source inconsistencies.

Summary updates: A new multi-document summary must take into account previous summaries in generatingnew summaries. In such cases, the system needs to be able to track and categorize events.

Effective user interfaces:

Attributability: The user needs to be able to easily access the source of a given passage. This could be the single document summary.

Relationship: The user needs to view related passages to the text passage shown, which can highlight source inconsistencies.

Source Selection: The user needs to be able to, select or eliminate various sources. For example, the user may want to eliminate information from some less reliable foreign news reporting sources.

Context: The user needs to be able to zoom in on the context surrounding the chosen passages.

Redirection: The user should be able to highlight certain parts of the synthetic summary and give a command to the system indicating that these parts are to be weighted heavily and that other parts are to be given a lesser weight.

To generate a summary, one must first start with relevant documents that one wishes toSummarize. Multi document summarization requires creating a short summary from a set of documents which concentrate on same topic. Generally an effective summary should be brief andrelevant. Which means the summary should cover all the salient features and main ideas of the documents, it should not contain redundant information and it should be well organized.

In this paper, we propose a multi document summarizer, based on clustering Principle and weight based sentence extraction. Clustering is an important issue in the analysis and exploration of data. There is a wide area of applications as e.g. data mining, VLSI design, computer graphics and gene analysis. See also [3]& [4] for an overview. Roughly speaking, clustering consists in discovering natural groups of similar elements in data sets.

Our system have the following steps, it will take a reference document then it will extract the similar documents (same domain as reference document) from group of documents provided by the user using vector space mode.The similarity between sentences has great influence on the similaritybetween documents. Commonly used approaches are often based on similarity between thekeyword sets (e.g., Dice similarity) or similarity between the vectors of keywords (e.g., cosinesimilarity). These methods seldom consider the semantic meaning of words.

Then an n x n sentence similarity matrix is created among all the sentences of extracted documents.Then Markov clustering principles applied to make sentence clusters. The similarities between words in different sentences have great influence on the similarity between two sentences. Words and their orders in the sentences are two important factors to calculate sentence similarity. Then in each cluster we rank each sentence based on sentence weight, positional weight and chronological weight (document level). Then we short each cluster then extract best sentences & arranged according to their weight.

2. RELATED WORK

Current multi document summarization systems follow clustering and extractive summarization framework. A loose definition of clustering could be “the process of organizing objects into groups whose members are similar in some way”. A *cluster* is therefore a collection of objects which are “similar” between them and are “dissimilar” to the objects belonging to other clusters. Clusters are created among sentences across documents of same domain. Then rank the sentences and extract the most salient sentences to compose summaries for a good coverage of the concepts.

A lot of research work has been done in the domain of multi-document summarization based on clustering and extractive summarization framework.

ISSN: 1992-8645www.jatit.org E-ISSN: 1817-3195

Dragomir R. Radev, Hongyan Jing, Malgorzata stys, Daniel Tam 2004 [5],MEAD is a centroid based multi document based multi document summarizer which generates summaries using cluster centroids produced by topic detection and tracking system used both for single and multi document summaries.

Gunes Erkan, Dragomir R. Radev 2004 [6] used an approach,LexRank, for computing sentence importance based on the concept of eigenvector centrality in a graph representation of sentence. In this model, a connectivity matrix based on intra-sentence cosine similarity is used as the adjacency matrix of the graph representation of sentences.

Rada Mihalcea, Courtney Corley and Carlo Strapparava 2006[7], Used corpus-based and knowledge-based measures of similarity for measuring the semantic similarity of texts which outperforms methods based on simple lexical matching.

Junsheng Zhang, Yunchuan Sun, Huilin Wang, Yanqing He 2011[8]used statistical method to measure similarity between sentences, based on symbolic characteristics and structural information to measure the similaritybetween sentences without any prior knowledge but only on the statistical information of sentences.

Regina Barzilay, Kathleen R. McKeown, Michael Elhadad 1999[9] used a method to automatically generate a concise summary by identifying and synthesizing similar elements across related text from a set of multiple documents and usage of language generation to reformulate the wording of the summary.

Jade Goldstein, Vibhu Mittal, Jaime Carbonell, Mark Kantrowitz 2000[10] used domain-independent techniques based mainly on fast, statistical processing, a metric for reducing redundancy and maximizing diversity in the selected passages, and a modular framework to allow easy parameterization for different genres, corpora characteristics and user requirements.

4. CLUSTERING & EXTRACTION MODEL

We are using a clustering and sentence extraction model in which we tracked the similar documents and then we cluster similar sentences and extracted best sentences using statistical sentence extraction approach. The proposed Clustering and Extraction based multi-document summarization consists of following steps.

4.1Find Relevant Documents Using Vector Space Model.

To generate a summary, one must first start with relevant documents that one wishes tosummarize. We are using the Vector Space model to find the similar document (Same domain) from a group of document given by the document. We are finding the similar documents based on a reference document given by the user.

The document set D = {d1, d2, d3 . . . dk} and reference document(R). We have to find similar documents as R from D. Let |D|=k(Total no of documents in D is k let's say D=k).

Then create an n x m matrix to find the frequency of each term in each document (tfi) i.e. (for example the term 'summary' exists 2 times in D1, 3 times in D2 and once in D3). Where n are total no of terms except stop words in all the documents including the reference document. m is total no of documents including reference document.

Then calculate
Sum of frequency (dfi) of a term in all documents including reference document
$df_i = \sum_{i=1}^{m} tf_i$ where tf_i is frequency of a term in each document.

Inverse Document Frequency (IDF_i)
$IDF_i = \log (D/df_i)$ Where D is total no of documents & df_i is the sum of frequency of a term in all documents including reference document.

Weight of a term in a document w_i

$$w_i = tf_i * IDF_i$$

Now Compute all vector length for each document and Reference document except zero terms.

$$|D_i| = \sqrt{\sum_i (W_{i,j})^2}$$

Now calculate all dot products except zero products

$$R * D_i = \sum_i W_{R,j} W_{i,j}$$

now calculate similarity values

$$Cosine\theta D_i = Sim(R, D_i)$$

Journal of Theoretical and Applied Information Technology
31st December 2012. Vol. 46 No.2

ISSN: 1992-8645www.jatit.org E-ISSN: 1817-3195

$$\text{sim}(\text{R}, D_i) = \frac{\sum_i W_{R,j} W_{i,j}}{\sqrt{\sum_j (W_{\text{R,j}})^2} \sqrt{\sum_i (W_{\text{i,j}})^2}}$$

those documents are similar which more close to Reference document R.

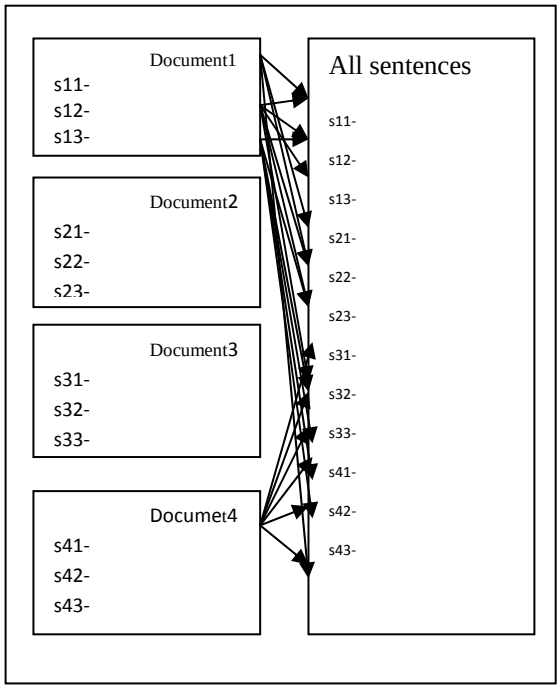
We are taking the documents whose
Sim(R,D_i) >= cos (30^0) =0.86

4.2Creation of Similarity Matrix.

After Getting the Similar documents we are creating an n x n sentence similarity matrix. If total no of sentences in all documents including the reference document is n then each sentence is compared with rest (n-1) sentences. Similarity of two sentences is defined on statistical approach as below.

$$\text{sim}(s1, s2) \frac{\text{No of semantic similar tokens}}{\text{Total no of tokes in both the sentences}}$$

the score sim (s1, s2) can be more accurate set if stemmer and lexicon are used to match the equivalent words. WordNet can be used to match the equivalent words if they are synonymous.Once the similarity matrix is created then it is used to make clustering of sentences using Markov's clustering principle.



4.3 GroupingSimilar Sentences Using Markov's Clustering Principle.

The MCL algorithm is designed specifically for the settings of simple and weighted graph. It is possible to apply MCL and identify cluster of similar sentences as multi document summarization problem can be represented in the framework of weighted graph structure. MCL process consists of following steps;in the first step the sentence similarity matrix which is the associated matrix of the document is normalized.

1.0	0.21	0.19	0.290	0.30	0.0	0.18
0.13	1.0	0.14	0.26	0.22	0.0	0.21
0.11	0.11	1.0	0.17	0.18	0.0	0.18
0.2	0.39	0.33	1.0	0.32	0.0	0.26
0.22	0.39	0.43	0.38	1.0	0.0	1.24
0.0	0.0	0.0	0.0	0.0	1.0	0.03
0.12	0.25	0.29	0.21	0.84	0.5	1.0

Normalize

0.56	0.09	0.08	0.13	0.1	0.0	0.06
0.07	0.43	0.06	0.11	0.08	0.0	0.07
0.06	0.05	0.42	0.07	0.06	0.0	0.06
0.11	0.17	0.14	0.43	0.11	0.0	0.08
0.12	0.17	0.18	0.16	0.35	0.0	0.4
0.0	0.0	0.0	0.0	0.0	0.67	0.01
0.07	0.11	0.12	0.09	0.29	0.33	0.32

in the 2nd step of MCL process simulates random walks in the Markov graph by iteratively performing two operations, expansion and inflation. The process will converge to a limit. The MCL process generates a sequence of stochastic matrices starting from the given Markov matrix. Expansioncoincides with taking the power of stochastic matrix using the normal matrix product and Inflation corresponds to taking the Hadamard power (entry wise power) of the matrix, followed by scaling step, such that the resulting matrix is stochastic again, i.e., the matrix elements correspond to probability value. We got the cluster as below:

Cluster No1
Document List=[C:\Users\CLIA\inv\d1.txt,
C:\Users\CLIA\inv\d4.txt]
Line No in doc= [1, 5]

Cluster No2
Document List=[C:\Users\CLIA\inv\d1.txt,
C:\sers\CLIA\inv\d1.txt,
C:\sers\CLIA\inv\d4.txt]
Line No in doc= [2, 3, 4]

Journal of Theoretical and Applied Information Technology
31st December 2012. Vol. 46 No.2

ISSN: 1992-8645www.jatit.org E-ISSN: 1817-3195

4.4 Calculate The Weight Of Each Sentence Of Each Cluster

Each sentence in each cluster is given three weights as below;
a. Sentence weight
b.Chronological weight
c.Position weight
Now in each cluster weight of each sentence is calculated as below

Weight of sentence = Sentence weight + Chronological weight + Position weight;

a. sentence weight is calculated as below

We are assigning a weight value to each individual term in the sentence [11]. We are counting the frequency each term across the term Then weight is calculated as:

$$\text{Term}_{\text{weight}}(\text{Tw}) = \text{Frequency of the term} * \log\left(\frac{n}{df}\right),$$

Where n = Total No. of Sentence exist in the document.
df = No. of sentence contains the Term.
sentence weight(Sw) $= \sum_{i=0}^{t} \text{Tw}$, Where t is total no of terms in the sentence

b. Chronological Weight is calculated as below

This is a document level weight. This weight is same for all the sentences in a document. From the properties of the document we get the date of creation of document, and then we get the difference of creation date and current date in no of days. Then we normalized that value using Min – Max Normalization. We are giving more value to recent created or recent updated documents because we assume that recent documents are more updated and more accurate and more informative.

Chrono_weight (Cw) = 1-((days-min)/range);
Example

Documents	D1	D2	D3
Diff in days	5	15	25

Here
Max = 25, Min=5
Range=Max-Min=25 – 5 = 20
for document D1
Chrono_weight (Cw) =1-((5-5)/20) =1

c. Positional Weight calculation

we are giving a weight to the sentence according to its position in the document i.e. sentence no one have different weight value than sentence no two because we assume that sentence with initial positions are more informative. Position weight is calculated as below

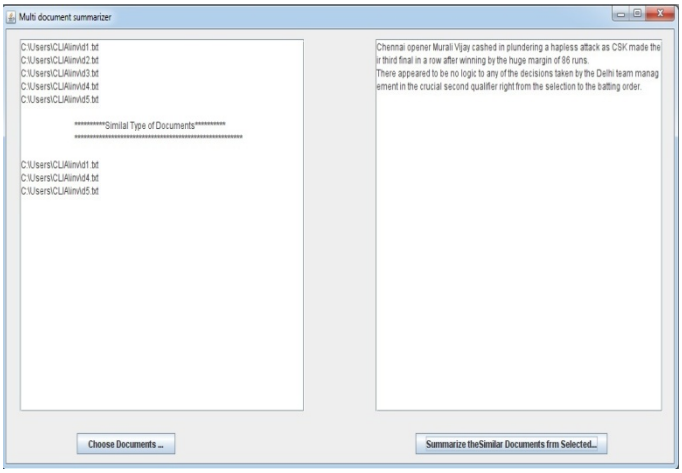
Position Weight (Pw) = (total_sentences_in_document – line_no _of _sentence)/ tota_sentences_in_the_document;
so sentence weight can be written as:
Sentence Weight Sw = Sw + Cw + Pw

4.5 ExtractTheBest Sentences From Each Cluster

Now; we get the weight of each sentence of each cluster. Then we sort the sentences of each cluster according to weight of sentences. 25% (upper ceiling) of total no of sentence present in each cluster is extracted and display as summary of similar documents as per step one.

Cluster No1
Document List=[C:\Users\CLIA\inv\d1.txt, C:\Users\CLIA\inv\d4.txt]
Line No in doc= [1, 5]
Weight of sentence= [15.999238, 14.908123]

Cluster No2
Document List=[C:\Users\CLIA\inv\d1.txt, C:\Users\CLIA\inv\d1.txt C:\Users\CLIA\inv\d4.txt]
Line No in doc= [3, 2, 4]
Weight of sentence=[37.679955, 19.908123, 19.408123]



Journal of Theoretical and Applied Information Technology
31st December 2012. Vol. 46 No.2

ISSN: 1992-8645www.jatit.org E-ISSN: 1817-3195

5. RESULT & DISCUSSION

We have tested our system with document of 5different domains.
(1) Current News.
(2) Health.
(3) History.
(4) Grate Personality.
(5) Sports News.

Each domain consists of 5 to 10 text documents. Then the summary is evaluated using Kappa measure.From each domain we are extracting similar documents, so no of documents given, no of similar documents extracted and no of sentences in the extracted document are given below.

1. Domain: Current News
No of documents - 10
No of similar documents - 4
No of sentences in each document - (5, 5, 3, 7)
2. Domain: Health
No of documents - 8
No of similar documents - 4
No of sentences in each document - (3, 6, 5, 5)
3. Domain: History
No of documents - 8
No of similar documents - 4
No of sentences in each document - (6, 4, 4, 6)
4. Domain: Grate Personality
No of documents - 5
No of similar documents - 2
No of sentences in each document - (7, 7)
5. Domain: Sports News
No of documents - 5
No of similar documents - 3
No of sentences in each document - (4, 4, 4)

	B (Yes)	B (No)
A(Yes)	3	0
A(No)	1	1

	B (Yes)	B (No)
A(Yes)	4	0
A(No)	0	1

	B (Yes)	B (No)
A(Yes)	3	1
A(No)	0	1

	B (Yes)	B (No)
A(Yes)	2	0
A(No)	1	2

	B (Yes)	B (No)
A(Yes)	4	0
A(No)	0	1

$$k = \frac{\Pr(a) - \Pr(e)}{1 - \Pr(e)}$$

Calculate the value of k for each table
K1=0.54
K2=1.00
K3=0.54
K4=0.61
K5=1.00
Average K = (0.54+1.00+0.56+0.61+1.00)/5
=0.738

6. CONCLUSION AND FUTURE WORK

In this paper firstly we are using Vector Space Model to filter out the relevant documents which should take part in the multidocument summarization then we have used Markov clustering Principle to create cluster of similar sentences which are likely to figure in the summarization. For this reason the accuracy rate of our system as measured by the human experts is better and we are getting an average kappa measure of 0.738. Since the summarization follows the extraction method, when it extracts the important sentences it might happen that one sentence contains a proper noun and the next sentence contains a pronoun as a reference of the proper noun. In that case, if the summary considers the second sentence without considering the first one, then it does not give its proper meaning. It is a big issue in automatic text summarization. We are working to resolve this type of anaphoric problems in text summarization.

7. ACKNOWLEDGEMENT

Weare much indebted to the Department of information Technology (DIT), Ministry of Communication and Information Technology (MCIT), Govt. of India for this research work.

REFERENCES:

[1] D. Marcus: “From discourse structure to text summaries” in proceedings of the ACL/EACL ’97 Workshop on Intelligent Scalable Text Summarization, pp 82-88, Madrid, Spain.

ISSN: 1992-8645www.jatit.org E-ISSN: 1817-3195

[2] Dona Tatar , Emma Tamaianu-Morita, Andreea Mihis and Dana Lupsa: "Summarization by Logic segmentation and Text Entailment" Advances in Natural Language Processing in Computing Science 33, 2008,pp. 15-26.

[3] Jain, A.K., Dubes, R.C.: Algorithms for Clustering Data. Prentice Hall (1988)

[4] Van Dongen, S. (2000) Graph Clustering by Flow Simulation. PhD Thesis, University of Utrecht, the Netherlands.

[5] Dragomir R. Radev, Hongyan Jing, Malgorzata stys, Daniel Tam, 2004. "Centroid-based summarization of multiple documents" Information Processing and Management. 40, 919-938

[6] GunesErkan, Dragomir R. Radev. "LexRank: Graph-based Lexical Centrality as Salience in Text Summarization" Journal of Artificial Intelligence Research 22 (2004) 457-479

[7] Rada Mihalcea, Courtney Corley and Carlo Strapparava. "Corpus-based and Knowledge-based Measures of Text Semantic Similarity" American Association for ArtificialIntelligence(www.aaai.org) (2006)775-780

[8] Junsheng Zhang, Yunchuan Sun, Huilin Wang, Yanqing He."Calculating Statistical Similarity between Sentences" Journal of Convergence Information Technology, Volume 6, Number 2. February 2011, 22-34

[9] Regina Barzilay, Kathleen R. McKeown, Michael Elhadad. "Information Fusion in the Context of Multi-Document Summarization", Proceedings of the 37th Annual Meeting of the ACL 1999, 550-557

[10] Jade Goldstein, Vibhu Mittal, Jaime Carbonell, Mark Kantrowitz. "Multi-Document Summarization by Sentence Extraction", in proceedings of the 2000 NAACL-ANLP Workshop on Automatoc summarization – Volume 4 pages 40-48

[11] R.C. Balabantaray, D.K. Sahoo, B. Sahoo, M.Swain, " Text Summarization using Term Weights" IJCA volume 38-Number 1, JANUARY-2012,Pages 10-14.