Multi-Document Summarization Using Sentence Clustering

Virendra Kumar Gupta
Samsung India Software Operation
Bangalore, India
virendragupta30@gmail.com

Tanveer J. Siddiqui
Dept. of Electronics & Communication
University of Allahabad
Allahabad, India
tanveerjk@yahoo.com

***Abstract*—This paper presents an approach to query focused multi document summarization by combining single document summary using sentence clustering. Both syntactic and semantic similarity between sentences is used for clustering. Single document summary is generated using document feature, sentence reference index feature, location feature and concept similarity feature. Sentences from single document summaries are clustered and top most sentences from each cluster are used for creating multi-document summary. We observed an average F-measure of 0.33774 on DUC 2002 multi-document dataset, which is comparable to three best performing systems reported on the same dataset.**

***Index Terms*—Multi document summarization, sentence clustering method, feature extraction, DUC-2002.**

I. Introduction

With the increasing amount of online information, it becomes extremely difficult to find relevant information to users. Information retrieval systems usually return a large amount of documents listed in the order of estimated relevance. It is not possible for users to read each document in order to find useful ones. Automatic text summarization systems helps in this task by providing a quick summary of the information contained in the document. Quite often, a person might be interested in information contained in various documents on a particular topic. An ideal summary in these situations will be one that does not contain repeated information and includes unique information from multiple documents on that topic. Single document summary will be of little use in such cases. Multi document summarization (MDS) can help. MDS is the process of filtering important information from a set of documents to produce a condensed version for particular users and application. It can be viewed as an extension of single document summarization. However, issues like redundancy, coverage, temporal relatedness, compression ratio, etc., are more prominent in MDS and bring additional complexity in it [1].

Automatic text summarization methods usually consider sentence as the basic unit. These methods cluster documents, paragraphs or sentences across documents and extract important sentences from each cluster to create multi-document summary. In order to create coherent summaries lexical chains have been used to extract related sentences [2-3]. Some systems, e.g., SUMMONS [4], GISTEXTER [5] took a template-based approach for summary generation. Instead of extracting sentences, these systems use information extraction techniques to extract pieces of information and fill them into pre-defined templates to create summary. Zhao et al. [6] proposed a graph-based algorithm for query-focused multi-document summarization. Summarization systems using deep semantic approaches have also been proposed [9]. But text summarization using extractive summary is more common. This paper presents and evaluates a multi-document summarization method that builds on singe document summarization and semantic based sentence clustering.

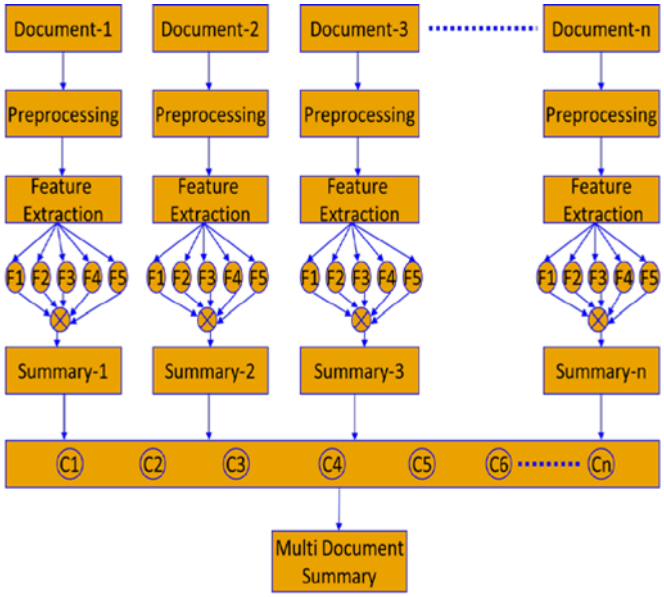

Fig. 1.Multi-document summary generation

II. Our Approach

We are using sentence clustering approach to generate multi-document summary. In this approach, single document summaries are combined using sentence clustering method to generate multi document summary.Fig.1 depicts our approach for multi-document text summarization. As shown in the figure, each document is first pre-processed and then features are extracted based on which summary is created. The sentences appearing in the individual summaries are clustered. Sentences from each cluster are extracted to create a multi-document summary. The extracted sentences are arranged according to their position in the original document. For clustering, semantic and syntactic similarity between

978-1-4673-4369-5/12/$31.00 ©2012 IEEE

sentences is used. The semantic similarity between words is combined to get semantic similarity between sentences.

The steps in multi-documents summarization are:

1. Preprocessing

The actions performed in this step are noise removal, tokenization, stemming, frequency computation, and sentence splitting.

Noise removal is concerned with removing header, footer, etc., from the document.

Tokenization breaks the text into separate lexical. Words are separated by white space, comma, dash, dot, etc.

The stop words are function words like is, am, are, a, an, the, etc. These words are not important from the point of view of summary generation and hence not considered for scoring a sentence.

Stemming is used to reduce morphological variants of a word to stem. We are using Porter Stemmer [7] in this work.

The frequency of words is computed and normalized by dividing it by the maximum frequency of any word in the document.

Sentence splitting decides the end of a sentence. The sentence splitting using end markers (. ? !) do not work in many cases. For example words like, e.g., i.e., 4.5, Mr., Dr., etc., results in false sentence boundary identification. In order to solve this problem regular expressions and simple heuristics are being used.

2. Feature extraction

In this step, we extract document feature, location feature, sentence reference index feature and concept similarity feature.

Document feature: The weight of a sentence within the document is called document feature. The sentence weight is the sum of the weights of content words appearing in it.

$$DF= w1+w2+\ldots\ldots\ldots\ldots\ldots wn \quad (1)$$

Where, DF is the document feature and wi is the normalized weight of the ith word present in the sentence. Only words having normalized frequency greater than or equal to 0.3 (Normalized frequency >=0.3) are being considered.

Location feature: The location feature gives high weight to the top and bottom sentences and low weight to the middle sentences. The underlying assumption is that in top and bottom sentences are more important as compared to middle sentence.

Sentence reference index (SRI) feature: This gives more weight to a sentence that precedes a sentence containing pronominal reference. In order to assign weight to a sentence using SRI feature a list of pronouns is maintained. If a sentence contains a pronoun then the weight of the preceding sentence is increased [11].

Concept similarity feature: The concept similarity of a sentence is the number of synsets of query words matching with words in the sentence. The set of synsets obtained from WordNet[13] is used to assign concept similarity weight to a sentence. For example, WordNet lists following concepts as synsets of the word "cancer":

Cancer: cancer, malignant neoplastic disease, Crab, Cancer the Crab, Crab Cancer, genus Cancer.

3. Single document summary generation

The sentence weight is calculated by combining the individual features using the following expression:

$$SW = v* DF +w* LF +x*SRI +y*CS \quad (2)$$

Where SW represents sentence weight, DF represents document feature, LF represents location feature, SRI represents sentence reference index feature, CS represents concept similarity feature and v, w, x, y are constant. The constants are set experimentally to v=0.5, w = 0.2, x = 0.2 and y = 0.1

The sentence weights are normalized as follows:

$$\text{Normalized Weight}= \frac{\text{Weight of each sentence present in a document}}{\text{Maximum weight of any sentence present in the document}} \quad (3)$$

The sentences are ranked using normalized weight. Top k sentences are extracted from document to generate single document summary. Where, k depends on the % summary.

4. Multi-document summary generation

The sentences appearing in single document summaries are clustered and then top scoring important sentences are extracted from each cluster. The sentences are arranged according to their position in the original document to generate the final multi-document summary.

Sentence Clustering: The sentences are clustered using sentence similarity. The sentence similarity is calculated using syntactic and semantic similarity measures proposed by Liu [10].

Syntactic Similarity: Liu et al. [10] used a method to calculate the syntactic similarity between two sentences using their word order. They assigned a unique index to each word which is used to create an original order vector (v_0) and a relative order vector (v_r). The index number of the first sentence is considered as the original order. The relative order vector (v_r) is created using common words in both the sentence. For example, the original and the relative word order vector for the sentences *The cat runs faster than a rat* (S1) and The rat runs faster than a cat (S2) is calculated as:

Index no. for S_1: {1, 2, 3, 4, 5, 6, 7}

Index no. for S_2: {1, 2, 3, 4, 5, 6, 7}

Original order vector v_0= {1, 2, 3, 4, 5, 6, 7}

Relative order Vector v_r= {1, 7, 3, 4, 5, 6, 2}

Liu et al. [10] used correlation coefficient between S1 andS2 to calculate syntactic similarity:

$$Sim_0(S_1, S_2) = \frac{\sum(v_0 * v_r) - \frac{\sum v_0 * \sum v_r}{k}}{\sqrt{(\sum {v_0}^2 - \frac{(\sum v_0)^2}{k})(\sum {v_r}^2 - \frac{(\sum v_r)^2}{k})}} \quad (4)$$

Where k is the number of words in sentence S1. The maximum value of syntactic similarity is 1 when the original and relative word order is same.

Semantic similarity: We used the method proposed in Li et al. [8] to calculate the semantic similarity between two sentences. First, the semantic similarity between words is calculated with the help of WordNet [8, 10]. The word similarity is combined to get the semantic similarity between sentences. WordNet organizes words in a hierarchy based on their semantics. Fig. 2 shows a part of WordNet hierarchy.

Semantic similarity between words: An edge count based method is used to calculate the semantic similarity between

words. Words are more similar if they have more common features and less different features. The shortest path length in WordNet hierarchy and the depth of the common subsume is used to find the common and different features between words.

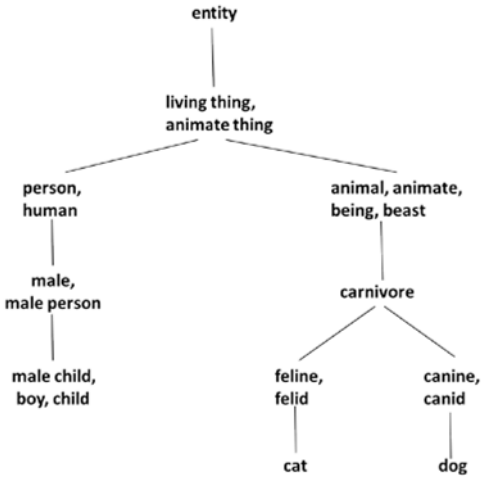

Fig. 2.Part of WordNet hierarchy

- Shortest Path Length (*l*): This gives the shortest path distance between two words. If the length of the shortest path between words is less, then words are more similar. If the length is more, then words are less similar. If the words are similar then the shortest path length between them is 0.
- Depth of Subsumer: The depth of subsumer (*d*) is the depth of the common parent between two words [8, 10]. The more general the common parents are, the less will be the semantic similarity between them.

The semantic similarity between words is defined as [10]:

$$S_w(w_1, w_2) = \frac{f(d)}{f(d)+f(l)} \quad (5)$$

Where, d is the depth of subsumer, l is the shortest path length and f is a transfer function of the form $f(x) = e^x$-1.

$S_w(w_1, w_2)$ gives a similarity value between 0 and 1. If the two words are exactly similar then the similarity between them is 1. If they are dissimilar then similarity between them is 0. When d= 0, the two words have no common parent, that is, $S_w(w_1, w_2) = 0$; When l = 0, the two words are in the same synset, and $S_w(w_1, w_2)$= 1. If bothd and l are non-zero then the similarity between word w1 and w2 is defined as follows [10]:

$$S_w(w_1, w_2) = \frac{e^{\alpha d}-1}{e^{\alpha d}+e^{\beta l}-2} (0 < \alpha, \beta \leq 1) \quad (6)$$

Where α andβ are smoothing factors.

- Information Content: Information content is a measure of information contained in a word and is computed as:

$$I(w) = -\frac{\log p(w)}{\log(N+1)} \quad (7)$$

British National Corpus [12] is used for calculating the frequency of words. The corpus contains 100 million words. The probability of words is calculated as:

$$p(w) = \frac{n+1}{N+1} \quad (8)$$

Where n is the frequency of the word in the corpus and N is the total number of words in the corpus.

With the help of information content and semantic similarity between words, we calculate the semantic similarity between sentences using the following formula [8].

$$Sim_s(S_1, S_2) = \frac{\sum_{w_i \in S_1} \max_{w_j \in S_2} (S_w(w_i, w_j) * I_{w_i})}{\sum_{w_i \in S_1} I_{w_i} + \sum_{w_j \in S_2} I_{w_j}} \quad (9)$$

Where $I(w)$ is the information content of the word, and $S_w(w_1, w_2)$is the semantic similarity between words.

The overall similarity between two sentences, S1 and S2 is calculated as [10]:

$$Sim_{sen} = Sim_s(S_1, S_2) * \big((1-\gamma) + \gamma * Sim_0(S_1, S_2)\big) + Sim_s(S_2, S_1) * \big((1-\gamma) + \gamma * Sim_0(S_2, S_1)\big) \quad (10)$$

Where γ is a smoothing factor.

4. Multi Document Summary: The sentences are clustered using sentence similarity. From each cluster, we extract single sentence. The extracted sentences are arranged according to their position in the original document to produce the multi document summary.

III. DATASET AND EXPERIMENTS

We have conducted two experiments to evaluate our system. First experiment evaluates single document summarization. The second experiment focuses on the evaluation of multi document summarization. The evaluation has been done using DUC 2002 dataset.

A. Dataset

DUC 2002 single document dataset contains 533 news articles about news in English language. The dataset also provides the gold summary for each of these articles. The DUC 2002 dataset for multi-document text summarization consists of 60 document sets and their gold summaries of size 10 words, 50 words, 100 words, and 200 words.

B. Evaluation Measure

The evaluation measure used is ROUGE. ROUGE, or Recall-Oriented Understudy for Gisting Evaluation, is a set of metrics and a software package used for evaluating automatic summarization and machine translation software in natural language processing. The metrics compare an automatically produced summary or translation against a reference or a set of references (human-produced) summary or translation.

C. The Experiments

a) Experiment 1

We performed the first experiment to evaluate single document summary generated by our system. We used DUC 2002, 100 words single document gold summaries as baseline in this experiment. Our system is well suited for query focused document summarization but we do not have a query-focused

data to evaluate single document summaries. Hence, for evaluation we have given zero weight to the user query. We observed an average F-measure of 0.46768 over 533 documents given in DUC 2002 dataset. Table 1 shows the results of this experiment.

TABLE 1. AVERAGE PRECISION, RECALL AND F-MEASURE FOR SINGLE DOCUMENT SUMMARY

Average Recall	0.45947
Average Precision	0.47989
Average F-Measure	0.46768

b) Experiment 2

In experiment 2, we evaluate multi-document summarization using sentence clustering on DUC 2002 multi-document summarization dataset. We cluster the sentences based on sentence similarity. A threshold of 0.45 is used for sentence clustering. We obtain threshold through empirical investigation. Table 2 shows the results of empirical investigation for setting the threshold value. As shown in table 2, the highest F-measure corresponds to a threshold of 0.45. If the sentence similarity between two sentences is above this threshold then it is assigned the same cluster. Fig. 3 shows F-measures for different threshold.

TABLE 2.PERFORMANCE OF MULTI-DOCUMENT SUMMARY FOR DIFFERENT THRESHOLD

Threshold 0.25	
Average Recall	0.26882
Average Precision	0.31612
Average F-Measure	0.28746
Threshold 0.30	
Average Recall	0.30815
Average Precision	0.32964
Average F-Measure	0.31834
Threshold 0.35	
Average Recall	0.31697
Average Precision	0.32935
Average F-Measure	0.32287
Threshold 0.40	
Average Recall	0.32729
Average Precision	0.33917
Average F-Measure	0.33309
Threshold 0.45	
Average Recall	0.33358
Average Precision	0.34221
Average F-Measure	0.33774
Threshold 0.50	
Average Recall	0.32308
Average	0.33021
Average F-Measure	0.32774

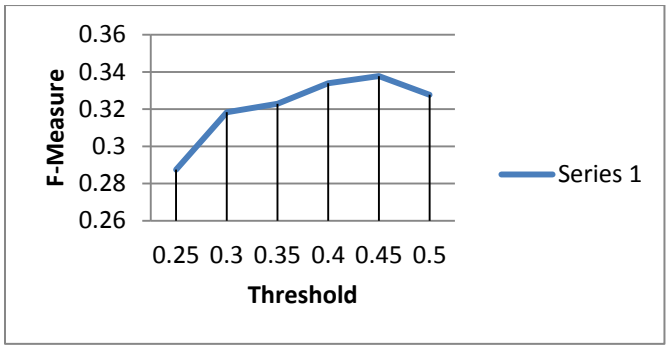

Fig 3.F-Measure vs. Threshold

IV. RESULTS AND DISCUSSIONS

As shown in table 1, we observed an average recall of 0.45947, an average precision of 0.47989 and F-measure of 0.46768 on DUC 2002 dataset. The best result reported on DUC 2002 dataset had a recall of 0.4804. Multi-document summarization using sentence clustering gives the recall of 0.33358, precision of 0.34221 and F-measure of 0.33774. The results obtained on multi-document summarization using sentence clustering are more promising. Table 3 shows F-measures of the five best performing multi-document summarization systems on DUC 2002 dataset. Our results are quite close to the second best performing system. This shows the potential of sentence clustering.

TABLE 3.DUC 2002 RESULTS

Top 5 Systems (DUC 2002)					
S26	S19	S29	S25	S20	Baseline
0.3578	0.3447	0.3264	0.3056	0.3047	0.2932

V. CONCLUSIONS AND FUTURE WORK

This paper presents a method for multi document summarization by combining single document summaries. The features used for generating single document summaries are sentence weight, sentence reference index feature, location feature and concept similarity feature. The single document summaries using sentence clustering method to generate multi document summary. When evaluated on DUC 2002 summarization data, the method gives results comparable to the best performing systems reported on the same dataset. The sentence similarity measure used in this paper uses semantic similarity based on shortest path length and the depth of subsumer. A number of semantic similarity measures based on these concepts exist in literature, which can be used to decide semantic similarity between sentences. The syntactic similarity used in this paper is based on word order, which can be replaced with other structural similarity measures. We would also like to evaluate our system on DUC 2005 or DUC 2006 datasets for query-based summarization.

REFERENCES

[1] D.Radev, Hongyan Jing, Malgorzata Stys and Daniel Tam,"Centroid-based summarization of multiple documents",Information Processing and Management 40 919−938, 2004.

[2] Silber, H. Gregory and Kathleen McCoy, "Efficiently computed lexical chainsas an intermediate representation forautomatic text summarization",Computational Linguistics, 28(4), 487–496, 2002.

[3] Regina Barzilay and Michael Elhadad, "Using Lexical chains for Text Summarization", ACL/EACL Workshop on Intelligent Scalable Text Summarization, pages 10-17, Madrid, 1997.

[4] K. McKeown and D. Radev, "Generating summaries ofmultiple news articles", In Proceedings of the 18th AnnualInternational ACM Conference on Research and Developmentin Information Retrieval (ACM SIGIR) (pp.74-82). Seattle, WA, 1995.

[5] S. M. Harabagiu, and F. Lacatusu, "Generating single andmulti-document summaries with GISTEXTER",DocumentUnderstanding Conferences, 2002.

[6] Lin Zhao, Lide Wu andXuanjing Huang, "Using query expansion in graph-based approach for query-focusedmulti-document summarization", Information Processing and Management 45 (2009) 35–41, 2009.

[7] The Porter Stemming Algorithm [Online].Available:http://tartarus.org/~martin/PorterStemmer/

[8] Yuhua Li, David McLean, Zuhair A. Bandar, James D. O'Shea, and Keeley Crockett, "Sentence Similarity Based on Semantic Nets and Corpus Statistics",IEEE Transactions on Knowledge and Data Engineering, Vol. 18, No. 8, August 2006, p. 1138-1150.

[9] K. McKeown and D.Radev, "Generating summaries of multiple news articles".In Proceedings, 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 1995, pp. 74–82.

[10] Xiao-Ying Liu, Yi-Ming Zhou and Ruo-Shi Zheng,"Measuring semantic similarity within sentences", Proceedings of the Seventh International Conference on Machine Learning and Cybernetics, Kunming, 12-15 July 2008, pp. 2558-2552.

[11] Alkesh Patel, Tanveer J Siddiqui and U S Tiwary, "A language independent approach to Multi-lingual Text Summarization", In the proceedings of RIAO 2007, May 30 to June 1, 2007. Available at: http://riao.free.fr/papers/30.pdf

[12] British National Corpus [Online]. Available: http://www.natcorp.ox.ac.uk/

[13] George A. Miller, "WordNet: A Lexical Database for English", COMMUNICATIONS OF THE ACM November 1995/Vol. 38, No. 11, pp. 39-41.