Clustering approach for Automatic Text Summarization

[1]Jaya Kapoor, [2]Kailas K.Devadkar

[1]Alamuri Ratnamala Institute of Engineering and Technology
[2]Sardar Patel Institute of Technology, Andheri

Abstract: Automatic text summarization is the technology which plays an important role in information retrieval and text classification and also helps in providing a vital solution to information overload problem. It is the process of reducing the size of text while preserving its information content. This paper presents a clustering approach for automatic text summarization tool. The method explains the summarization technique in three steps: first clustering of the sentences is based on semantic distance between sentences in the document and using multi-feature combination on each cluster it calculates the accumulative sentence similarity, at last chooses the topic sentences by some extraction rules. Our method Clustering Approach for Automatic Text Summarization (CAATS) is experimented on the predesigned dataset to show that this technique provides better efficiency compared to other summarization method.

Keywords- text summarization, similarity measure, sentences Clustering, sentence extractive technique.

I. INTRODUCTION

The tremendous growth of World Wide Web and on-line textual compilation makes a large volume of information available and accessible to users. The concept of information overload either leads to the wastage of significant time in browsing all the information or else useful and important information are missed out. The technology of automatic text summarization is evolving and maturing and may provide a solution to the problem of information overload. Text summarization is the method of automatically creating a compressed version i.e. a summary of a given text that gives useful and meaningful information to the users, and multi-document summarization is to give the outcome a summary delivering the majority of information content from a set of documents about an explicit or implicit main topic [14].

Text summarization is a complicated task which ideally would involve wide range of natural language processing capacities. In order to simplify the summarization issues, current research is focused on extractive-summary based generation. These sentence based extractive summarization techniques are very commonly used in automatic text summarization to produce extractive summaries. Traditional method of summarization uses the sentence features to evaluate the importance of sentences of a document. This paper presents a sentence based similarity computing method based on the three features of the sentences, firstly, analyzing of the word form feature, the word-order feature and the semantic feature, using weight to give details of the contribution of each feature of the sentence, describes the sentence similarity more preciously. Determinates the total number of the clusters, uses the K-means method to cluster the sentences within the document, and extracts the topic sentences to generate the extractive summary for the document. Experiments show that our method is outperforms than other summarization methods using the dataset 1 and dataset 2 evaluation metrics. The rest of the paper is compiled as follows: Section 2 introduces related works. About clustering approach and techniques formulas methods for automatic text summarization are presented in Section 3. Section 4 presents evaluation result dataset along with algorithm and comparison. The last section gives the conclusions.

II. RELATED WORK

Earlier, extractive summarizers have been mostly based on scoring sentences in the source document. The most common and recent text summarization techniques use either statistical approaches, for example [15], [13], [3], [9]; or linguistic techniques, for example[6], [10], [8]; or some kind of a linear combination of these: [4],[7] and [2]. The algorithm which is present in this paper markedly different from each of these and tries to capture the semantic distance of the sentences within the document. We analyzed that none of the above approaches to text summarization selects or uses sentences based on the semantic content of the sentence and the relative importance of the content to the semantic of the text. This algorithm is based on identifying semantic relations among sentences and is for automatic text summarization unlike almost all previous ones.

III. SENTENCE CLUSTERING AND SUMMARIZATION

3.1 Similarity measure between sentences

Definition 1: Word Form Similarity

It is used to describe the form similarity between 2 sentences, which is measured by the number of same words within two sentences. It should be getting rid of the stop words in the calculation. Here S1 and S2 are two sentences, and the word form similarity within these sentences is computed by the following method or following formula.

$$Sim_1\left(S_1,S_2\right)=2*\left(Same\ Word\left(S_1,S_2\right)/\left(Len\left(S_1\right)+Len\left(S_2\right)\right)\right) \qquad (1)$$

Here the Same Word (S1, S2) is the number of the same words in two sentences, Len(S) is the word number in the sentence S.

Definition 2: Word Order Similarity

The word-order similarity is mostly used to describe the sequence in similarity between the two sentences. The Chinese sentence can be given by many kinds of ways and methods, the different sequence of the words stand for different meanings and style. Here we illustrate the sentence as three vectors as follows:

V1={d11,d12,…,d1n1},

V2={d21,d22,…,d2n2},

V3={d31,d32,…,d3n3}.

Here, weight of document d1i in the vector V1 is the tf-idf value of the words; the weight of document d2i in vector V2 is the bi-gram whether it occurs in the sentence (0 is for non-occurring, 1 stands for occurring); the weight d3i in vector V3 is the tri-gram whether it exists within the sentence:

$$Sim_2\left(S_1,S_2\right)=\lambda_1*Cos\left(V_{11},V_{21}\right)+\lambda_2*Cos\left(V_{12},V_{22}\right)+\lambda_3*Cos\left(V_{13},V_{23}\right) \qquad (2)$$

Here λ1+λ2+λ3=1. λi stands for the ratio of each part.

Definition 3: Semantic Similarity Between words

It is widely used to describe the semantic similarity among sentences. Here the word semantic similarity computing [11], Based on semantic similarity among words, we define Word Sentence Similarity (WSSim) by the maximum similarity among the word w and word within the sentence S. Therefore, we project WSSim(w,S) using following formula:

$$WSSim\left(w,S\right)=max\{Sim\left(w,\ W_i\right)|W_i\in S,\ \text{where w and Wi are words}\} \qquad (3)$$

Here the Sim(w,Wi) is the word similarity among w and Wi. With WSSim(w,S), we define the sentence similarity as follows:

$$Sim_3(S_1,S_2)=\frac{\sum_{w_i e S_1}WSSim(w_i,S_2)+\sum_{w_j e S_2}WSSim(w_j,S_1)}{\left|S_1\right|+\left|S_2\right|} \qquad (4)$$

Here S_1 and S_2 are sentences; |S| is the number in sentence S.

Definition 4: Sentence Similarity

The sentence similarities are commonly described as a number between zero and one, the zero stands for non-similar, and the one stands for completely similar. The larger the number is, the more sentences are identical. The sentence similarity among S1 and S2 is defined as follows:

$$Sim\left(S_1,S_2\right)=\lambda_1*Sim_1\left(S_1,S_2\right)+\lambda_2*Sim_2\left(S_1,S_2\right)+\lambda_3*Sim3\left(S_1,S_2\right) \qquad (5)$$

Here λ_1, λ_2, λ_3 is the constant: $\lambda_1+\lambda_2\ +\lambda_3=1$.

Here in this paper, $\lambda_1=0.2$, $\lambda_2=0.1$, $\lambda_3=0.7$.

3.2 Estimating number of clusters

Determination of the optimal number of cluster within sentence in a text document is a difficult issue and depends on the compression ratio of summary and chosen similarity measure, and simultaneously on the document topics. For clustering of sentences, user can't predict the latent topic number in the document, so it's impossible to offer k effectively. The approach that we used to determine the optimal number of clusters (the number of topics in a document) is based on the distribution of words in the sentences:

$$k = n\frac{|D|}{\sum_{i=1}^{n}|S_i|} = n\frac{\left|\bigcup_{i=1}^{n} S_i\right|}{\sum_{i=1}^{n}|S_i|} \qquad (6)$$

Where |D| is the number of terms in document D, |Ai| is number of terms in the sentence Ai, n is number of sentences in document D. Here we analyze the property of this estimation by two extreme cases.

(1) The document has n sentences which have the same set of terms. Therefore, the set of terms within document coincides with the set of terms of each sentence:

D= (t1, t2, …, tm)=Ai=A. From the definition (6) follows that

$$k = n\frac{\left|\bigcup_{i=1}^{n} S_i\right|}{\sum_{i=1}^{n}|S_i|} = n\frac{\left|\bigcup_{i=1}^{n} S\right|}{\sum_{i=1}^{n}|S|} = n\frac{|S|}{\sum_{i=1}^{n}|S|} = 1 \qquad (7)$$

(2) The document has n sentence which do not have any term in common, that is, Si∩Sj=Φ for i≠j. This means that each term belonging to

$$|D| = \left|\bigcup_{i=1}^{n} S_i\right| = \sum_{i=1}^{n}|S_i| \qquad (8)$$

belongs to only one sentence $S_{i.}$ Therefore

From which follows that k=n.

3.3 Extraction of Topic and Sentences

We assume the sentences clusters are: D = {C1, C2, … , Ck}, based on results shown by section C. Firstly, we determine the central sentence μi of each cluster based on the accumulative similarity between the sentence Si and other sentences, then we calculates the similarity between the sentence Si and the central sentence μi. Assume that the similarity of central sentence μi as 1, sorts the sentences based on its similarity weight, and chooses the high weight sentences as the topic sentences. At the same time, considering the recall rate of the text summarization, the text summary should include every cluster sentences according to the principle of priority extract clusters in the process of extracting sentences.

IV. EXPERIMENTS AND RESULTS

Here, in this section, we conduct experiments to evaluate the performance of the automatic text summarization system based on sentences clustering.

4.1 Runs and Evaluation Results

For evaluation the performance of our automatic tool summarization tool called as CAATS, we conduct the experiments on the document dataset, compares our method with K-mean [10] methods.

4.1.1 Algorithm

The pseudo–code for Clustering Approach for Automatic Text Summarization (CAATS)is:

```
1. Construct the normalization mapping.
2. Initialize the critical score to zero.
3. Populate and sort the sorted table with all states for all words using the normalized scores.
4. Remove the most probable state and insert into the indexed table.
5. While the sorted table contains uncombined states:
6. Remove the most probable from the sorted table as the pivot.
7. Return if the pivot is a terminal state.
8. Combine pivot with all adjacent states in the indexed table that don't fall below the critical score.
9. for every state that has been created:
10Adjust the critical score if the produced state is a terminal state and the score is better.
11. Insert the created states into the sorted table with the normalized score.
12. Insert the pivot into the indexed table.
13. Return failure.
```

4.1.1 K- Mean Method:

K-means is an unsupervised and unrestricted learning algorithm which solves the notable clustering problem. The method divides a given data set through a particular number of clusters (let say, k clusters) fixed a priori. The main idea is to define k centroids, one for every single cluster. The centroids are chosen to place them as much as possible far away distance from each other. The next step is to take each point belonging to a given dataset and accomplice it to the nearest centroid. When all points have been classified and organized, we re-calculate k with new centroid as new centers of the clusters resulting from the initial step. After we have these k new and unused centroid, a new association is created among the same data set points and the nearest new centroid. The k centroid develops their location in each step until no more changes or development occurs. Although the K-means algorithm will always ends, it does not necessarily find the most optimal and ideal configuration, corresponding to the global objective function minimum. The algorithm is also extremely sensitive to the initial randomly selected cluster centers.

$$J = \sum_{j=1}^{k} \sum_{i=1}^{x} \left\| x_i^{(j)} - c_j \right\|^2 \qquad (9)$$

Where $\left\| x_i^{(j)} - c_j \right\|^2$ is a chosen distance measure between a data point $x_i^{(j)}$ and cluster center c_j is an indicator of the distance of n data points from their respective cluster centers.

4.2.2 K- Mean Algorithm:

1. Place K points into the space illustrated by the objects that are being clustered. These points represent initial (start) group centroids.

2. Allocate each object to the group that has the closest or nearest centroid.

3. When all objects have been allocated, recalculate the positions of the K centroids.

4. Repeat Steps 2 and 3 until the centroids no longer move. This produces a separation of the objects into groups from which the metric to be minimized

4.3 Comparison:

4.3.1 Evaluation metrics

Evaluation of summaries and automatic text summarization systems is not a straight-forward process. The topical similarities between two summaries can be calculated using various different ways and measures. For calculating the results we use two ways. The first one is by P which is called as precision, and second one is R known as Recall. Which are widely used in Information Retrieval from each document, the manually extracted sentences denoted by Summref are considered as the reference summary . This approach compared Summcand i.e. candidate summary with the reference summary and computes the P, R values as shown in formula (9). [12]

$$P = \frac{\left| Summ_{ref} \cap Summ_{cand} \right|}{\left| Summ_{cand} \right|} \quad R = \frac{\left| Summ_{ref} \cap Summ_{cand} \right|}{\left| Summ_{ref} \right|} \quad F_1 = \frac{2PR}{P+R} \qquad (10)$$

The second measure we use the dataset for evaluation, which was adopted for automatically summarization evaluation. It measures summary quality by counting overlapping units such as the N-gram, word sequences and word pairs among the candidate summary and the reference summary. The dataset measure compares N-grams of two summaries, and counts the number of matches. The measure is defined by formula (10):

$$ROUGE - N = \frac{\sum S \in Summ_{ref} \sum N - gram \in S \; {}^{Count} match^{(N-gram)}}{\sum S \in Summ_{ref} \sum N - gram \in S \; {}^{Count(N-gram)}} \qquad (11)$$

Table 1: Results of evaluation of word count at input and output using our approach on various sample text

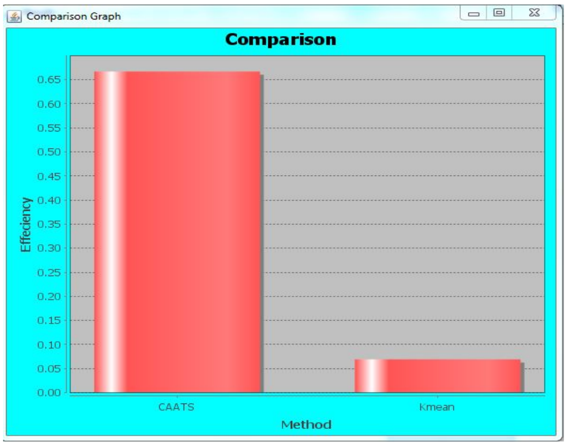
Dataset	1	2	3	4	5	6	7
Number of word counts in original input textual file	178	282	168	122	230	130	1497
Number of word count in summary generated by our method-CAATS	54	94	66	53	97	54	177

As shown in Table 2, the values of dataset-1, dataset-2 of our method-CAATS is better than K-means clustering methods.

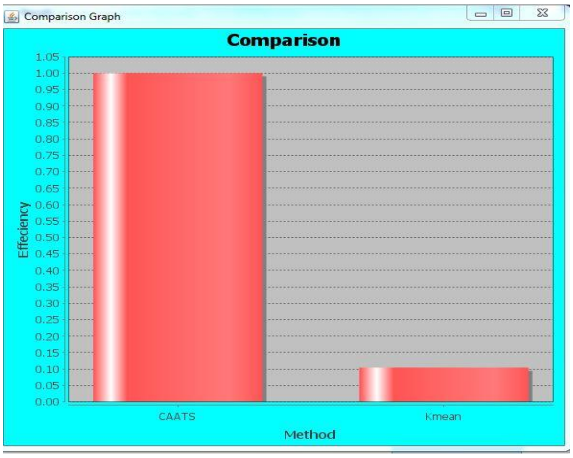
TABLE 2Efficiency values of evaluation metrics for summarization methods on various sample dataset

Methods/ Datasets	1	2	3	4	5	6	7
K-mean Efficiency	0.07	0.10	0.08	0.27	0.051	0.062	0.70
CAATS Efficiency	0.65	1.00	1.00	0.70	0.325	0.45	1.00

4.3.2 Graphical Comparison



Graphical Comparison of dataset-1 using CAATS v/s k-mean



Graphical Comparison of dataset-2 using CAATS v/s k-mean

V. CONCLUSION

We have presented the approach to automatic text summarization based on the sentences clustering and extraction. Our approach consists of three steps. First clusters the sentences in document, and then on each cluster calculates the accumulative sentence similarity based on the multi-features combination, at last chooses the topic sentences by the rules. When comparing our method known as CAATS with other existing summarization methods on datasets, we found that our method can improve the summarization results significantly using the evaluation metrics of dataset-1, dataset-2. It provides a sentence similarity computing method based on the three features of the sentences, on the base of analyzing of the word form feature, the word order feature and the semantic feature, using the weight to describe the contribution of each feature of the sentence, describes the sentence similarity more preciously. It has given a method of determinate the number of the sentence clusters. It gives an approach of text summarization based on the sentences clustering.

REFERENCES

[1] Dong Zhen-dong. How Net [OL]. http://www.keenage.com

[2] Barzilay, Elhadad, 1997. Using lexical chains for text summarization. Proceedings of the ACL'97/EACL'97 Workshop on Intelligent Scalable Text Summarization.

[3] Berger and Mittal, 2000. Query-relevant summarization using faqs. Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics.

[4] Goldstein, Kantrowitz, Mittal, and Carbonell, 1999. Summarization text documents: Sentence selection and evaluation metrics. Proceedings SIGIR.

[5] Uplavikar Nitish Milind, Wakhare Sanket Shantilalsa, Prof. Dr. R.S. Prasad, 2012"International Journal of Advances in Computing and Information Researche ISSN: 2277-4068, Volume 1– No.2,"

[6] Klavans, Shaw, 1995. Lexical semantics in summarization. Proceedings of the First Annual Workshop of the IFIP working Group for Natural Language Processing and Knowledge Representation.

[7] Mani, 2002. Automatic summarization. A tutorial presented at ICON.

[8] Nakao, 2000. An algorithm for one-page summarization for a long text based on thematic hierarchy detection. Proceedings of the 38thAnnual Meeting of the Association for Computational Linguistics.

[9] Nomoto, Matsumoto, 2001. A new approach to unsupervised text summarization.Proceedings of the 24th ACM SIGIR.

[10] AartiPatil, Komal Pharande, Dipali Nale, Roshani Agrawal, January 2015."Automatic Text Summarization." Volume 109-No.17, International Journal of computer Applications

[11] Ramiz M. Aliguliyev, 2008. A new sentence similarity measure and sentence based extractive technique for automatic text summarization. Expert System with Applications.

[12] Shen,D., Sun,J.-T.,Li,H., Yang, Q.,& Chen, Z. 2007. Document summarization using conditional random fields. In Proceedings of the 20thinternational joint conference on artificial intelligence (IJCAI 2007), January 6-12 (pp. 2862-2867) Hyderabad, India

[13] Strzalkowski, Wise, Wang, 1998. A robust practical text summarization system. Proceedings of the Fifteenth National Conference on A1.

[14] Wan, X. 2008. Using only cross-document relationships for both generic and topic-focused multi-document summarizations. Information Retrieval.

[15] Zechner, 1996. Fast generation of abstracts from general domain text corpora by extracting relevant sentences. COLING.

[16] Zhang Qi, Huang Xuan-jing, Wu Li-de, 2004. A new method for calculating similarity between sentences and application on automatic text summarization. Journal of Chinese information processing.

◈◈◈