Text Summarization Project

Erol Özkan
Department of Computer Engineering, Hacettepe University
erolozkan@outlook.com

İbrahim Ardıç
Department of Computer Engineering, Hacettepe University
ardicib@gmail.com

***Abstract*- An automatic summarization is needed to help users extract the most important pieces of information from the vast amount of text digitized into electronic form every day. In this paper we propose to make use of different similarity measures at the same time for text summarization task. We calculate different similarity measures based on sentiment and continuous vector space model representations. Then we use these measures along with classical TfxIdf similarity. Continuous vector representations make our system semantically aware of representations of sentences. And sentiment gives a bias to sentences that is loaded with sentiment. We evaluate our results with Rouge package with different parameters and compositions.**

Keywords—Summarization, Word Embeddings, Continuous Vector Space Models, Text Rank

I. Introduction

Due to the great amount of information we are provided with and thanks to the development of Internet technologies, it is nearly impossible for a single human being to study, analyze and digest this much of data. For this reason, needs of producing summaries have become more and more widespread.

In this paper we introduce an application that also takes into account continuous vector representations and sentiment analysis similarity measures to the problem domain of multi-document summarization. We evaluate our results with Rouge package, using different compositions of similarity measures. Also, we implement a page-rank algorithm as well as a clustering algorithm for sentence selection part. Our experiments strongly indicate the benefits of using continuous word vector representations and sentiment information for the text summarization tasks.

The remainder of the paper is organized as follows. In Section II and Section III, we review the text summarization task and explain the related work. Then our approach is detailed in Section IV. In Section V and Section VII, evaluation strategies and experimental results are presented. Finally, we conclude our work in Section VII.

II. Summarization Problem

In text summarization problem, our aim is to find a summary that both maximizes the coverage of the input text and diversity of the sentences. This objective function can be formulated as follows:

$$\mathcal{F}(S) = \mathcal{L}(S) + \lambda\mathcal{R}(S)$$

Here; S represents the summary, L(S) is the coverage of the input text and R(S) is a diversity reward function. The λ is a trade-off coefficient that allows us to define the importance of coverage versus diversity of the summary [1]. For text summarization task we want to maximize this F(S) function.

Summarization Types

There are various types of automatic summarization techniques. Firstly, there are extractive and abstractive summaries. In extractive summaries, summaries are created by choosing representative sentences from a sentence set while in abstractive summaries which is generally considered a more difficult problem; new sentences are created by using some techniques from the natural language processing field.

Also there are single-document and multi-document summarization. In single-document text summarization there is only a single document, extracting the most important sentences from it is the main purpose. In multi-document text summarization, there are multiple documents and extracting a summary from all of them is the main purpose.

Summarization techniques can also be distinguished between generic summaries and user-focused summaries (a.k.a query-driven). The generic summaries serves as surrogate of the original text as they may try to represent all relevant features of a source text. They are intended to cover the topics in the source text. User-focused summaries on the other hand rely on a specification of a user information need, such as a topic or a query.

For this paper our focus is only on the extractive, generic and multi-document summarization, that is, we have a set of documents D and we choose the sentences from this sentence set based on their similarity score that we assign.

III. Related Work

Radev et al. [2], [3] was the first, who proposed a method for centroid based multi-document summarization. To do this, the number of cluster centroids are obtained and in each clusters, the similar sentences are identified by cosine similarity measure. Centroids are the top ranking TfxIdf of each cluster and once sentences are grouped, the selection of sentences process is taken into account by choosing a subset of the sentences from each cluster [1]. This summarization architecture can be shown in the Figure 1.

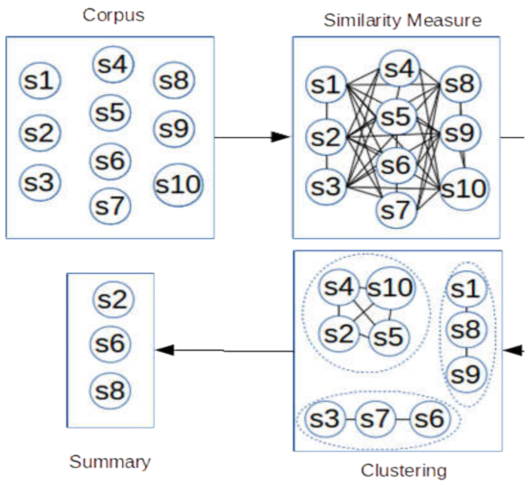

Figure 1 : A Generalized Architecture for Cluster based Summarization [5]

However clustering approach gives lots of benefits, optimal cluster numbers are need to be considered to process summarization task more effective. To find optimal clusters Xia et al. [4] determines the weights of sentences and terms based on the sentence-term co-occurrence matrix. Then, the diversity and redundancy information is extracted from this sentence-term matrix and highest ranked sentences in each clusters are selected to form summaries [5].

Cluster based methods are useful and successful in extractive summarization to find the diversity and redundancy of sentences, but in multi-document summarization task, the selected summary sentences could not form a meaningful summary because of clustering method task that the sentences are selected to the similarity measure to its centroid according to the frequency of terms [5].

Another approaches of summarization task proposed by Erkan and Radev as LexRank [3] and Mihalcea and Tarau as TextRank [6] are simply used graph-based method to find salient sentences in a document or set of documents. The graph is constructed by the sentences similarities in a set of documents then important sentences are selected via random walk on the graph to include in the summary [7]. Unlike clustering approach, this approach concerns with the graph-based algorithms HITS algorithm [8] and Google's PageRank algorithm [9].

Bonzanini et. al [10] issued an algorithm that based on the sentence removal iteratively. It starts with the set of all sentences and chooses to remove unimportant ones. They've worked with the dataset Opinosis [11] and got a good results with this approach.

IV. Method

Discounting the preprocessing part, our text extractive summarization system is composed of two main components, the similarity measures used to compare sentences and the summarization framework, which is used to select sentences using these similarity measures. Next four subsection addresses these components with their responsibility.

A. *Preprocessing*

In the preprocessing phase, to convert the original text into the text that will be used as an input parameter to our summarization system; firstly we remove stop words that do not have any specific data and do not show any value just before calculating similarity measures. We then, convert all letters into lowercase. Furthermore, in some cases where the derived word is not found in trained continuous vector space model, we stem the words back to their original form. Also, we follow the below schedule for anaphors in our input text to further preprocess our input text.

B. *Anaphora Replacement*

Anaphora is co-reference of one expression with its antecedent. The antecedent provides the information necessary for the expression's interpretation. In other words, Anaphora is an expression "referring" back to the antecedent. Some examples of anaphora in sentences where one word refers to another are shown in the sentences in Table I. Here the anaphora is underlined and substituted word shown in parentheses.

TABLE I
ANAPHORA EXAMPLES

She dropped the glass and it broke into pieces. (the glass)
The party was over and that upset everybody. (The party was over)
The child wanted a pony but her parents didn't buy one for her. (pony)
If my son moves to Florida, I will do that as well. (move to Florida)
The teacher was disappointed and so were his students. (disappointed)
Fred asked Ginger to pass him the potatoes. (Fred)
I know it and she does, too. (knows it)
Sue needed the glue and asked me to finish with it. (glue)
The dog really wanted the bone but Sam threw it away. (bone)

It is noted by some papers that the anaphors actually contain non-basic information and if a sentence contains an anaphor, its contents are covered by other related sentences. That's why it should either not be included in the final summary or its content should be changed. For this reasons, in our work, we try to replace pronouns that have the same expression as explained above. Table II shows some of detected anaphors by our system.

TABLE II
DETECTED ANAPHORS BY OUR SYSTEM

"It looks like the eye is going to move lengthwise across that island, and they're going to bear the full brunt of this powerful hurricane," Sheets said.(is pronoun to replace they're)
"People were running around in the main lobby of our hotel (on Grand Cayman) like chickens with their heads cut off," said one vacationer who was returning home to California through Miami.(is pronoun to replace their)
"It's moving at about 17 mph to the west and normally hurricanes take a northward turn after they pass central Cuba".(is pronoun to replace they)
Pemex officials however said all their vessels were secure.(is pronoun to replace their)
"The wind would blow them away," said an army official at city hall who declined to give his name.(is pronoun to replace his)

C. *Similarity Measures*

In the most of the text summarization studies, the optimization objective is a function scoring a candidate summary by coverage and diversity, expressed using cosine similarity between sentences represented as bag-of-terms vectors. In this work, we combine different similarity measures by simply multiplying them.

TfxIdf Similarity

Most extractive similarity system uses TfxIdf [12] sentence similarity that calculates the cosine angle between the vectors of sentences. Each sentence is represented by the words w = {w_1, … , w_n} where n is the vocabulary size. Vocabulary is constructed from the all distinct words in corpus. $Tf_{w,d}$ represents the word occurrence in document d, and Idf_w represents the word inverse document occurrence. The similarity measure can be calculated as following equation.

$$Sim(i,j) = \frac{\sum_{w \in i} tf_{w,i} \times tf_{w,j} \times idf_w^2}{\sqrt{\sum_{w \in i} tf_{w,i}^2 \times idf_w^2}\sqrt{\sum_{w \in j} tf_{w,j}^2 \times idf_w^2}}$$

Sentiment Similarity

Some research say that negative emotion words appear at a relative higher rate in summaries written by humans [13]. In our work we make use of this sentiment information by comparing the level of sentiment in each sentence with other sentences and create a new similarity matrix.

Firstly, we create two lists of positive and negative sentiment words respectively. Then, we give each sentence a sentiment score based on the number of words found in the positive and the negative list, divided by the total number of the words in the sentence. And finally, we calculate a similarity score for positive sentiment between sentences as follows;

$$M_{\mathbf{s}_i,\mathbf{s}_j} = 1 - |positive(\mathbf{s}_i) - positive(\mathbf{s}_j)|$$

Table III shows the top five sentences that have the highest positive sentiment score in "d061j" document set.

TABLE III
TOP FIVE SENTENCES THAT HAVE THE HIGHEST POSITIVE SENTIMENT SCORE

That's something meteorologists would like to know more about. 0.33(3/9) positive words found : like, know, more
Telephone communications were affected. 0.25(1/4) positive words found : affected
It was something new. 0.25(1/4) positive words found : new
"It's certainly one of the larger systems we've seen in the Caribbean for a long time," said Hal Gerrish, forecaster at the National Hurricane Center in Coral Gables, Fla. 0.17 (5/29) positive words found : one, in, for, at, in
Most Jamaicans stayed home, boarding up windows in preparation for the hurricane. 0.16(2/12) positive words found : in, for

And similarly we calculate a similarity score for negative sentiment as follows.

$$M_{\mathbf{s}_i,\mathbf{s}_j} = 1 - |positive(\mathbf{s}_i) - positive(\mathbf{s}_j)|$$

Table IV shows the top five sentences that have the highest negative sentiment score in "d061j" document set.

TABLE IV
TOP FIVE SENTENCES THAT HAVE THE HIGHEST NEGATIVE SENTIMENT SCORE

Gilbert came to the attention of center forecasters Sept. 3 as a dry low pressure trough moving west out of Africa. 0.09(2/21) negative words found : dry, out
Shelters had little or no food, water or blankets and power was out. 0.07(1/13) negative words found : out
The first shock let up as the eye of the storm moved across the city. 0.06(1/15) negative words found : shock
That broke the 26.35 inches of the 1935 hurricane that devastated the Florida Keys . 0.06(1/15) negative words found : devastated
"The sound of the wind outside is horrible" said receptionist Pablo Torres at Cancun's Hotel Carrillos as the storm approached. 0.05(1/20) negative words found : horrible

This sentiment similar measure allows our summaries to be representative and diverse in sentiment.

Continuous Vector Space Models

Typically, In order to have a high similarity between sentences using the above measures, two sentences must have an overlap of highly scored TfxIdf words. But, in cases where a different word with same meaning is used, it is impossible to catch this overlap. For instance; although the terms The US President and Barack Obama in different sentences have the same meaning, they will not add towards the similarity of the sentences.

To capture this similarity, we make use of the continuous vector representations for measuring similarity between sentences. We use GoogleNews-vectors-negative300.bin.gz [14] pre-trained word vector model to find similarities between sentences by looking the vector distances of given sentences.

Table V shows top ranked sentences using continuous vector representations model in "d061j" document set.

TABLE V
TOP RANKED SENTENCES THAT HAVE THE HIGHEST SCORE

There were no reports of casualties. There were no immediate reports of casualties.
There were no reports of casualties. But there are no reports of injuries or damage.
The storm was approaching from the southeast with sustained winds of 75 mph gusting to 92 mph. A National Weather Service report said the hurricane was moving west at 17 mph with maximum sustained winds of 115 mph.
How it develops, we don't know. We can't do it yet.
It reaches hurricane status when sustained winds hit 74 mph. When sustained winds reach 39 mph, the system becomes a named tropical storm.
The storm was approaching from the southeast with sustained winds of 75 mph gusting to 92 mph. The Mexican National Weather Service reported winds gusting as high as 218 mph earlier Wednesday with sustained winds of 179 mph.
"They have reported maximum winds of 25 knots and gusts up to 50 knots," said Ross. The Mexican National Weather Service reported winds gusting as high as 218 mph earlier Wednesday with sustained winds of 179 mph.

V. Evaluation

We evaluate our results with Rogue package that was introduced by Lin [15] paper. Rouge is a set of metrics called recall-oriented understudy for gisting evaluation. It became a standard for automatic evaluation of summaries. It basically works by counting word overlaps between generated summaries and gold standard summaries.

Our results include some metrics that Rogue calculates such as Rogue-1, Rogue-2, Rogue-3, Rogue-L and Rogue-SU4. The first three accounts for matches in unigrams, bigrams and trigrams respectively.

The Rogue-SU4, on the other hand, accounts for matches in skip-bigrams. Skip-bigram, is any pair of words in their sentence order, allowing for arbitrary gaps. Here, it allows four words in between. It outputs the co-occurrence statistics measure that is calculated from the overlap of skip-bigrams between a candidate translation and a set of reference translations.

Another metric Rogue-L is calculated by applying the concept of longest common subsequences (LCS). The rationale in here is that the longer the LCS between two summary sentences, the more similar they are. The Rogue-L can be defined by below formula:

$$\text{ROUGE-L}(s) = \frac{(1+\beta^2)R_{\text{LCS}}P_{\text{LCS}}}{R_{\text{LCS}} + \beta^2 P_{\text{LCS}}}$$

VI. Experimental Results

Table VI shows the top 10 scores obtained from our experiments. The option set that is used for this experiment is shown at Table VII.

TABLE VI
Rouge Scores for "D061J" Document Set

No	ROUGE 1	ROUGE 2	ROUGE 3	ROUGE SU4	ROUGE L
1	**0,63068**	0,33456	0,26634	0,32781	0,6012
2	0,61576	**0,35966**	**0,31668**	**0,36093**	**0,6036**
3	0,60658	0,31013	0,235	0,30675	0,5797
4	0,59052	0,3313	0,2906	0,34076	0,5710
5	0,58343	0,29362	0,20748	0,27908	0,5570
6	0,58154	0,3117	0,2439	0,29911	0,5208
7	0,5799	0,3126	0,2461	0,2981	0,5206
8	0,5783	0,2911	0,2381	0,2993	0,5121
9	0,5734	0,2898	0,2276	0,2888	0,5518
10	0,5710	0,2371	0,1680	0,2454	0,5400

Similarity Matrixes column in Table VII shows the multiplication of the similarity measure of related cell, and when the method is clustering, the Centroid part shows the centroid determination of clustering by the given matrix and Similarity Matrixes are then used in iteration.

According to the results; clustering method using multiple similarity matrixes gives the best ROUGE scores. Clustering size is simply calculated as the 20 percent of the number of sentences in the document set. Using Word Embedding (Word2Vec) as similarity matrix without anaphora resolution and centroid identification with only TfxIdf measure outperforms on all ROUGE scores except unigram score. Since Word2Vec has semantic measurement system, it does not require sentiment matrix or TfxIdf matrix individually. Also, it does not require even anaphora replacement. However, it still gives high results when used with these similarity measures together.

Furthermore, sentiment similarity matrix is meaningful only when used together with the TfxIdf similarity matrix. That means TfxIdf measure only statistical approach based on only given corpus, and adding sentiment analysis to this information might give better understanding of the similarity distance. On the other hand, since word2vec matrix is formed using a huge pre-trained corpus, it gives better distance measurement within the sentences and more meaningful in the clustering method.

PageRank algorithm gives good results using TfxIdf and Sentiment similarity matrixes with anaphora resolution. TfxIdf and sentiment similarity matrixes are good couple for PageRank algorithm, but word2vec similarity matrix does not affect the result positively as expected. PageRank with using only Word2Vec (not shown in the result) gives 0.44 ROUGE-1 score with this document set and it can not be considered as PageRank with Word2Vec similarity matrix both are not like they can be handled together.

TABLE VII
The Option Set That Is For This Experiment

No	Method Clustering/PageRank	Similarity Matrixes	Anaphora Removal
1	Clustering Centroid: Similarity Matrix	TfxIdf Sentiment Word2Vec	Yes
2	Clustering Centroid: TfxIdf	Word2Vec	No
3	Clustering Centroid: Similarity Matrix	TfxIdf Word2Vec	Yes
4	Clustering Centroid: TfxIdf	Sentiment Word2Vec	No
5	Clustering Centroid: Similarity Matrix	TfxIdf Word2Vec	No
6	PageRank	TfxIdf Sentiment	Yes
7	PageRank	TfxIdf	Yes
8	PageRank	TfxIdf Sentiment Word2Vec	Yes
9	Clustering Centroid: TfxIdf	TfxIdf Word2Vec Sentiment	Yes
10	Clustering Centroid: Similarity Matrix	TfxIdf Sentiment Word2Vec	No

VII. Conclusion

News reports, social media streams, blogs, digitized archives, books… there is an excessive amount text on web. And people, somehow, need to find what they are looking for. This raises the question of how to best generate automatic summaries. Many existing methods for extracting summaries rely on comparing the similarity of two sentences in some way. In this work, we presented to use new ways of measuring this similarity, based on sentiment analysis and continuous vector space representations. We investigated the effects using these sentence similarity measures at the same time, and infer that combined similarity measures can achieve better results for text summarization task.

References

[1] M. Kageback, O. Mogren, N. Tahmasebi, and D. Dubhashi, "Extractive Summarization using Continuous Vector Space Models," *Proc. 2nd Work. Contin. Vector Sp. Model. their Compos.*, pp. 31–39, 2014.

[2] D. R. Radev, H. Jing, and M. Budzikowska, "Centroid-based summarization of multiple documents: sentence extraction, utility-based evaluation, and user studies," *Inf. Process. Manag. 40.6 919-938.*, vol. 40, no. 6, p. 10, 2000.

[3] G. Erkan and D. R. Radev, "LexRank : Graph-based Centrality as Salience in Text Summarization," vol. 22, pp. 457–479, 2004.

[4] Y. Xia, Y. Zhang, and J. Yao, "Co-clustering sentences and terms for multi-document summarization," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 6609 LNCS, no. PART 2, pp. 339–352, 2011.

[5] Y. J. Kumar, O. S. Goh, H. Basiron, N. H. Choon, and P. C. Suppiah, "A review on automatic text summarization approaches," *J. Comput. Sci.*, vol. 12, no. 4, pp. 178–190, 2016.

[6] R. Mihalcea and P. Tarau, "TextRank: Bringing order into texts," *Proc. EMNLP*, vol. 85, pp. 404–411, 2004.

[7] M. Gambhir and V. Gupta, "Recent automatic text summarization techniques: a survey," *Artif. Intell. Rev.*, pp. 1–66, 2016.

[8] J. M. Kleinberg, "Authoritative Sources in a Hyperlinked Environment," *J. ACM*, vol. 46, no. May 1997, pp. 668–677, 1999.

[9] L. Page, S. Brin, R. Motwani, and T. Winograd, "The PageRank Citation Ranking: Bringing Order to the Web," *World Wide Web Internet Web Inf. Syst.*, vol. 54, no. 1999–66, pp. 1–17, 1998.

[10] M. Bonzanini, M. Martinez-Alvarez, and T. Roelleke, "Extractive Summarisation via Sentence Removal: Condensing Relevant Sentences into a Short Summary," *Proc. 36th Int. ACM SIGIR Conf. Res. Dev. Inf. Retr.*, pp. 893–896, 2013.

[11] K. Ganesan, C. Zhai, and J. Han, "Opinosis: A Graph Based Approach to Abstractive Summarization of Highly Redundant Opinions," in *Proceedings of the 23rd International Conference on Computational Linguistics*, 2010.

[12] G. Salton and M. J. McGill, "Introduction to modern information retrieval.," *Introduction to modern information retrieval.* p. 400, 1983.

[13] O. Mogren, M. Kågebäck, and D. Dubhashi, "Extractive Summarization by Aggregating Multiple Similarities," *Recent Adv. Nat. Lang. Process.*, pp. 451–457, 2015.

[14] Google, "Word2Vec." [Online]. Available: https://code.google.com/archive/p/word2vec/. [Accessed: 16-Jan-2017].

[15] C. Y. Lin, "Rouge: A package for automatic evaluation of summaries," *Proc. Work. text Summ. branches out (WAS 2004)*, no. 1, pp. 25–26, 2004.