I.J. Modern Education and Computer Science, **2019, 4, 23-31**
Published Online April 2019 in MECS (http://www.mecs-press.org/)
DOI: 10.5815/ijmecs.2019.04.03

Textual Coherence Improvement of Extractive Document Summarization Using Greedy Approach and Word Vectors

Mohamad Abdolahi
Kharazmi International Campus Shahrood University Shahrood, Iran
Email: mabdolahi512@yahoo.com

Morteza Zahedi
Kharazmi International Campus Shahrood University Shahrood, Iran
Email: zahedi@shahroodut.ac.ir

Received: 04 January 2019; Accepted: 26 February 2019; Published: 08 April 2019

***Abstract*—**There is a growing body of attention to importance of document summarization in most NLP tasks. So far, full coverage information, coherence of output sentences and lack of similar sentences (non-redundancy) are the main challenges faced to many experiments in compacted summaries. Although some research has been carried out on compact summaries, there have been few empirical investigations into coherence of output sentences. The aim of this essay is to explore a comprehensive and useful methodology to generate coherent summaries. The methodological approach taken in this study is a mixed method based on most likely n-grams and word2vec algorithm to convert separated sentences into numeric and normalized matrices. This paper attempts to extract statistical properties from numeric matrices. Using a greedy approach, the most relevant sentences to main document subject are selected and placed in the output summary. The proposed greedy method is our backbone algorithm, which utilizes a repeatable algorithm, maximizes two features of conceptual coherence and subject matter diversity in the summary. Suggested approach compares its result to similar model Q_Network and shows the superiority of its algorithm in confronting with long text document.

***Index Terms*—**Natural language processing, Extractive summarization, Text coherence, Word vector, Language models.

I. Introduction

Text summarization is the process of condensing the input text using computer programming into a shorter size and saves much more its main information. Generating a compressed text of original document is an intelligent effort that requires an overview and full comprehension of the main subject matter. Over the past recent decades there has been a dramatic increase in text summarization field. The most important document summarization fields are story summarization, novels, screenplays, information on the internet, patient medical records, voice services for deaf, data retrieval and document sorting. Two important aims of a text summarization are complete coverage of original text important information and the coherence of the output summary. But the output coherence is more important than information completeness. Incomplete information is determined by full awareness of all entire contents of original text. But the lack of coherence is considered at the beginning of review. The lack of summary coherence is not only reducing its readability, but also reduces the reader's trust to summary content and encourage them to overview the original text. As result generating coherent summary is the main concern of the text summarization systems in both local and global coherence. In addition to creating a coherent summary, all of these systems have tried to evaluate their output coherence as far as improve it [1, 2].

Automatic text summarization is divided into two major sections of extractives and abstracts summarization. Due to algorithms simplicity and lack of engaging with semantic concepts, there are much more studies on extractive summarization. The emphasis of these approaches is determining important sentences and lexical relations between them. Extractive methods are engaging with some criteria such as frequency of words and phrases, position of signs, sentence length and statistical association of sentences word and document title. The total of mentioned criteria produces a weight for each sentence, which determines the degree of sentence significance and its chance to presence in summary text. According to mentioned difficulty, extractive methods have much more incoherence consecutive sentences. To date, most researches have tended to focus on semantic algorithms or linguistic patterns introduced by Halliday [3]. These patterns are also highly dependent on words semantic meaning, and so difficult to implement the algorithms to extract the features.

This study provides a statistical approach based on greedy algorithms to advance our optimized extractive summarization method. Due to word vectors generated by word2vec algorithm, the proposed method provides coherence summaries that they have no dependency on subject and special field. In light of recent events in statistical methods, the proposed method has some advantages. The mentioned method in this paper attempts to shows suggested approach does not engage with words semantic meanings and no dependency on specific field's knowledge. These advantages make the proposed model suitable for high redundancy documents, very close sentences concept texts and long narrative documents.

In this method, at first preprocessing algorithm is taking place on input text. Then the text is partitioned into its main components. In the next step, using the word2vec algorithm, each word is converted into a vector, and each sentence becomes as numerical matrix. Then, using the most likely n-grams in the same text, the matrices of sentences are normalized. Finally, unlike other summarizing methods, without considering lexical meaning and textual concept, the numeric matrices of sentences are scored and the most appropriate and relevant sentences are extracted. In addition to more speed, the most interesting finding of the method is producing higher topic coherence summaries, and creating more optimal output in long documents.

The overall structure of the study takes the form of these sections. The following section of this paper gives a brief overview of the recent history of text summarization and its coherence evaluation. Then focusing on Google word2vec algorithm, all words are converted into numerical vectors. In the next part, concerning with the methodology used for the study and focusing on sentences are converted to numerical matrices and matrices are normalized. At the end of this section proposed greedy algorithm and its criteria is employed to select prominent sentences. The followed section presents the findings of the research, focusing on the proposed system evaluation criteria and the experimental results. Finally, in the end section, conclusion of the proposed approach is presented.

II. Related Works

The first systematic study of automatic text summarization was reported by Luhn et al. in 1958 [4]. Their preliminary works have attempted to detect and extract important sentences based on their simple revelation features. The most important features that they concern with are sentence position in text, words frequency, and some important keywords related to main subject [4, 5]. Selecting important sentences to generate a summary can be done statistically and semantically. Statistical approaches, however, are easier to implement with have higher speeds, because they do not interfere with semantic concepts. In recent years, researchers have investigated a variety of statistical text summarization approaches. A considerable and most common amount of statistical methods has been published in the field of naive Bayesian classifier [6, 7]. The other most important statistical prominent sentences extraction is LSA algorithm [8]. While LSA algorithm has brought many benefits to document summarization, there have been some shortcomings and recent approaches have mentioned prominent methods to optimize its shortcomings [9]. Applying SVD algorithm on text summarization was first carried out by Gong. Y et al [10]. In their method, input text was inserting into matrix and applying the algorithm to detect and extract important sentences. Steinberger et al suggested an optimized SVD method in far more cost effective, and improved the shortcomings of previous methods [11]. Lexical chains extraction methods were studied at first by Barzilay and Elhadad [12]. One of the most important advantages of lexical chains extraction is the simplicity of identifying their communications in the documents. This method uses the wordnet database to determine the relationship between textual units and creates a conceptual chain link between them. The assigned scores are determined by the number and type of chains links. The final summary consists of sentences that have the strongest chain link.

Some other authors have also mainly been interested in lexical chain sentence extraction. Pourvali et al suggests a method that firstly found the correct concept of each word, and then based on the words concept, a lexical chain was created and sentences were scored [13]. Finally, low score chains were eliminated and the main subject matter is discovered in respect to remaining chains. Santos and Sofia offers a clustering theory for extractive summarization. They put the words into semantic clusters and extracted semantic patterns [14]. Created clusters are hierarchically grouped into a binary tree. They introduced and used LS database in their thesis. In this database words with the same meaning were grouped together and a word can belong to different clusters [4, 15].

More recently, variety of approaches has emerged to offers effective findings about using graph theories in extractive summarization [16, 17]. In graph based approaches, after initial preprocessing, the sentences are placed as graph nodes and the conceptual relation between sentences forms as its edges. Given the importance linking of two sentences, the relationship has a weight (weighted graph), and each node can communicate with several other nodes (multinational). Text Rank algorithm is one of the most important methods used in graph based text summarization [17]. The algorithm scores to text sentences using unsupervised methods. In another major study, Christensen et al proposed novel G-Flow graph techniques [18]. Parveen and Mesgar draws their attention to name and name phrase entities to provide a new approach [19]. They offer an extractive summarization model based on combination theory of entity-grid [12] and Guinaudeau_Strube graph-based model [20].

Data mining methods and clustering algorithms have long history and very significant impact on producing optimal summaries [21]. In these methods, different components of text place into clusters. The different components can be words, phrases, sentences, and even

paragraphs. These methods provide an exciting opportunity and better performance to advance big data and texts with high number of sentences [9]. Euclidean distance, Cartesian similarity and Cosine similarity are the most common criteria of sentences clustering methods [22]. Some other writers have attempted to propose clustering methods for user-question based document summarization approaches [23, 24, 25]. Kumar and Soumya performed a combination clustering and SVM method to propose coherence extractive summarization system [26]. Fuzzy logic is also implementing to create coherent summaries. In fuzzy approaches, the fuzzy rules and membership functions have a direct impact on the function of the fuzzy system and output summary [27, 28, 29]. Text summarization is an interesting field in medical science and clinical biographies. Moradi and Ghadiri draw a simple Bayesian classifier approach of document summarization in medical fields [30].

This paper proposes a new and different methodology for single document extractive summarization which is a combination of greedy algorithm and statistical methods based on backpack algorithm. The main advantage of proposed method is maximizing two properties of conceptual coherence and thematic diversity of the remaining sentences.

III. Google Word2vec Algorithm

In most text processing techniques, each word is represented as a numeric vector. The numerical representation of the text offers benefits such as the possibility of numerical processing and machine learning algorithms [31]. One of the most important approaches of words converting into numeric vectors is word2vec algorithm. This approach was proposed by Google in 2013 and aims to create a vector space for similar words [32]. Word2vec creates a numerical vector for each word based on specific word's features such as its actual meaning in relation to the neighboring text parts, without human intervention. In a simpler say, word2vec offers understanding of the semantic meaning of a word based on its previous presence in a large set of text documents. In many research areas, such as sentiment analysis, similar texts searching, auto-scoring articles, text generating and text simplifying, word2vec is able to generate simpler algorithms and produce more accurate results.

IV. Research Methodology

Generating more coherent summaries, evaluating the coherence of the summary or, if possible, improving it, is one of the biggest concerns of all extractive document summarizations. A summary text should have three features:

- Including the important information of input document
- There is no unnecessary and extra information
- Consecutive sentences have coherence and thematic relation

This study set out with the aim of extracting related and coherence sentences to generate optimal summary. The greedy and numerical matrix based combined method tried to produces a summary with related sentences and non- extra information in determinable output size. Fig. 3 provides an overview of architecture of proposed method. This study provides new insights into three headings of preprocessing, generating and normalizing numeric matrices of sentences, and greedy algorithm for extracting related and coherent sentences.

A. Text Preprocessing

The first and most important issue in any text processing operation is preparing input text for applying the subsequent algorithms. The initial preprocessing is different according to text type, language, and text processing field. The preprocessing types and the amount of its exploiting rate have huge impact on speed and accuracy of final processing algorithms. Most important preprocessing algorithms are tokenization, stop word removal, stemming and POS tagging. However, their challenges are restricting to a particular filed, having no ability to apply and extend to all areas and languages.

In our proposed method, all sentence components are needed. Therefore, preprocessing algorithms are different from other previous methods. We do not apply stop word removal, stemming and POS tagging. Our reasons for not applying them are as follows:

- **Stop words**: Sentence matrix is including all word vectors. Stop words have significant effects on sequential sentences coherence. Every word has an important impact on other words and it should not be deleted.
- **Stemming**: There are conceptual differences between each word and its root. There are also differences between each word and its stem vector. Each word has a different meaning in its own form with its own unique suffixes and prefixes. In our proposed approach, word position, its writing form and the word concept have direct effect on output result. Therefore, every word vector created by the word2vec algorithm determines its position in the sentence and its grammatical rule.
- **POS tagging**: It is highlighting the grammatical behavior of every sentences part. Our method is purely statistical and word2vec is generating vectors which have its built-in grammatical behavior. POS tagging is a semantic operation and no need to perform on our input texts.
- **Do not convert uppercase to lowercase characters**: Due to grammatical role and word place in sentence, in most cases, words beginning with uppercase or lowercase have different concepts and also different vectors. As result, they should appear in original writing characters.

Our study offers some important insights into other preprocessing methodology and input text normalizing. Fig.1 provides an overview of proposed text preprocessing.

- **Separating sentences and words (tokenization)**: Firstly, sentences and words are separated using NLTK tokenizer.
- **Removing extra space characters between words:** It is one of the earliest preprocessing in any text preprocessing, but it has much more importance in our proposed approach.
- **Insert an empty space between words and the following punctuation mark**: In standard texts, there is no distance between any word and its following punctuation mark. When a word is converted into a vector, this mark is superimposed on its word, and the created vector is different. Given that word2vec algorithm generates specific vector for each punctuation mark, and processed it individually.
- **Uniform accents characters**: Accent characters are particular type of characters in different Latin form languages with slight difference in their pronunciation. Usually these characters are borrowed names or words of different languages in the text. In our proposed approach, all these characters are converted into Standard English characters.
- **Select the closest word vector for words that they are missing in the word2vec database**: Although our using word2vec database is a collection of 132400 words and their vectors, but there are possible words in the text that cannot be included in the database. In this case, the vector of word with largest common subclass is selected and placed in the sentence matrix.

B. Converting sentences to matrices and matrices normalizing

Previous published studies are limited to extracting prominent sentences based on some features like keywords, resemblance to text title, sentences location, tag phrases and etc. [33, 34]. In one well-known recent experiment, Lin and Bilmes offer a deep learning word vector algorithm model (Q-Networks method) [35]. The proposed model uses 50- dimensional GloVe vectors generated by Pennington [36]. They have used two vectors to convert the input text to numerical values. The first vector (ConVec) represents meaning of sentences, and the second vector (PosVec) represents sentence position in the text. The present research in this study explores a deep learning vector space like Lin and Bilmes method. But our proposed vectors are 100-dimensional vectors of Google's word2vec model.

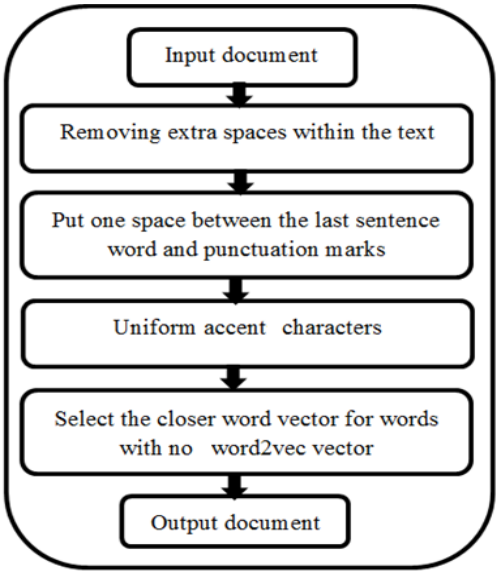

Fig.1. The preprocessing model

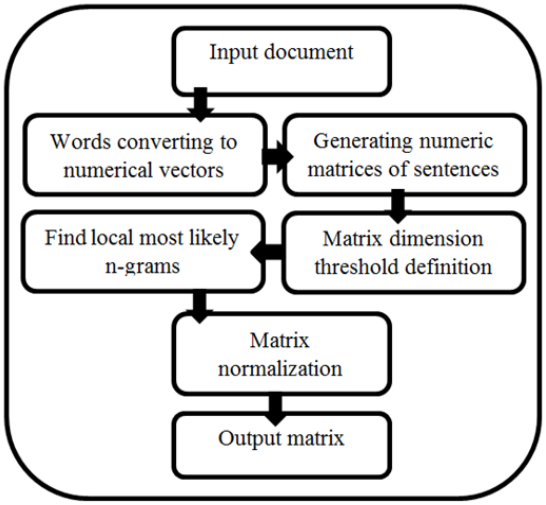

Fig.2. Converting sentences to numerical matrices and normalizing them

The novelty of this study is to investigate that instead of sentences importance evaluations, importance consideration are performed on numerical matrices of sentences. Our first steps are applying necessary preprocessing and sentences separating. Then, using word2vec word vectors, each sentence convert to a numerical matrix. Applying the local most likely n-grams, this step, lead to normalized numerical matrices [37]. The meaning of local n-grams is n-grams in the same original text and meaning of global n-grams, n-grams in huge body of the web corpuses. Using local n-grams tend to be more speed and local concept selection whereas global n-grams have a low search speed, enormous database and n-grams with less relatedness to local text. Fig. 2 presents an overview of converting and normalizing sentences to numerical matrices [37].

C. Proposed Greedy Algorithm

There have been much more proposed solutions to achieve three main goals in text summarization (completeness of information, coherence of output summary, and non-overlapping of sentences thematic). While some research has been carried out on entity grid based approaches [38], our study has been proposed a greedy approach based on backpack algorithm and target sampling. The method attempts to maximize two properties of conceptual coherence and sentences subject variation in summary output. Target sampling is one of the most effective algorithms to deal with imbalance numbers data. Because of the imbalances between selective and non-selective sentences, extractive summarization is also deal with imbalance numbers data. In target sampling method, data balancing is done according to removing and adding sentences in summary set.

In this essay, we attempt to defend a simple sampling method [39]. Therefore, according to output capacity, sentences importance degree, non-overlapping, and simple sampling method, the most prominent sentences are selected based on the sub modularity function (1) and placed in the summarized text set [35].

$$f(S) = L(S) + \alpha R(S) \tag{1}$$

In this function S is the summary text, $L(S)$ is overlapping rate of extracted sentences and input text, $R(S)$ is the amount of sentences variety in output summary, and α is a constant coefficient ($0 < \alpha < 1$). It is a nonzero amount for determining the effect of sentences variety concept and obtained by (3). The $L(S)$ value is calculated by (2):

$$L(S) = \sum_{j \in v} \min\{\sum_{j \in S} sim(i, j), \alpha \sum_{j \in v} sim(i, j)\} \tag{2}$$

Where $L(S)$ stands for minimum summation of sentence similarity with other sentences in the summary text and its similarity to other sentences not included in the summary. Parameter α is also the same previous coefficient value between zero and one ($0 < \alpha < 1$) that adjusts the effect of the second parameter. The value of $R(S)$ calculates the difference between selected sentence and other not selected sentences in the original text. To calculate $R(S)$, firstly, text sentences are clustered based on their topic. The initial score of each cluster is zero. By selecting each sentence to include in the summary set, an increasing score is given to its cluster. To select a new sentence to put in the summary set, a sentence has more priority that its cluster has les score. In this case, topic variety of selected sentences goes up to the optimum level.

$$\alpha = 1/L \tag{3}$$

Where L is stands for the number of sentences in cluster k at the end of clustering. In this case, sentences in the smaller clusters have higher priority, because they have a more thematic difference with other sentences.

$$R(S) = \max\left(1/p(cluster)_{i=1..k}\right) \tag{4}$$

In equation (4), k is the number of created clusters, $p(cluster)_i$ is the cluster score of selected sentence. Fig.3 provides the summary statistical proposed algorithm. Condition A refers to the number of sentences in the summary set that have not reach to predefined threshold of output set. Condition B refers to output summary set capacity which is completed but the algorithm finds a new sentence with merit and ability to insert in the summary set.

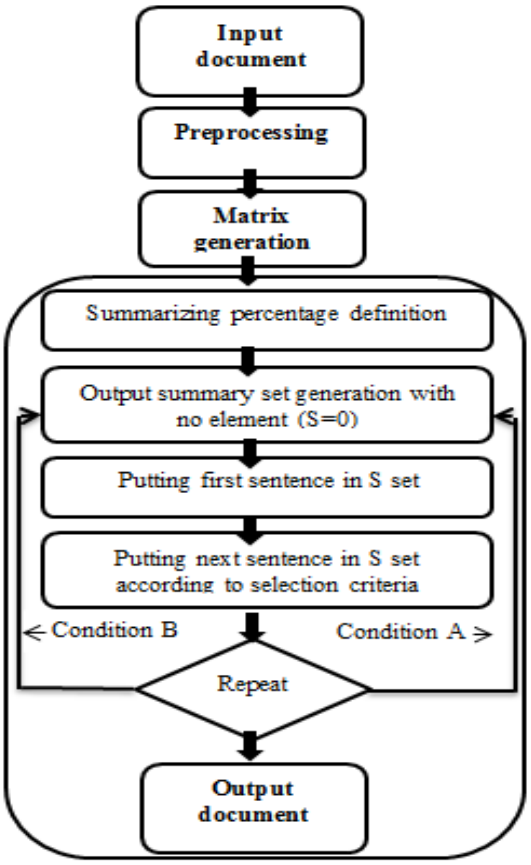

Fig.3. Proposed algorithm diagram

The following explanation seeks to explain the proposed algorithm:

1. Preprocessing input document and converting sentences into numeric matrices.
2. Determining the length of output summary (the output set capacity and the number of sentences in the summary text).
3. Creating output summary set with initial value of zero sentences.
4. The first sentence (subject sentence) is put into summary set.

5. Comparing next sentences with the subject's sentence (in summary set) according to predefined criteria and threshold level, put into summary set if it is nominated.

6. Repeating step five to satisfy one of two conditions:

- Consider entire original text sentences, the number of sentences in the summary set have not reach to predefined threshold of output set.
- Output summary set capacity is completed; the algorithm finds a new sentence with merit and ability to insert in the summary set.

7. **First condition**: algorithm starts at second stage with lower threshold level.

8. **Second condition**: Compare score value of newly selected sentence and sentence with the minimum score value in summary set, if the new sentence is higher score, the minimum score sentence in summary set is deleted and new sentence are placed into the end of the summary set.

The proposed algorithm offers some important advantages:

- There is no need to rearrange the sentences order. Summarized sentences are placed in same place as in the original text.
- The summary size is approximately equal to requested or defined percentage.
- The sentences in the summary have the highest score level based on scoring criteria. Therefore, the output text has three main conditions of optimal summary given the desired output percentage.
- The proposed method provides an optimal response to texts with high number of sentences.

D. Sentences selecting criteria

In order to selecting important and more related sentences, Cosine Similarity and Euclidean Distance are conducted. Due to converting sentences to numerical matrices, the matrices of sequences sentences are compared and detached in order to put in the summary set. In Cosine Similarity, *A* and *B* are two compared matrices and *n* is the number of matrix components:

$$\text{Cos}ine\_Sim = \sum_{i=1}^{n} A_i \cdot B_i \Big/ \left(\sqrt{\sum_{i=1}^{n} A_i^z} \cdot \sqrt{\sum_{i=1}^{n} B_1^z} \right) \quad (5)$$

In Euclidean distance equation (6), *n* is also the number of matrix components, *p* is the first sentence matrix and *q* the matrix of the compared sentence:

$$Eclidead\_Dist = \sqrt{\sum_{i=1}^{n} \left(p_i - q_i \right)^2} \quad (6)$$

V. Results and Discussion

A. Proposed system evaluation

To evaluate the quality of summary texts, novel ROUGE criteria are proposed and used in many text summarization approaches [9, 40]. Summary evaluation criteria have different parameters according to evaluation methods. In most evaluation approaches, the results are compared to human produced summaries, or artificial intelligence algorithms like supervised learning methods [35]. ROUGE_L and ROUGE_N are two important criteria used in this study. ROUGE_N, compares the number of n_grams of machine summarized and expert summarized text. ROUGE_L, compares the highest common sub-sequences common sentences of machine summarized and expert summarized text. Based on ROUGE_L, the higher common sub-sequences machine summarized and expert summarized text, the summary text has a higher accuracy. ROUGE_N is calculated by (7):

$$ROUGE\_N = \frac{\sum_{S \in sum\_ref} \sum_{n\_gram \in S} count\_match(n\_gram)}{\sum_{S \in sum\_ref} \sum_{n\_gram \in S} count(n\_gram)} \quad (7)$$

Where *n* stands for the length of the n-gram, count_match (gram) is the maximum number of n-grams co-occurring in a candidate summary and *S* is set of expert summaries. Count (n-gram) is the maximum number of n-grams co-occurring in a reference summary and expert summaries. *ROUGE_L* is calculated by (8):

$$ROUGE\_L = \left(1 + B^2\right) R_{LCS} \times P_{LCS} \Big/ R_{LCS} + \left(B^2 \times P_{LCS}\right) \quad (8)$$

R_{LCS} and P_{LCS} values are obtained by (9) and (10):

$$\boldsymbol{P}_{LCS} = \sum_{i=1}^{n} LSC\left(\boldsymbol{r}_i, S\right) \Big/ \sum_{i=1}^{u} \left|\boldsymbol{r}_i\right| \quad (9)$$

$$\boldsymbol{R}_{LCS} = \sum_{i=1}^{n} LSC\left(\boldsymbol{r}_i, S\right) \Big/ |S| \quad (10)$$

Where *S* stands for machine summary text, *r* stands for expert summary text, |s| is the number of sentences in the summary text, *LCS* (r_i , *s)* the length of common *LCS* machine summary and expert summary and *B* is a constant parameter.

B. Experimental results

This section is concerned with the experimental results. To evaluate the proposed method, the output system is compared to human generated summaries. In order to examine the method, ten different lengths Hans Christian Anderson's stories have been selected. For the purpose of analysis, ten summaries were created by various experts for each story. In all different expert summaries, the number of each sentence repetitions in all ten summaries is calculated. Then the sentences with the most repetition are put into a new summary named reference summary. The generated reference summaries contain more repetitive sentences in ten expert summaries, accordance with the percentage of the requested summary (in our propose method 50%). To compare proposed method and expert summaries, the number of their common sentences was considered and compared with reference summary. The results of applied algorithm on one of stories are presented in table 1. The selected story has 192 sentences and 4506 words. The summary length is 96 sentences with words number depending on selected sentences. Table 1 compares the results obtained from preliminary analysis of Q-Networks approach and our proposed model. In this analysis, at first, the number and percentage of common sentences in expert created and reference summary are calculated. Then the algorithm extract the number and percentage of common sentences in expert created and Q-Networks method summary and finally the method obtained the number and percentage of common sentences in expert created and our proposed method summary.

Table 1. Comparison of the proposed algorithm results and Q-Networks model on one of the summarized stories

A	B	C	D
1	91.67	92.71	91.67
2	92.71	91.67	93.75
3	89.58	94.79	95.83
4	93.75	90.63	91.67
5	95.83	92.71	94.79
6	94.79	92.71	95.83
7	89.58	90.63	92.71
8	90.63	84.46	91.67
9	91.67	91.67	93.75
10	87.5	90.63	95.83
averages		**91.26**	**93.75**

Table 1 Shows the results obtained from the preliminary analysis of proposed algorithm results and Q-Networks model on one of the summarized stories and 2.49% optimization in the summary generated in long texts. Table columns are presented the following details:

A- Ten different sample expert summaries.

B- The percentage of common sentences in expert created and reference summary.

C- The percentage of common sentences in expert created and Q-Networks method summary.

D- The percentage of common sentences in expert created and our proposed method summary.

Table 2 presents experiments results on ten selected different lengths documents. It is also presents the number of each document sentences before and after summarization (50% compression). The experiments results on different lengths documents show that the Q_Network method provides better result on short documents. But there are significant results in our proposed method in long documents. Overall, these results indicate that our proposed method obtain significant improvement (1.47%) in average results of ten different lengths document (short to long documents). Table columns are presented the following details:

A- Selected documents.
B- The number of sentences of each document.
C- The number of sentences of each reference summary.
D- Common sentences of reference summary and Q_Network method.
E- The percentage of common sentences in reference summary and Q-Networks method summary.
F- Common sentences of reference summary and proposed method.
G- The percentage of common sentences reference summary and our proposed method summary.

Fig. 4 compares the results obtained from the analysis of ten reference summaries on our proposed and Q-Networks method. From the chart, it can be seen that proposed method provides an optimal response to texts with high number of sentences (short 45 sentences documents to long 260 sentences documents).

Table 2. Comparison of the proposed algorithm results and Q-Networks model on ten selected stories

A	B	C	D	E	F	G
1	260	130	118	90.77	123	94.62
2	192	96	88	91.67	91	94.79
3	169	84	77	91.67	81	96.43
4	111	55	43	78.18	45	81.82
5	101	50	44	88	45	90
6	86	43	38	88.37	38	88.37
7	82	41	34	82.93	36	87.8
8	70	35	32	91.43	32	91.43
9	68	34	28	82.35	27	79.41
10	45	22	20	90.91	19	86.36
averages				**87.63**		**89.1**

VI. CONCLUSION

Generating coherence summaries and including much more original input text information are the most important goals of all proposed text summarization approaches. Therefore, all researchers try to produce more compact summaries and full coverage information. The present study is designed a single document extractive text summarization to extract prominent sentences. The study has gone some way towards maximizing two properties of conceptual coherence and variation in the subject matter of the sentences. Therefore,

according to output capacity, sentence important degree and non-overlapping, the most important sentences are selected and put in the summarized set. In this approach, firstly, sentences are converted into matrices using generated vectors of word2vec algorithm. Then, using statistical method and most likely n-grams, the matrices are normalized. Consequently, instead of comparing and choosing outstanding sentences semantically, the matrices are statistically compared and the most appropriate sentences are extracted.

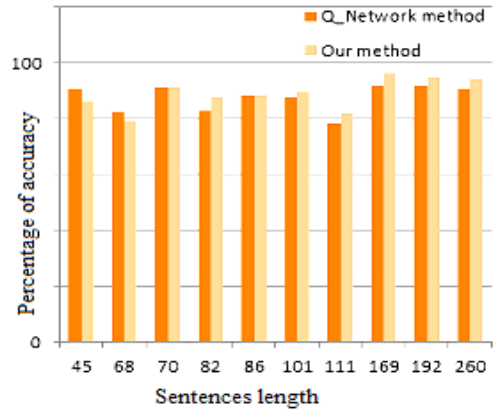

Fig.4. Comparison diagram of our proposed and Q-Networks method

The findings of this study make several optimizations such as non-dependency on document field, subject, language, not engaging with words semantic meanings and sentences, simplicity and high speed algorithm, and much more coherence summaries. The result of mentioned method shows more optimal results on long documents. Other optimizations of this study is including important information of input text, summary with no un-necessary and extra information, and having coherence and thematic relation in consecutive sentences. The experimental results show that the proposed method has more coherent summary and closer to human generated samples, and simple methodology comparing to entity-based and graph-based methods.

REFERENCES

[1] L. Alonso, i Alemany and M. F. Fort, "Integrating cohesion and coherence for Automatic Summarization," *in Proceedings of the tenth conference on European chapter of the Association for Computational Linguistics*, Vol. 2, 2003, pp. 1-8, 2003.

[2] M. Abdolahi and M. Zahedi, "An overview on text coherence methods," *Eighth International Conference on Information and Knowledge Technology (IKT)*, pp. 1-5, 2016.

[3] M. A. K. Halliday and R. Hasan, Cohesion in English, *Longman Group Limited London*, 1976.

[4] H. P. Luhn, "A business intelligence system," *IBM Journal of research and development*, vol. 2, no. 4, pp. 314-319, 1958.

[5] H. P. Edmundson, "New methods in automatic extracting," *Journal of the ACM (JACM)*, vol. 16, no. 2, pp. 264-285, 1969.

[6] N. Ramanujam and M. Kaliappan, "An automatic multidocument text summarization approach based on Naive Bayesian classifier using timestamp strategy," *The Scientific World Journal*, vol. pp.1-15, 2016.

[7] A. P. Louis, "A Bayesian Method to incorporate background knowledge during automatic text summarization," *Association for Computational Linguistics*, pp.333-338, 2014.

[8] N. Alami, M. Meknassi, and N. Rais, "Automatic texts summarization: Current state of the art," *Journal of Asian Scientific Research*, vol. 5, no. 1, pp. 1-15, 2015.

[9] J. P. Verma and A. Patel, "Evaluation of Unsupervised Learning based Extractive Text Summarization Technique for Large Scale Review and Feedback Data," *Indian Journal of Science and Technology*, vol. 10, no. 17, pp.1-6, 2017.

[10] Y. Gong and X. Liu, "Generic text summarization using relevance measure and latent semantic analysis," *in Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 19-25, 2001.

[11] J. Steinberger and K. Jezek, "Using latent semantic analysis in text summarization and summary evaluation," *Proc ISIM*, vol. 4, pp. 93-100, 2004.

[12] R. Barzilay and M. Elhadad, "Using lexical chains for text summarization," *Advances in automatic text summarization*, pp. 111-121, 1999.

[13] M. Pourvali and M. S. Abadeh, "Automated text summarization base on lexicales chain and graph using of wordnet and wikipedia knowledge base," *arXiv preprint arXiv:1203.3586*, 2012.

[14] C. Santos, "Alexia-acquisition of lexical chains for text summarization," *University 0f Beira Interior, Covilhã, Portugal ,Citeseer*, 2006.

[15] P. Jain and S. Jain, "Summarizing Text Using Lexical Chains." *International Journal on Recent and Innovation Trends in Computing and Communication*. No.4, vol. 4, 2016.

[16] A. A. Natesh, S. TBalekuttira and A. P. Patil, "Graph based approach for automatic text summarization", *International Journal of Advanced Research in Computer and Communication Engineering (IJARCCE)*, no.5, vol. 2, 2016.

[17] R. Mihalcea, "Graph-based ranking algorithms for sentence extraction, applied to text summarization," *in Proceedings of the ACL 2004 on Interactive poster and demonstration sessions*, 2004.

[18] J. Christensen, S. Soderland, and O. Etzioni, "Towards coherent multi-document summarization," *in Proceedings of the 2013 conference of the North American chapter of the association for computational linguistics: Human language technologies*, pp. 1163-1173, 2013.

[19] D. Parveen, M. Mesgar, and M. Strube, "Generating coherent summaries of scientific articles using coherence patterns," *in Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pp. 772-783, 2016.

[20] C. Guinaudeau and M. Strube, "Graph-based local coherence modeling," *in Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, vol. 1, pp. 93-103, 2013.

[21] C. E. Crangle, "Text summarization in data mining," *in Soft-Ware 2002: Computing in an Imperfect World: Springer*, pp. 332-347, 2002.

[22] A. Agrawal and U. Gupta, "Extraction based approach for text summarization using k-means clustering," *Int. J. Sci. Res. Publ.(IJSRP)*, vol. 4, no. 11, 2014.

[23] A. R. Deshpande and L. Lobo, "Text summarization using

Clustering technique," *International Journal of Engineering Trends and Technology*, vol. 4, no. 8, 2013.
[24] M. N. Ingole, M. Bewoor, and S. Patil, "Text Summarization using Expectation Maximization Clustering Algorithm," *International Journal of Engineering Research and Applications*, vol. 2, no. 4, pp. 168-171, 2012.
[25] V. Hatzivassiloglou, et al., "Simfinder: A flexible clustering tool for summarization," *Columbia University New York United States*, 2001.
[26] K. Shivakumar and R. Soumya, "Text summarization using clustering technique and SVM technique," *International Journal of Applied Engineering Research*, vol. 10, no. 12, pp. 28873-28881, 2015.
[27] F. Kyoomarsi, H. Khosravi, E. Eslami, and M. Davoudi, "Extraction-based text summarization using fuzzy analysis," *Iranian Journal of Fuzzy Systems*, vol. 7, no. 3, pp. 15-32, 2010.
[28] P. D. Patil and N. Kulkarni, "Text summarization using fuzzy logic," *International Journal of Innovative Research in Advanced Engineering (IJIRAE)* Volume, vol. 1, 2014.
[29] A. Kiani and M. R. Akbarzadeh, "Automatic text summarization using hybrid fuzzy ga-gp," *in Fuzzy Systems, 2006 IEEE International Conference*, pp. 977-983, 2006.
[30] M. Moradi and N. Ghadiri, "A Bayesian Approach to Biomedical Text Summarization," *CoRR, arXiv preprint arXiv:1605.02948*, 2016.
[31] G. H. Lee and K. J. Lee, "Automatic Text Summarization Using Reinforcement Learning with Embedding Features," *in Proceedings of the Eighth International Joint Conference on Natural Language Processing*,vol. 2, pp. 193-197, 2017.
[32] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *arXiv preprint arXiv:1301.3781*, 2013.
[33] S. R. Rahimi, A. T. Mozhdehi, and M. Abdolahi, "An overview on extractive text summarization," *in Knowledge-Based Engineering and Innovation (KBEI), 2017 IEEE 4th International Conference*, pp. 0054-0062, 2017.
[34] M. Allahyari, et al., "Text summarization techniques: a brief survey," *arXiv preprint arXiv:1707.02268*, 2017.
[35] H. Lin and J. Bilmes, "A class of submodular functions for document summarization," *in Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*-Vol. 1, pp. 510-520, 2011.
[36] J. Pennington, R. Socher, and C. Manning, "Glove: Global vectors for word representation," *in Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pp. 1532-1543, 2014.
[37] M. Abdolahi and M. Zahedi, "Sentence matrix normalization using most likely n-grams vector," *in Knowledge-Based Engineering and Innovation (KBEI), 2017 IEEE 4th International Conference*, pp. 0040-0045, 2017.
[38] R. Barzilay and M. Lapata, "Modeling local coherence: An entity-based approach," *Computational Linguistics*, vol. 34, no. 1, pp. 1-34, 2008.
[39] B. M, Derhami and V. Zarifzadeh, "A Method for Automatic Key phrase Extraction from Persian Web News", *Tabriz Journal of Electrical Engineering*, 47(3), 857-866, 2017, [in Persian]
[40] C.-Y. Lin, "Rouge: A package for automatic evaluation of summaries," *Text Summarization Branches Out*, pp. 74-81, 2004.

Authors' Profiles

Mohamad Abdolahi was born in Mashhad, Iran, on October 22, 1964, Thursday. He is Ph.D. candidate in Shahrood University of Technology in the field of computer engineering - artificial intelligence. His is lecturer in Iranian Academic Center for Education, Culture and Research (ACECR), Mashhad, Iran. His special fields of interest are NLP, data mining, image processing and machine learning.

Morteza Zahedi graduated from the RWTH-Aachen University, Aachen, Germany and assistant Professor in Shahrood University of Technology. His special fields of interest are NLP, pattern recognition, image and video processing.

How to cite this paper: Mohamad Abdolahi, Morteza Zahedi, "Textual Coherence Improvement of Extractive Document Summarization Using Greedy Approach and Word Vectors", International Journal of Modern Education and Computer Science(IJMECS), Vol.11, No.4, pp. 23-31, 2019.DOI: 10.5815/ijmecs.2019.04.03