Automatic Text Summarization Based on Multi-Agent Particle Swarm Optimization

Hamed Asgari
Department of Computer and
Information Technology Engineering
Science and Research Branch,
Islamic Azad University,
Qazvin, Iran
E-mail: H.asgari@Qiau.ac.ir

Behrooz Masoumi
Department of Computer and
Information Technology Engineering
Qazvin Islamic Azad University
Qazvin, Iran
E-mail: Masoumi@Qiau.ac.ir

Omid sojoodi sheijani
Department of Computer and
Information Technology Engineering
Qazvin Islamic Azad University
Qazvin, Iran
E-mail: o_sojoodi@m.ieice.org

Abstract—**Text summarization is the objective extraction of some parts of the text, such as sentence and paragraph, as the document abstract. If there are documents with a large amount of information, extractive text summarization would be arisen as an NP-complete problem. To solve these problems, meta-heuristic algorithms are used. In this paper, a method based on multi-agent particle swarm optimization approach is proposed to improve the extractive text summarization. In this method, each particle will be upgraded with the status of multi-agent systems. The proposed method is tested on DUC 2002 standard documents and analyzed by ROUGE evaluation software. The experimental results show that this method has better performance than other compared methods.**

Keywords- Text summarization, Particle swarm optimization, Multi-Agent Systems, Extractive method.

I. Introduction

Due to rapid growth of the information in the world, extensive access to the Internet and the creation of Internet sites and online textual resources, text summarization while investigating and studying the essential materials has received much more attention. Text summarization is a solution that gives users an overview of all relevant literature data needed, and this helps the user in next decisions making. Early studies on text summarization were proposed in the late 1950s, where it was possible to summarize the text by computer [1].

In general, there are several ways with different bases to summarize a text. In terms of how to summarize the text, there are two approaches for text summarization including extractive and abstractive summarization [2]. Based on the number of input documents, there are two other approaches including single document and multi-document summarization techniques. Another classification is based on the summarization purpose that is included query-based and public summarization. Extractive summarization that is discussed in this paper means the objective extraction of some text portions, such as sentences and paragraphs as the document abstract. Abstractive summarization is based on text understanding and rewriting in the form of some sentences, which is carried out by using linguistic techniques.

Several methods have been suggested for extractive text summarization including supervised, unsupervised and meta-heuristic algorithms [3]. In unsupervised algorithms and instead of detailed information, it is used statistical and linguistic information obtained from the text. TF-IDF algorithm is one of these unsupervised algorithms [4, 5]. In this method, the weighting is according to term-frequency and inverse sentence-frequency. Sentence-frequency refers to the number of sentences including a term. In this algorithm, some parts of a sentence may be repeated in the other sentences. The advantage of this method is its simplicity. As the disadvantage of the method, it may be frequented some of the words are not so important which may cause a deviation in text summarization.

Usually, the documents are written in such a way that the various topics are organized into one after another. These topics are written in either explicit or implicit sentences. This text organization will also apply to the summary. Some summarizers use clustering method for these aspects [6]. In clustering, the degree of similarity between sentences is investigated based on a set of parameters and the same sentences are placed in a cluster. So, each cluster represents a topic. Then, in each cluster, the most similar sentences into cluster topics are subjects to high scores to be selected for the summary. The advantage of this method is that any topics of the text can be identified, properly. Its disadvantage is that it is important to choose the number of clusters and the number may be too high or too low and fails to make it properly summarize. On the other hand, selecting the optimal number of clusters of a text document is difficult.

Another method for text summarization is graph theory. Like the clustering methods, graph theory is also used to identify the topics of a document [7, 8]. In this method, the document is considered an undirected graph. Each sentence in the text is shown with a node in the graph. Between two nodes, there is a bar if edges between two nodes indicate that they are included, they are similar in meaning as they are related to a topic. There is an edge between two nodes when they are similar. In other words, the two sentences are related to a topic. After the graph is created, those high degree nodes are selected for the summary. The advantage of this method is its simplicity

978-1-4799-3351-8/14/$31.00 ©2014 IEEE

Authorized licensed use limited to: Penn State University. Downloaded on March 31,2023 at 02:11:13 UTC from IEEE Xplore. Restrictions apply.

and as a result, does not require specific calculations. The disadvantage is that the small documents containing small number of sentences, good result is not achieved and the accuracy is reduced. In another method, called latent semantic analysis [9], semantic relationships between terms and between sentences are extracted. Then, the extracted relations are applied to SVD algorithm in the matrix form and after performing the necessary calculations, key phrases are selected. The advantage of this method is the use of term-concept vectors like a relationship that is created in the human mind resulting in easily understood by humans. Its disadvantage is the large amount and heterogeneity of information which results a reduced performance.

Another class of algorithms for extractive text summarization is supervised algorithms. In these algorithms, it is used the data set which are labeled by human. In other words, there is a set of input text and its summary. In this method, the text is initially divided into sections based on a set of parameters and each section is also represented by a set of features (such as the number of term-frequency, term location and the number of title words in each section) [10]. After performing feature extraction in each section, a supervised learning algorithm is used to train the summarizer. These algorithms include decision trees, Bayes rule, neural networks and fuzzy logic [11, 12]. The disadvantage of these methods is the accuracy and speed reduction when working on large documents, due to the large number of comparisons. The major problem while using the fuzzy logic is that if the rules are not defined properly, the accuracy drops.

Meta-heuristic algorithms are another group of algorithms for extractive text summarization. The main goal of these algorithms is finding sentences with a high score. Genetic algorithm (GA) and particle swarm optimization (PSO) algorithm are two examples of this group [13, 14]. The disadvantage of these algorithms is that it may stop at a local maximum or minimum, and obtain incorrect results.

The main challenge in extractive text summarization is large volume of information and large search space that cause NP-Complete problem. To work with large texts containing a lot of terms, the scoring and more importantly, sentence selection is very difficult which cause to reduce speed and accuracy in text summarization. In such circumstances, the use of meta-heuristic methods can be instrumental in achieving the optimum solution to solve a problem, in an effective way.

In this paper, the problem of text summarization using multi-agent particle swarm optimization algorithm takes into consideration. In this algorithm, each particle enhances its status by greater autonomy, asynchronous performing and learning ability [15, 16]. The results show the better performance of this method compared to other approaches.

The remainder of this paper is organized as follows: Section II introduces principal concepts. In section III, the proposed algorithm is presented and its features will be mentioned. Finally, section IV discusses carried out experiments and the results are compared with other methods.

II. Definitions and Basic Concepts

A. *Extractive text summarization*

Text summarization is a process to extract important summarized version for a user or particular users with a task or specific tasks. Extractive summarization method includes selecting important sentences, paragraphs of text and linking them together in a shorter form. As the advantages of this method, simplicity, high-speed summarization, costs and study time reduction could be mentioned. One disadvantage of this method is that the extracted sentences may be too moderate. Important and relevant information may also be distributed between sentences and extractive method can not identify them. The overall process of extractive summarization is performed in two steps: preprocessing and processing. In the preprocessing step, identifying the end of sentences, removing terms that have no meaning and finding the word roots is performed. In the process step, the effectiveness and relevance of the sentences to the topic are identified and assigned a weight to each sentence. At last, sentences with the highest scores are selected to be in final summarized text.

B. *Multi-Agent Particle swarm optimization*

The particle swarm optimization has inspired by the birds moving towards a common goal that very particle has independence, intelligence, and limited speed. Furthermore, multi-agent systems are composed of a set of independent agents that cooperate to achieve a common goal. An agent has characteristics such as autonomy, learning ability and cooperation. These features show that the agents are able to understand their surrounding environment and help each other.

In Multi-Agent Particle swarm optimization algorithm (MAPSO), each particle is defined as an agent [15]. The environment also as an agent makes environmental information available to other agents. In addition, each point of the space is devoted to a Boolean value that indicates whether this point has been met or not by the other agents. When the search is restricted to the points where are still not met, then search speed increases. Moreover, the labeled points of the space are clustered which cause the understanding of the problem space by the environment agent. In addition to the point labeling, the agents should perform their duties such as getting information about the best current solution of the neighbor agent, sending their best solution to the neighbor agents, and asking the environment agent about the clusters information.

Definition 1: The environment in MAPSO of n dimension problem space, denoted by *E*, is defined by Equation (1):

$$E = \{P, C, n, FF, \varepsilon, \varepsilon_d\}. \tag{1}$$

p is the set of points in the problem space where $\{p \square P, p = \{d_1 \times d_2 \times \ldots \times d_n, tag\}\}$, is a point that includes the n dimensional position and an extra boolean value called tag. C represents the set of clusters in E and is defined by $C=\{(p_1, p_2, \ldots, p_{center}, \ldots, p_m)\ , d \mid d \geq \square_d\}$.

FF is the fitness function and is defined within the environment because different criteria requires different fitness functions. ε is the minimum error criteria that is used to terminate the MAPSO execution. A predefined maximum iteration is employed to terminate the system when the swarm cannot find

Authorized licensed use limited to: Penn State University. Downloaded on March 31,2023 at 02:11:13 UTC from IEEE Xplore. Restrictions apply.

any solution to fulfilled criteria. ε_d is the minimum density requirement for the environment agent to identify if a group of tagged, non-optimal points can be considered as a cluster in the problem space.

Definition 2: The particle agent in MAPSO, denoted by *PA*, is defined by *PA:{pB, gB, N}*. N contains a set of particle agents which are predetermined as neighbors of *PA*. *PA* is defined with following functions:

- requesting each neighbor's current personal best location.
- returning its current personal best to neighbors.
- obtaining center location of each surrounding cluster.
- observing if the current position is visited.
- tagging current position if not optimal.

Definition 3: The original updating function in PSO is enhanced in MAPSO to take advantage of the new cluster concept and is defined by Equation (2):

$$V_{d+1}^{m} = V_{d}^{m} + R_{pBest}\left(pBest^{m} - X_{d}^{m}\right) \tag{2}$$

$$+ R_{gBest}\left(gBest^{m} - X_{d}^{m}\right)$$

$$- R_c \Sigma_{i=1}^{N_c}\left(C_{i_{center}}^{m} - X_{d}^{m}\right).$$

$$X_{d+1}^{m} = X_{d}^{m} + V_{d+1}^{m}\,.$$

Here $V_d^m + 1$ is the velocity of particle agent m in the iteration $d + 1$ and is used to compute particle agent's next position. The idea is to move toward the better solution area in the problem space and move away from the clusters at the same time. Furthermore, R_{pBest}, R_{gBest} and R_c are random numbers to retain the stochastic feature of the original PSO algorithm. The overall optimization procedure can is shown as a flow-chart in Fig. 1.

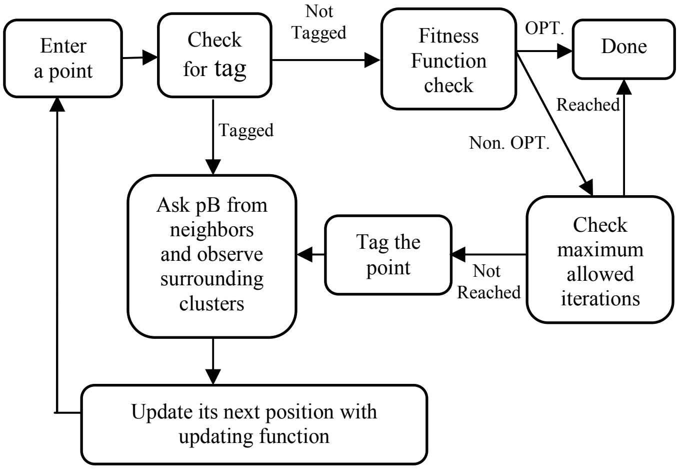
Enter a point
Check for tag
Not Tagged
Fitness Function check
OPT.
Done
Reached
Non. OPT.
Tagged
Ask pB from neighbors and observe surrounding clusters
Tag the point
Not Reached
Check maximum allowed iterations
Update its next position with updating function

Figure 1. Particle Agent Optimization Procedure [15].

III. THE PROPOSED METHOD

In this section, a new method for text summarization based on multi-agent particle swarm optimization is proposed. In this method, pre-processing must be performed on the input text, as in section II was noted. Then, using TF-IDF method, the sentences are extracted from the pre-processing step are given weight. In this weighting method, $freq_{i,j}$, $max_l freq_{L,j}$, N and n_i are supposed to be the number of term-frequency for *i*th term in *j*th sentence, maximum number of term-frequency for *l*th term in *j*th sentence, the number of sentences in the input text, and the number of sentences containing the word, respectively, Equation (3). After the sentences were given weight, similarity matrix should be obtained by using Equation (4). This matrix computes the similarity of sentences with key words.

$$tf_{i,j} = \frac{freq_{i,j}}{max_l freq_{l,j}} \qquad idf_i = log\frac{N}{n_i} \tag{3}$$

$$sim(s_i\,,\,q) = \frac{\Sigma_{i=1}^{t}\ w_{i,j} * w_{i,q}}{\sqrt{\Sigma_{i=1}^{t} w_{i,j}^{2}} * \sqrt{\Sigma_{i=1}^{t} w_{i,q}^{2}}} \tag{4}$$

So far, all terms and sentences existing in the input text are extracted and by calculations performing on them, the weight of each sentence and their similarity are specified. Now, this information must be applied to MAPSO algorithm to extract key sentences and summary of original text. As in section II was studied, MAPSO algorithm has a set of initial parameters that must be initialized at the beginning. To use this algorithm, the number of particle agent is considered 20 and the number of loop iterations is considered 100. Then, the total number of sentences, summarized sentences, and the similarity matrix are given to MAPSO algorithm as input parameters and also a number of sentences are randomly assigned to each particle agent. The cost of each particle agent is calculated by Equations (5) and (6). These two equations determine the dependency of sentences together and the readability of summarized sentences, respectively. Dependency factor cause the sentences contained in the summary text to discuss the same information. The sentence readability factor indicates that the first, second, and the other summarized sentences are related to each other with a high degree of similarity. Dependency factor is shown with CF_S where C_S represents the summary sentences mean similarity and M is the maximum weight of the sentence. Readability factor is shown with RF_S where s is represents the length of summary sentences.

$$C_s = \frac{\Sigma_{\forall s_i, s_j \in Summary\ subgraph}\ W(s_i s_j)}{N_s}, \tag{5}$$

$$CF_s = \frac{log\,(C * 9+1)}{log\,(M * 9+1)}$$

$$R_s = \sum_{0 \leq i < s} W(s_i, s_{i+1}) \ \ ,\ \ RF_s = \frac{R_s}{max_{\forall i}\, R_i} \tag{6}$$

Authorized licensed use limited to: Penn State University. Downloaded on March 31,2023 at 02:11:13 UTC from IEEE Xplore. Restrictions apply.

Finally, after full implementation of MAPSO algorithm, sentences with the highest values are extracted from the original text and displayed together as the output.

The procedure of proposed method has been as follow:

Step 1. Input source document.

Step 2. Preprocessing.

Step 2.1. Stemming.

Step 2.2. Remove stop words.

Step 2.3. Calculate the number of sentence.

Step 2.4. Making the similarity matrix.

Step 3. Processing.

Step 3.1. Initialize MAPSO Algorithm.

Step 3.2. Calculate cost of each agent.

Step 3.3. Calculate best sentences.

Step 4. Show extracted sentences.

IV. EXPERIMENT RESULTS

A. *Evaluation metrics*

To evaluate the extractive summarizers, it is usually used the precision, F-score, and recall measures [17]. These measures are respectively given in the Equations (7), (8) and (9).

$$Recall = \frac{RelevantSentences \cap RetrievedSentences}{Relevant\ Sentences} \tag{7}$$

$$Precision = \frac{RelevantSentences \cap RetrievedSentences}{Retrieved\ Sentences} \tag{8}$$

$$F\text{-}score = \frac{2 * Precision * Recall}{Precision + Recall} \tag{9}$$

Since the calculation of these measures is difficult and time-consuming, automated software called ROUGE is used for evaluation. This software is based on Perl language with different packages to evaluate the text. It will examine the aforesaid measures and show the results.

B. *obtained Results*

To evaluate the proposed technique, DUC 2002 standard documents are used. Then, obtained results are investigated by ROUGE 1 and compared with PSO and HBFO methods [18]. The results of this comparison are presented in Table I, and Fig. 2.

TABLE I. COMPARE F-SCORE RESULT BETWEEN MAPSO, PSO AND HBFO

F-score Results			
	MAPSO	***PSO***	***HBFO***
d061j	0.51016	0.42869	0.47926
d067f	0.48587	0.44637	0.47985
d073b	0.63195	0.58658	0.59096
d070f	0.50935	0.40453	0.46532
d079a	0.49198	0.41177	0.43530

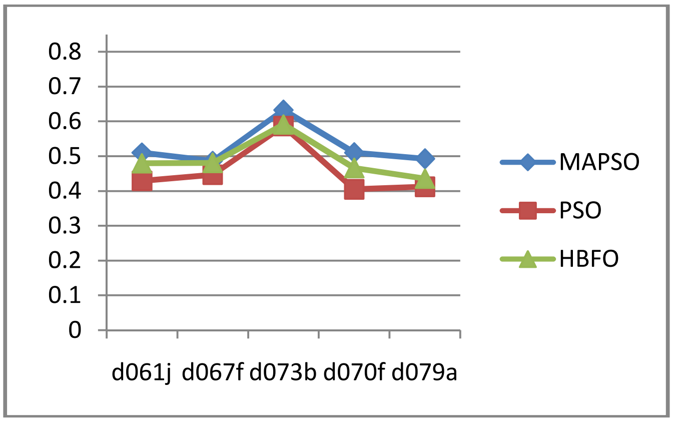

Figure 2. The results of comparison.

As can be seen above, a number of selected documents from the DUC 2002 set have been summarized. The results indicate that the proposed method has better performance than other methods.

V. CONCLUSION

In this paper, a new method based on multi-agent particle swarm optimization for automatic text summarization was proposed. the proposed method was tested with a set of DUC 2002 standard documents and the results were analyzed by ROUGE evaluation software and were also compared with other methods. The comparison showed that the proposed method has better performance than others.

REFERENCES

[1] E. Hovy, *Text summarization*. chapter The Oxford Handbook of Computational Linguistics, 2005, pp. 583-598.

[2] V. Gupta, "A Survey of Text Summarization Extractive Techniques " *Journal of Emerging Technologies in web Intelligence,* vol. 2, August 2010.

[3] M. N. Uddin and S. A. Khan, "A study on text summarization techniques and implement few of them for Bangla language," in *Computer and information technology, 2007. iccit 2007. 10th international conference on*, 2007, pp. 1-4.

[4] Y. Ledeneva, A. Gelbukh, and R. A. G. Hernández, "Terms derived from frequent sequences for extractive text summarization," in *Proceedings of the 9th international conference on Computational linguistics and intelligent text processing*, 2008, pp. 593-604.

Authorized licensed use limited to: Penn State University. Downloaded on March 31,2023 at 02:11:13 UTC from IEEE Xplore. Restrictions apply.

[5] R. A. Garcia-Hernandez and Y. Ledeneva, "Word Sequence Models for Single Text Summarization," in *Advances in Computer-Human Interactions, 2009. ACHI '09. Second International Conferences on*, 2009, pp. 44-48.

[6] P.-y. Zhang and C.-h. Li, "Automatic text summarization based on sentences clustering and extraction," in *Computer Science and Information Technology, 2009. ICCSIT 2009. 2nd IEEE International Conference on*, 2009, pp. 167-170.

[7] K. S. Thakkar, R. V. Dharaskar, and M. B. Chandak, "Graph-Based Algorithms for Text Summarization," in *Emerging Trends in Engineering and Technology (ICETET), 2010 3rd International Conference on*, 2010, pp. 516-519.

[8] K. Patil and P. Brazdil, " Sumgraph: text summarization using centrality in the pathfinder network " *IADIS International Journal on Computer Science and Information Systems* vol. 2, pp. 18-32, 2007.

[9] I. V. Mashechkin, M. I. Petrovskiy, D. S. Popov, and D. V. Tsarev, "Automatic text summarization using latent semantic analysis," *Programming and Computer Software,* vol. 37, pp. 299-305, 2011.

[10] W. Chuang and J. Yang, "Text Summarization by Sentence Segment Extraction Using Machine Learning Algorithms," in *Knowledge Discovery and Data Mining. Current Issues and New Applications*. vol. 1805, T. Terano, H. Liu, and A. P. Chen, Eds., ed: Springer Berlin Heidelberg, 2000, pp. 454-457.

[11] W. Song, L. Cheon Choi, S. Cheol Park, and X. Feng Ding, "Fuzzy evolutionary optimization modeling and its applications to unsupervised categorization and extractive summarization," *Expert Systems with Applications,* vol. 38, pp. 9112-9121, 2011.

[12] L. Suanmali, N. Salim, and M. Salem Binwahlan, "Fuzzy Logic Based Method for Improving Text Summarization," *International Journal of Computer Science and Information Security (IJCSIS),* vol. 2, Jun 2009.

[13] M. A. Fattah and F. Ren, "GA, MR, FFNN, PNN and GMM based models for automatic text summarization," *Computer Speech & Language,* vol. 23, pp. 126-144, 2009.

[14] F. Oi-Mean and A. Oxley, "A hybrid PSO model in Extractive Text Summarizer," in *Computers & Informatics (ISCI), 2011 IEEE Symposium on*, 2011, pp. 130-134.

[15] R. Ahmad, L. Yung-Chuan, S. Rahimi, and B. Gupta, "A Multi-Agent Based Approach for Particle Swarm Optimization," in *Integration of Knowledge Intensive Multi-Agent Systems, KIMAS 2007. International Conference on*, pp. 267-271, 2007.

[16] B. Zhao, C. X. Guo, and Y. J. Cao, "An improved particle swarm optimization algorithm for optimal reactive power dispatch," in *Power Engineering Society General Meeting, 2005. IEEE*, pp. 272-279 Vol. 1. 2005.

[17] A. Abuobieda, N. Salim, Y. Kumar, and A. Osman, "An Improved Evolutionary Algorithm for Extractive Text Summarization," in *Intelligent Information and Database Systems*. vol. 7803, A. Selamat, N. Nguyen, and H. Haron, Eds., ed: Springer Berlin Heidelberg, pp. 78-89, 2013.

[18] H. Asgari and B. Masoumi, " presented a hybrid methods to improve text summarization performance ", *the seventh Iran Scientific Society of Command and Control (c4i) conference* , 2013.

Authorized licensed use limited to: Penn State University. Downloaded on March 31,2023 at 02:11:13 UTC from IEEE Xplore. Restrictions apply.