

Extractive multi document summarization using harmony search algorithm

Zuhair Hussein Ali¹, Ahmed Kawther Hussein², Haithem Kareem Abass³, Elham Fadel⁴

^{1,2,4}Department of Computer Science, College of Education, Mustansiriyah University, Baghdad, Iraq

³Department of Computer Engineering, Al-Mansour University College, Baghdad, Iraq

Article Info

Article history:

Received Feb 7, 2020

Revised Mar 22, 2020

Accepted Sep 16, 2020

Keywords:

Coverage

Diversity

Harmony search

Readability

TAC-2011

ABSTRACT

The exponential growth of information on the internet makes it troublesome for users to get valuable information. Text summarization is the process to overcome such a problem. An adequate summary must have wide coverage, high diversity, and high readability. In this article, a new method for multi-document summarization has been supposed based on a harmony search algorithm that optimizes the coverage, diversity, and readability. Concerning the benchmark dataset Text Analysis Conference (TAC-2011), the ROUGE package used to measure the effectiveness of the proposed model. The calculated results support the effectiveness of the proposed approach.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Zuhair Hussein Ali

Department of Computer Science, College of Education

Mustansiriyah University

Baghdad, Iraq

Email: zuhair72h@uomustansiriyah.edu.iq

1. INTRODUCTION

Information overload is one of the most common problems because of the fast evolution of information in the World Wide Web [1]. Text summarization (TS) is the dismissal of such a problem. Also, TS, the process of producing a document summary from a series of documents or one document without losing its main ideas, aims to extract useful information from the sources to the users [2]. The summary offers a helpful guide to generate attention on information, to make decisions on whether a document is useful or not and to assist as a time saver for users [3]. Based on the quantity of the document to be summarized, TS can be classified as a single document summarization (SDS) or multi-document summarization (MDS). For instance, in SDS just a single document can be summarized into shorter ones, while in MDS a set of related documents with the same topic is summarized into one summary [4]. MDS is more complicated than SDS although some similar techniques can be used for both MSD and SDS due to information overload and a high degree of redundancy. The redundancy occurs because summarized documents deal with similar topics and share the same ideas. As a result, reducing redundancy can lead to a high-quality summary [5].

The way of a summary creating is either extraction or an abstraction according to the function to be performed [6]. Extractive summarization is a mechanism for a professional extraction of the literary components like sentences, passages, and so on from the original meaning. Whereas, abstractive summarization will depend on the natural language processing (NLP) techniques, which need a complicated understanding of NLP strategies to analyze the sentences of documents and paragraphs where several changes have to be made

on the selected sentences. While in the extractive summarization, no need for modification will be applied to the sentences that are included in the resulted summary. Therefore, abstractive summarization is time-consuming and much difficult than extractive summarization [7]. Moreover, summarization can be classified as either generic summarization or query summarization. The Generic summarization generates a summary, which always includes the essential content of the documents. However, the restriction of generic summarization is never a topic or query is available for the summarization procedure. While in query-based summarization, a summary is created depending on the query of the user, where the documents are searched to be matched with such query [8]. This paper approach supposes a new model for extractive generic MDS based on harmony search algorithm (HSA) that improves coverage, diversity, and readability. The experimental results utilized to the TAC-2011 dataset and ROUGE package applied to measure the performance of the model.

2. RELATED WORKS

Even though text summarization has drawn consideration basically after the information expansion on the Internet, the primary work has been done in 1958 [9]. From that year a variety of summarization techniques have been proposed and assessed. For example, some researchers [10, 11] applied sentence clustering in text summarization successfully. The basic idea behind the cluster-based approach for MDS is based on sentences with high degrees of similarities that are grouped into one cluster, then one sentence is selected from each cluster to be included in the generated summary. Sentence selection depends on selecting sentences that are closest to the centroid of the cluster [12]. Graph-based approaches, which are based on an assumption that the sentence importance will increase if it has more similarity to other sentences in the document, are also used widely in MDS by the researcher. The process begins by representing each sentence as a node in the graph and the cosine similarity can be used as an edge between nodes [13].

The page rank [14] or text rank [15] is then applied to score the sentences, sentences with high scores are included in the final summary. Some researchers focus on machine learning approaches which have been commonly used in the field of TS. This approach depends on categorizing the sentences into two classes; summary sentences or non-summary sentences. Such an approach requires dividing the dataset into training, testing that data for labeling, and categorizing it accordingly. Some of the machine learning approaches are Naïve Bayes [16], neural network [17], decision trees [18] and support vector machine [19]. Many researchers have also investigated optimization approaches. Many optimization techniques such as differential evolution (DE) [20], particle swarm optimization (PSO) [21] and genetic algorithm (GA) [22] are used for TS. Optimization techniques are based on multiple agents in the population that search for candidate solutions which are considered as points in the search space. In [23] authors applied a bee colony for MDS. Here, the bees were considered as agents that search for nectar in flowers where the food is considered as a candidate solution, there is a single bee for every food source. As a result, the objective function is for the bees to collect a portion of food. When the food is abandoned, the bee then turns into a scout and looks for another food source. They search for neighboring areas and select the best candidate. When moving to a neighbor, a sentence is deleted randomly from the present summary and another sentence is included so the length limitation is not violated.

3. PROPOSED FRAMEWORK

In this paper, a new approach for MDS is proposed. It is decomposed of four main steps. First the preprocessing is done. Secondly, word similarity measure and summary quality factors are applied, and finally, harmony search is performed. These four steps are described as follows;

3.1. Preprocessing

There are four steps for preparing the data, these steps include:

- Sentence segmentation: each document is divided individually into several sentences based on the dot between them.
- Tokenization: the process of separating sentences into terms.
- Stop word removal: involves removing redundant and repeated terms in the document that do not offer the required information for recognizing an important sense of the document.
- Stemming: the process of generating the root of the word.

3.2. Similarity measure

Similarity measure plays a significant role in the field of text mining [20]. To compute the similarity between each term, they must be represented as a vector. The well-known representation scheme for terms units is the vector space model (VSM). Let $T = \{t_1, t_2, \dots, t_p\}$ represent the distinct terms that exist in the document collection D , where p is the number of terms in D . Through VSM every sentence (s_i) is represented using these terms as a vector in n -dimensional space, $s_i = \{w_{i,1}, w_{i,2}, \dots, w_{i,p}\}$, for $i=1$ to p . Each element in the vector represent a term within a given sentence. The value of each element in the vector assigns a weight using term frequency-inverse-sentence-frequency as explained shown in (1) [24].

$$W_{i,k} = TF_{i,k} * \log\left(\frac{n}{n_k}\right) \quad (1)$$

where:

$TF_{i,k}$ is the term frequency, represents how many term k appears in the sentence (S_i).

n . the number of sentences in D .

n_k . the number of sentences in which term t_k appears.

The weight $W_{i,k}$ of the term t_k should be zero if it does not exist in the sentence S_i .

The VSM requires high dimensionality of feature space that affects the performance of TS. Depending on the number of terms in each sentence the specified vector dimension p is very large and has numerous null elements, which can be a major disadvantage of VSM. The center of the document collection (o) can be calculated as the average of weights $W_{i,k}$ of term t_k for all S_i in the document collection as shown in (2) [25].

$$O_k = \frac{1}{n} \sum_{i=1}^n W_{i,k} \text{ for } k = 1 \text{ to } p \quad (2)$$

3.3. Summary of quality factors

In this section, the important factors for summary quality are demonstrated. That consists of coverage, diversity and readability. Each factor plays important role in the summarization process. These factors are described as below:

3.3.1. Coverage

The goal of TS is to cover the main content of the summarized documents by choosing subset $S \subset D$ that covers as many conceptual sentences as possible. Summary coverage can be calculated by measuring the cosine similarity between the center of document collection (O) and each sentence (S_i) as shown in (3).

$$\text{sim}(o, s_j) = \frac{\sum_{i=1}^p o_i s_i}{\sqrt{\sum_{i=1}^p (o_i)^2} * \sqrt{\sum_{i=1}^p (s_i)^2}} \text{ for } j = 1 \text{ to } n \quad (3)$$

The similarity between the center of document collection and each sentence decides the importance of the sentence and whether it is included in the generated summary [26].

3.3.2. Diversity

A summary that has a high diversity between its sentences can be considered as a good summary because its sentences solve the problem of information redundancy that occurs in most summarization models, especially in MDS. Thus, to achieve an adequate summary, the sentences should have a high diversity among them. Summary diversity is computed by considering the total value of sentence similarity. A good summary is associated with lower diversity values that ensure minimum information redundancy. As shows in (4) the formulation to compute sentences diversity [27].

$$\text{diversity}(S_i) = \sum_{i=1}^{n-1} \sum_{j=i+1}^n \text{sim}(S_i, S_j) \quad (4)$$

3.3.3. Readability

Readability is an important factor for document summary that indicates the sentences in the summary are highly related to the next sentence in the document summary. The readability (R_s) of summary (s) with length (S) can be formulated as shown in (5) and (6) respectively [28].

$$R_s = \sum_{0 \leq i \leq s} \text{sim}(S_i, S_{i+1}) \quad (5)$$

$$RF_s = \frac{R_s}{\text{Max}_{v_i \times R_i}} \quad (6)$$

The objective function is to maximize the three factors coverage, diversity and readability as shown in (7).

$$F(s) = fcov(s) + fdiv(s) + fred(s) \quad (7)$$

3.4. Harmony search based MDS

Harmony search algorithm (HSA) is a meta-heuristic algorithm that was developed by Z. W. Green, *et al.* in 2001 [29]. HSA requires less mathematical operations and can be easily used in many optimization problems compared to other meta-heuristic algorithms. HSA algorithm tries to search for a global solution specified by the objective function. The decision variables assign values to determine the objective function, is similar to tones of musical instruments that decide the aesthetic quality. Thus, the HSA algorithm works similarly to a musician who is looking for the best harmony [30].

The harmony vector values are stored in the harmony memory (HM) matrix as follows;

$$HM = \begin{bmatrix} X_1^1 & X_2^1 & X_n^1 \\ \vdots & \vdots & \vdots \\ X_1^{HMS} & X_2^{HMS} & X_n^{HMS} \end{bmatrix}$$

where $[x_1^i, x_2^i, \dots, x_n^i]$ is a candidate solution. The HM is initialized by random variables. Also, two important parameters that should be initialized are Harmony memory considering rate (HMCR) and pitch adjusting rate (PAR). These two parameters are updated by harmony memory consideration (HMC) and pitch adjusting (PA) respectively. The HMCR plays an important role in selecting a value from memory while the PA is important for both exploitation and exploration. The exploitation is used to find optimal solutions, whereas exploration is used to avoid local minima [31]. The following algorithm shows how HSA is used for text summarization.

- Step1: collect a set of multiple documents $D = \{D_1, D_2, \dots, D_N\}$ where each D_i represent individual document
- Step2: apply preprocessing steps to each D_i
- Step3: for each D_i calculate the coverage as shown in (3)
- Step4: for each D_i calculate the diversity as shown in (4)
- Step5: for each D_i calculate the readability as shown in (6)
- Step6: initialize the HM with random solutions and also initialize HMCR, PAR
- Step7: sort the entire solution of HM and rank them according to shown in (7)
- Step8: improves a new solution from HM as follows;
 - a. if $\text{rand}(0.1) < \text{HMCR}$ then choose new solution from HM
 - else choose a solution randomly.
 - b. if $\text{rand}(0.1) < \text{PAR}$ then choosing an adjacent value of the selected value to depend on bandwidth.
- Step9: if the new solution is better than worst stored {based shown in (7)} one then update the HM by the new solution
- Else eliminate the new solution
- Step10: check stop condition if the result be in a stable state then end
- Else go to step 7

4. DATASET AND EVALUATION METRICS

TAC-2011 dataset was used to test the system performance. The dataset consists of seven languages (English, Arabic, Greek, Czech, French, Hindi, Hebrew). There are 10 topics, each of 10 documents for each language [32]. The proposed model deals with the English language only. Recall-oriented understudy for gisting evaluation (ROUGE) [33] was used to evaluate the proposed system the outputs of a rouge package are three numbers which represent Precision, Recall, and F-score. They are formulated as follows.

$$\text{Precision} = \frac{\text{systemsummarysentences} \cap \text{idealsummarysentences}}{\text{numberofsentencesinthesystemsummary}} \quad (8)$$

$$\text{Recall} = \frac{\text{systemsummarysentences} \cap \text{idealsummarysentences}}{\text{numberofsentencesintheidealsummary}} \quad (9)$$

$$F - \text{score} = \frac{2 * \text{precision} * \text{recall}}{\text{precision} + \text{recall}} \quad (10)$$

5. RESULTS AND DISCUSSION

ROUGE-1 and ROUGE-2 matrices have been used to measure the performance of the summary. These matrices are very similar to human judgment. The summary performance is measured by computing the overlap between system summaries with human summaries. ROUGE-1 is concerned with computing unigrams overlaps while ROUGE-2 is concerned with computing bigrams overlaps. The results are compared with the results of [12] that included peer summaries in the TAC-2011 data set. Tables from 1 to 4 show the results of the proposed model [12] using ROUGE-1 and ROUGE-2 respectively.

As seen from Tables 1 and 2, compared to the result of [12], using ROUGE-1. The tables show the recall and F-score of the proposed model are higher. However, the precision is lower. The Judgment between the recall and the precision is the F-score that consider them both. As known, the precision is computed by dividing the number of sentences overlap between system summary and ideal summary by the number of sentences in the system summary. Whereas the recall is computed by dividing the number of sentences overlap between system summary and ideal summary by the number of sentences in the ideal summary. Thus, by increasing the number of words in the system summary leads to decreasing precision. While decreasing the number of words in the system summary leads to decreasing the recall. The length of each ideal summary between 240-250 words, while the length of each individual generated summary is more than 250, the reason behind the length of the generated summary is the mechanism of creation that is based on adding sentences to the summary without any change to the length of the sentence. Which causes the summary length of more than 250 words, especially when the last sentence is too long.

Table 1. Comparison between the proposed model and [12] results using ROUGE-1

Model Topic	Proposed model Results			[12] Results		
	Precision	Recall	F-Score	Precision	Recall	F-Score
ID1	0.37200	0.48818	0.42224	0.41253	0.40524	0.40776
ID2	0.39220	0.52223	0.44797	0.45655	0.46481	0.46062
ID3	0.41266	0.56887	0.47833	0.47909	0.43169	0.45404
ID4	0.40216	0.56634	0.47033	0.44966	0.44423	0.44691
ID5	0.41213	0.61686	0.49412	0.43513	0.41092	0.42243
ID6	0.37625	0.46825	0.41723	0.45122	0.3547	0.39617
ID7	0.35321	0.55367	0.43128	0.3953	0.39586	0.39547
ID8	0.42142	0.69583	0.52492	0.39265	0.38714	0.38985
ID9	0.37251	0.55123	0.44458	0.37726	0.38105	0.37912
ID10	0.43711	0.60251	0.50665	0.51806	0.52488	0.52141

Table 2. Average precision, recall and F-score using ROUGE-1

Model	Precision	Recall	F-Score
Proposed model	0.395165	0.563396	0.46376
[12]	0.436745	0.420052	0.427376

Tables 3 and 4 show the results of the proposed model using ROUGE-2. The efficiency of the proposed model was evident when it was used ROUGE-2 because ROUGE-2 is closer to human summary than ROUGE-1. The Average of the three metrics recall, precision and F-score are better than [12]. This is because of the good definitions of coverage, diversity, and readability and due to the good performance of HSA in regards to choosing the most suitable sentences to be included in the final summary.

Table 3. Comparison between proposed model and [12] results using ROUGE-2

Model Topic	Proposed model Results			[12] Results		
	Precision	Recall	F-Score	Precision	Recall	F-Score
ID1	0.24557	0.17591	0.20498	0.12448	0.12125	0.12247
ID2	0.14211	0.17829	0.15815	0.16779	0.17052	0.16914
ID3	0.19230	0.26710	0.22361	0.19256	0.1733	0.18237
ID4	0.14393	0.26182	0.18574	0.15369	0.1517	0.15269
ID5	0.17256	0.32551	0.22555	0.14404	0.13605	0.13985
ID6	0.09323	0.13967	0.11181	0.1367	0.10655	0.11937
ID7	0.13855	0.20151	0.16420	0.09612	0.09662	0.09635
ID8	0.22213	0.36548	0.27631	0.12298	0.12144	0.12219
ID9	0.20374	0.14912	0.17220	0.10841	0.10962	0.109
ID10	0.19303	0.27252	0.22598	0.2483	0.25177	0.25

Table 4. Average precision, recall and F-score using ROUGE-2

Model	Precision	Recall	F-Score
Proposed model	0.193524	0.290582	0.231984
[12]	0.149507	0.143882	0.146342

6. CONCLUSION

The need for influential MDS approaches to extract significant information from a document collection becomes of necessity. This paper used HSA based MDS to create a generic extractive summary. The Summarizer used a benchmark dataset called TAC-2011, and ROUGE package was applied to evaluate the performance of the summarizer. The proposed model is based on three important issues in MDS that include coverage, diversity, and readability. Good results were obtained from the proposed model. The limitation of this method is controlling the parameters of HMCR and PAR that require special treatment.

ACKNOWLEDGEMENTS

The author would like to thank Mustansiriyah University (www.uomustansiriyah.edu.iq) Baghdad-Iraq for its support in the present work. Special thanks to Miss Mawada Al Faisal for her valuable scientific advice.

REFERENCES

- [1] A. Ahmad, T. Ahmad, "A game theory approach for multi-document summarization," *Arabian Journal for Science and Engineering*, vol. 44, pp. 3655-3667, 2019.
- [2] R. M. Alguliev, R. M. Aliguliyev, N. R. Isazade, "Multiple documents summarization based on the evolutionary optimization algorithm," *Expert Systems with Applications*, vol. 40, no. 5, pp. 1675-1689, April 2013.
- [3] J. M. Sanchez-Gomez, *et al.*, "Extractive multi-document text summarization using a multi-objective artificial bee colony optimization approach," *Knowledge-Based Systems*, vol. 159, pp. 1-8, November 2018.
- [4] R. M. Alguliev, *et al.*, "DESAMC+ DocSum: Differential evolution with self-adaptive mutation and crossover parameters for multi-document summarization," *Knowledge-Based Systems*, vol. 36, pp.21-38, December 2012.
- [5] M. Gambhir, V. Gupta, "Recent automatic text summarization techniques: a survey," *Artificial Intelligence Review*, vol. 47, pp. 1-66, 2017.
- [6] R. Ferreira, *et al.*, "Assessing sentence scoring techniques for extractive text summarization," *Expert systems with applications*, vol.40, pp. 5755-5764, October 2013.
- [7] M. Yousefi-Azar, L. Hamey, "Text summarization using unsupervised deep learning," *Expert Systems with Applications*, vol. 68, pp. 93-105, February 2017.
- [8] R. Aliguliyev, *et al.*, "A model for text summarization," *International Journal of Intelligent Information Technologies*, vol. 13, no. 1, pp. 67-85, January 2017.
- [9] H. P. Luhn, "The automatic creation of literature abstracts," *IBM Journal of research and development*, vol. 2, no. 2, pp.159-165, April 1958.
- [10] D. Wang, T. Li, "Document update summarization using incremental hierarchical clustering," *Proceedings of the 19th ACM international conference on Information and knowledge management*, pp. 279-288, January 2010.
- [11] D. Wang, *et al.*, "Integrating document clustering and multi document summarization," *ACM Transactions on Knowledge Discovery from Data (TKDD)*, vol. 5, no.3, pp. 1-26, August 2011.
- [12] M. El-Haj, U. Kruschwitz, C. Fox, "Multi-document arabic text summarization," *Conference: In the 3rd Computer science and Electronic Engineering Conference (CEEC'11)*, July 2011.
- [13] K. S. Thakkar, *et al.*, "Graph-based algorithms for text summarization," *2010 3rd International Conference on Emerging Trends in Engineering and Technology*, pp. 516-519, November 2010.
- [14] S. Brin, L. Page, "The anatomy of a large-scale hypertextual web search engine," *Computer networks and ISDN Systems*, vol. 30, no. 1-7, pp.107-117, April 1998.
- [15] R. Mihalcea, P. Tarau, "Textrank: bringing order into text," *Proceedings of the 2004 conference on empirical methods in natural language processing*, pp. 404-411, July 2004.
- [16] H. N. T. Thu, "An optimization text summarization method based on naive Bayes and topic word for single syllable language," *Applied Mathematical Sciences*, vol. 8, pp. 99-115, January 2014.
- [17] K. Svore, *et al.*, "Enhancing single-document summarization by combining RankNet and third-party sources," *Proceedings of the 2007 joint conference on empirical methods in natural language processing and computational natural language learning (EMNLP-CoNLL)*, pp. 448-457, January 2007.
- [18] P. M. Sabuna, D. B. Setyohadi, "Summarizing Indonesian text automatically by using sentence scoring and decision tree," *2017 2nd International conferences on Information Technology, Information Systems and Electrical Engineering (ICITISEE)*, pp. 1-6, November 2017.

- [19] A. Fanani, *et al.*, "Regression model focused on query for multi documents summarization based on significance of the sentence position," *TELKOMNIKA Telecommunication Computing Electronics and Control*, vol. 17, no.6, pp. 3050-3056, December 2019.
- [20] R. M. Alguliev, *et al.*, "Sentence selection for generic document summarization using an adaptive differential evolution algorithm," *Swarm and Evolutionary Computation*, vol. 1, no. 1, pp. 213-222, December 2011.
- [21] R. Rautray, *et al.*, "Document summarization using sentence features," *International Journal of Information Retrieval Research (IJIRR)*, vol. 5, no. 1, pp. 36-47, January 2015.
- [22] M. A. Fattah and F. Ren, "GA, MR, FFNN, PNN and GMM based models for automatic text summarization," *Computer Speech & Language*, vol. 23, no. 1, pp. 126-144, January 2009.
- [23] M. Peyrard, J. Eckle-Kohler, "A general optimization framework for multi-document summarization using genetic algorithms and swarm intelligence," *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pp. 247-257, 2016.
- [24] R. M. Alguliev, *et al.*, "CDDS: Constraint-driven document summarization models," *Expert Systems with Applications*, vol. 40, no. 2, pp.458-465, February 2013.
- [25] R. Rautray, R. C. Balabantaray, "An evolutionary framework for multi document summarization using Cuckoo search approach: MDSCSA," *Applied computing and informatics*, vol. 14, no. 2, pp.134-144, July 2018.
- [26] R. Rautray, R. C. Balabantaray, "Cat swarm optimization based evolutionary framework for multi document summarization," *Physica A: Statistical Mechanics and its Applications*, vol. 477, pp.174-186, February 2017.
- [27] K. Umam, *et al.*, "Coverage diversity, and coherence optimization for multi-document summarization," *Jurnal Ilmu Komputer dan Informasi*, vol. 8, no. 1, pp.1-10, 2015.
- [28] S. H. Mirshojaei, and B. Masoomi, "Text summarization using cuckoo search optimization algorithm," *Journal of Computer&Robotics*, vol. 8, pp.19-24, 2015.
- [29] S. H. Mirshojaei, *et al.*, "A new heuristic optimization algorithm: harmony search," *Simulation*, vol. 76, pp. 60-68, 2001.
- [30] B. Wu, *et al.*, "Hybrid harmony search and artificial bee colony algorithm for global optimization problems," *Computers&Mathematics with Applications*, vol. 64, no. 8, pp. 2621-2634, October 2012.
- [31] S. Abed, *et al.*, "Harmony search algorithm for word sense disambiguation," *PloS one*, vol. 10, 2015.
- [32] G. Giannakopoulos, *et al.*, "TAC 2011 MultiLing pilot overview," *Conference: Text Analysis Conference (TAC 2011), MultiLing Summarisation Pilot*, January 2011.
- [33] C. Y. Lin, "Rouge: A package for automatic evaluation of summaries," *Text summarization branches out*, pp. 74-81, 2004.