





BBMA-MDS: Binary Biology Migration Algorithm for Multi-Document Text Summarization

Mohamed Boussalem^{1,2}, Samia Aitouche³, Moumen Hamouma^{1*}, Hichem Haouassi², Hichem Rahab²,
Abdelaali Bekhouche²

¹ Department of Computer Science, University of Batna 2, Batna 05000, Algeria

² Department of Computer Science and Mathematics, ICOSI Laboratory, Abbes Laghrour University, Khenchela 40004, Algeria

³ Department of Industrial Engineering, Laboratory of Automation and Manufacturing (LAP, Laboratoire d'Automatique et de Productique), University Batna 2, Batna 05000, Algeria

Corresponding Author Email: hamouma.moumen@univ-batna2.dz

<https://doi.org/10.18280/ria.370506>

ABSTRACT

Received: 21 March 2023

Revised: 24 July 2023

Accepted: 31 July 2023

Available online: 31 October 2023

Keywords:

text summarization, multi document text summarization, optimization, binary biology migration algorithm, swarm intelligence, ROUGE metrics

As the World Wide Web continues to expand, the process of identifying pertinent information within its vast volume of documents becomes increasingly challenging. This complexity necessitates the development of efficient solutions, one of which is automatic text summarization; an active research area dedicated to extracting key information from extensive text. The difficulties are further compounded when addressing multi-document text summarization, due to the diversity of topics and sheer volume of information. In response to this issue, this study introduces a novel approach, the Binary Biology Migration Algorithm for Multi-Document Summarization (BBMA-MDS). Viewing multi-document summarization as a combinatorial optimization problem, this approach leverages the biology migration algorithm to select an optimal combination of sentences. Evaluations of the proposed algorithm's performance are conducted using the ROUGE metrics, which facilitate a comparison between the automatically generated summary and the reference summary, commonly known as the 'gold standard summary'. For a comprehensive evaluation, the well-established DUC2002 and DUC2004 datasets are employed. The results demonstrate the superior performance of the BBMA-MDS approach when compared to alternative algorithms, including firefly and particle swarm optimization, as indicated by the selected metrics. This study thus contributes to the field by proposing BBMA-MDS as an effective solution for the multi-document text summarization problem

1. INTRODUCTION

The vast amount of information and documents accessible on the internet, especially in textual form makes finding relevant information, a difficult task. By relevance, we intend here the user requirements in terms of searched information [1]. As a solution to this problem, automatic text summarization (ATS) approaches have appeared. The process of ATS aims to produce a short text version of the original document [2], containing the crucial information. The produced text is called a summary.

The summary may be extractive or abstractive, depending on the approaches in which it is generated [3, 4]. Extractive approaches [5-7] generate a summary by selecting the pertinent sentences from the source text. Whereas, abstractive approaches [8, 9] generate the summary using new words or sentences that are not necessary present in the original text. For the aforementioned reason, extractive approaches are the most commonly used in the literature. In addition, Extractive approaches frequently show better results compared to abstractive ones [10]. In the study [11, 12], a hybrid technique is used to combine the extractive and the abstractive approaches. Considering the objective, the summary can be generic or query-oriented [4, 13]. The generic summary [14,

15] is generated based on the main content of the original document without using any additional information. But, the query-oriented summary [16] is based on the information contained in the given query. The summary can be divided into single-document and multi-document summaries considering the number of documents to be summarized [3, 13]. Multi-Document Summarization (MDS) is more challenging than single-document summarization [17], due to the size of the documents and the multiplicity of topics. MDS is classified as an NP-hard problem due to the tremendous amount of texts and the vast number of sentence combinations that need to be considered and selected in the summary.

Text summarization employs several techniques, including statistical approaches [5], graph-based approaches [18], topic-based approaches [19], machine learning-based approaches [14, 20, 21], as well as metaheuristic approaches [22, 23].

The current methods of multi-document text summarization have limitations that hinder their ability to produce highly effective and accurate summaries. These limitations include; scalability issues, difficulty in content selection, and challenges in maintaining coherence and consistency across the summary, especially when dealing with diverse sources of information. Due to these limitations, there is a need for more advanced algorithms like the Biology Migration Algorithm

(BMA) [24].

To the best of our knowledge, the BMA algorithm is never used in MDS problem before. In our case, we used differently the BMA, which is a metaheuristic algorithm, to address the MDS problem by striking an effective balance between exploration and exploitation strategies. In addition, we proposed a novel fitness function, composed of three factors; coverage, cohesion, and readability, enhancing the summary's quality by promoting coherence and ensuring the selection of relevant content.

Our approach uses the extractive method to create a summary; therefore, the summary is a sub-set of sentences selected from the original collection of documents. This makes the MDS problem a binary optimization problem. Thus, we introduced a binarization step for positions of particles to adapt the original algorithm with the binary nature of MDS. The sigmoid function is adapted to BBMA-MDS to convert the continuous positions to binary ones.

The proposed approach is evaluated on DUC2002 and DUC2004 datasets, then, compared to the performance of other algorithms like: firefly, particle swarm optimization (PSO). The obtained results show that it performs better than other ones in ROUGE metrics.

The remainder of this paper is organized as follows: The related work of MDS approaches is presented in Section 2. Section 3 includes the MDS problem formulation and the details of the original BMA. After that, the proposed approach (BBMA-MDS) is presented in Section 4 followed by the results of the experiments in Section 5. In the Section 6, the performance of our approach is compared to other works of the literature. Finally, Section 7 concludes the paper with perspectives on future works.

2. RELATED WORK

In the context of Multiple Document Summarization (MDS), metaheuristic approaches treat the summarization task as an optimization problem. This section offers an overview of the existing literature related to these metaheuristic-based methods.

In the study [22], the MDS problem is solved in two stages, in the first stage, the authors create a single document from the initial collection of documents, where similar sentences are removed to minimize the redundancy. Maximum coverage of the topic is considered to select the relevant sentences. The second stage is the summary generation, where the process is modeled as a shark smell optimization (SSO) problem. The experiments are performed on MultiLing13, TAC08, TAC11, DUC04, DUC06 and DUC07 datasets. The results show that the proposed system has better performance than the compared works in term of ROUGE-1 and ROUGE-2. The PSO algorithm is also used in step two in the experimentation phase.

In the study [25], a fuzzy evolutionary optimization model (FEOM) is introduced for MDS. FEOM performs document clustering then selects the most pertinent sentences for each cluster to generate a summary. The model utilizes genetic algorithms to generate solution vectors for the groups and incorporates three control parameters to regulate the probability of crossover and mutation for each solution.

A Cat Swarm Optimization algorithm (CSO) is used to create a Multi-document summary [15]. Documents are represented by TF-IDF representation to calculate sentences' informativeness and then the inter-sentences similarity is

calculate using the cosine similarity. The quality of the generated summary is measured using; the contents coverage, readability and cohesion. The datasets used in the experimentations are DUC2006 and DUC 2007. In the experimental study, the authors compare the performance of CSO against harmony search (HS) and PSO algorithms. The comparison shows that the CSO algorithm gives better results than the other two algorithms.

A new summarizer based on cuckoo search algorithm is proposed for resolving multi-document summarization problems is proposed [26]. In their objective function conception, the authors try to cover a set of objectives when building the summaries. The considered objectives include; readability, cohesion and non-redundancy. In their experiment, DUC 2006 and DUC 2007 dataset have been used. The proposed approach is compared with state of art summarization algorithms i.e., CSO and PSO using ROUGE-N as ROUGE-1 and ROUGE-2, readability and sentence similarity metrics. From three observations, the proposed cuckoo search algorithm based approach performs better, in most cases, when compared to CSO and PSO.

In the study [23], the improved Cuckoo Search Optimization Algorithm (ICSA) is proposed to resolve MDS issue. The intensification strategy is enhanced by the inclusion of the simulated annealing and orthogonal concepts. Cohesion, readability and coverage are adopted to evaluate the obtained summary quality. Two benchmarked corpora were included in the experiments; DUC2007 and DUC2006. The proposed approach, when compared with other similar works, achieves better results with 0.09582 of F-measure, 0.10010 of recall and 0.09387 of precision.

A Multi-document Summarization approach based on the algorithm Social Spider Optimization (SSO) is proposed in the study [27]. The experimental study is conducted on two datasets namely, DUC2006 and DUC2007. The adopted evaluation metrics include ROUGE-1 and ROUGE-2 in addition to accuracy and F-measure. The performance of the proposed approach is compared with similar summarization works which use PSO and harmony search algorithms. The evaluation results indicate that the SSO-based summarization approach outperforms both PSO and harmony search algorithms in terms of ROUGE-1, ROUGE-2, accuracy and F-score metrics. From the used methods, PSO-based summarization performs the worst in terms of the different used metrics.

In the research [28], a new MDS approach based on the firefly algorithm is proposed; to measure the quality of the resultant summary, the authors used a new fitness function composed of three factors; topic relation factor, cohesion factor and readability factor. The benchmark datasets used in the experiments are DUC2002, DUC2003 and DUC2004. The performance of the proposed summarization approach is compared with other summarization works which use nature-inspired algorithms such as genetic algorithm (GA) and PSO. The evaluation criteria used in the experiments are the ROUGE score. The results show that this approach is more efficient than other approaches.

The previously mentioned studies all share the goal of producing high-quality summaries in terms of content coverage, readability, cohesion, and non-redundancy. However, they differ in their approaches to summarization. In the next paragraph, we will highlight the distinct characteristics of each study.

The first notable distinction among these works is the

choice of a specific metaheuristic algorithm for the summarization task. Each study adopts a different algorithm to optimize the summarization process. The second difference lies in the fitness functions used to assess the quality of the generated summaries. Each work employs its own set of criteria and weights to evaluate the effectiveness of the summarization. Furthermore, each study follows a unique process to generate the summary. For example, in the study [22], the authors begin by eliminating similar sentences to reduce the text's size and then apply the SSO algorithm to create the summary. In contrast, in the study [25], the approach clusters the documents based on their topics and then selects the most relevant sentences from each cluster for the summary. Similarly, in the study [15], the focus is on selecting the least similar sentences among document clusters to reduce redundancy, followed by using the CSO algorithm to generate the final summary.

The overall findings from the related works show that different metaheuristic approaches contribute significantly to the field of multi-document summarization. These approaches offer enhanced optimization capabilities, allowing for efficient exploration of the vast solution space to generate high-quality summaries. By considering multiple objectives, such as coverage, cohesion, and readability, the approaches produce well-balanced and informative summaries.

The prevailing trends in the field include the emergence of hybrid approaches that combine multiple metaheuristic algorithms or integrate them with other techniques like machine learning. Novel fitness functions are being designed to better capture the desired characteristics of summaries, and diverse and challenging datasets are being used for evaluation. Real-world applications are also gaining attention, as researchers explore domain-specific summarization and personalized summarization to cater to individual user preferences.

3. PROBLEM FORMULATION AND ORIGINAL BIOLOGY MIGRATION ALGORITHM (BMA)

3.1 Problem formulation

The MDS is the process of automatically selecting the most important sentences from the original documents to create a compressed version, which provides useful information for the reader. In this paper, we aim to consider the MDS as a combinatorial optimization problem and proposing a solution based on a metaheuristic algorithm.

The input of our approach is a collection of documents;

$CD = \{D_1, D_2, \dots, D_n\}$, where; D_i is i^{th} document and n denotes the number of documents in the collection. Each document D_i is composed of a set of sentences $D_i = \{s_{i1}, s_{i2}, s_{i3}, \dots, s_{im}\}$. The sets of sentences of documents are fused to obtain the final set $CD = \{s_1, s_2, s_3, \dots, s_D\}$, where; D denotes the number of sentences in CD . The output is a summary Sum from the input CD , where Sum is a subset of sentences selected from the original CD with a size under a fixed threshold T of number of words, as shown in Eq. (1).

$$Sum = \{s_1, s_2, \dots, s_j\} / s_i \in CD, \text{Size}(Sum) \leq T \quad (1)$$

There are a huge number of combinations of sentences (summary or solution) that satisfy Eq. (1) called feasible solutions and presented in Eq. (2).

$$\text{Feasible solutions } (Sum_{candidate}) = \{Sum_1, Sum_2, \dots, Sum_k\} \quad (2)$$

K is the number of all possible solutions, and Sum_i is given in Eq. (1).

Our objective is to find the best solution among the feasible solutions. The best summary is the one that has the high quality expressed by a fitness function, explained as follows.

3.1.1 Quality of the summary

The quality of a summary depends on three criteria; coverage, cohesion and readability. Which are normalised between 0 and 1. These criteria are detailed below.

The coverage is presented in Eq. (3), it represents the content coverage of all documents in CD . In other words, it checks if the summary takes in account the content of all documents. The summary Sum_i with the highest coverage is considered to be the best summary.

$$Coverage = \frac{\sum_{i=1}^n \sum_{j=1}^N \frac{sim(s_i, s_j)}{N-1}}{N-1} \quad (3)$$

N is the number of sentences in CD and n represents the number of sentences in the summary.

The cohesion represents the connection between sentences in the summary Sum_i , as presented in Eq. (4); taking in account this factor; a good summary is expected to have a high cohesion value.

$$Cohesion = \frac{\sum_{i,j=1}^n (1 - sim(s_i, s_j))}{(n * (n - 1) / 2)} \quad (4)$$

where n gives the number of sentences in the summary, $i, j = 1 \dots n$, and $j \geq i$.

In a readable summary, each sentence should be related to the sentence following it. Readability is given by the Eq. (5):

$$Readability = \frac{\sum_{i=1}^n sim(s_i, s_j)}{n-1} \quad (5)$$

where n is the number of sentences in the summary; s_i, s_j are two consecutive sentences in the summary.

In what follows, we use these three criteria in a single formula to express the fitness function that measures the quality of the summary.

3.1.2 Fitness function

The fitness function expresses the quality of the summary by the formula (6). The used fitness function is a weighted sum of the aforementioned three criteria; coverage, cohesion and readability given in Eqs. (3), (4), and (5) respectively.

$$Fitness = \alpha * Coverage + \beta * Cohesion + \gamma * Readability \quad (6)$$

where $\alpha, \beta, \gamma \in [0,1]$, and $\alpha + \beta + \gamma = 1$.

A good summary has a high combination value between coverage, connection between sentences, readability and cohesion. The approach searches for the optimal solution which has the maximum value of the fitness function.

Then, the MDS optimization problem can be formulated as follow:

Maximize (Fitness (Sum))

where:

$$\begin{cases} \text{Sum is a summary of multi documents CD} \\ \text{CD is the input multi documents to be summarized} \\ \text{Size(Sum)} \leq T \end{cases}$$

3.2 Original Biology Migration Algorithm (BMA)

Inspired by the biological migration phenomenon of different species such as insects, mammals, fish and others, Zhang et al. [24] proposed recently a new swarm-based optimization algorithm, called biology migration algorithm.

In BMA, the biological species (particles) represent the population that searches for solutions to the problem in their habitat (search space). As all swarm-based optimization algorithms, BMA starts the optimization process with a randomly initialised population as N D-dimensional real vectors in the range $[0, 1]$, where N is the population size.

After the initialization, particles displace in the search space by generating a new generation at each iteration until a maximum number of iterations is met. The search process of BMA is processed in two main phases: migration phase and updating phase.

3.2.1 Migration phase

In this phase, species modify their position (they move from their current position toward a new position) according to two alternatives: On one hand, the best specie of the population as presented in Eqs. (7) and (8), and on another hand, its neighbourhood candidates as presented in Eq. (9). The neighbourhood candidates of each particle are the two randomly selected particles from the population. Then, the migration phase can be mathematically given by Eqs. (7), (8) and (9).

$$X_i(t+1) = X_i(t) + \lambda * \sec(t) * L(t) * |X_{best} - X_i(t)| \quad (7)$$

where, $X_i(t)$ and $X_i(t+1)$ are the positions of the i^{th} particle at iterations t and $t+1$ respectively, λ is a random vector in the range $[0, 1]$, $L(t)$ is the step size defined in the Eq. (8), X_{best} is the best position in the current iteration.

$$L(t) = 2 - 1.7 \left(\frac{t-1}{T-1} \right) \quad (8)$$

where, t is the current iteration and T is the maximum number of iterations.

$$X_i(t+1) = X_i(t) + \zeta * (X_j(t) - X_k(t)), i \neq j \neq k \quad (9)$$

where, $X_i(t)$ and $X_i(t+1)$ the positions of the i^{th} particle at iterations t and $t+1$ respectively, $X_j(t)$ and $X_k(t)$ are two positions selected randomly from the population, ζ is a random vector in the range $[0, 1]$

3.2.2 Updating phase

In the updating phase, the current position $x_i(t)$ and the new obtained position by applying the migration phase $x_i(t+1)$ are compared. If the new position $x_i(t+1)$ cannot improve the quality of the current solution within a predetermined number of cycles C , it will be abandoned and replaced by a new randomly generated position to explore other search zones.

4. PROPOSED MDS APPROACH

The MDS problem is known as an NP-hard problem because of the big size of texts and a large number of sentence combinations of sentences to be selected in the summary. Therefore, we need a long time to discover an optimal summary. It is possible to obtain a near-optimal summary in a reasonable time for such problems by using metaheuristic methods.

One of the advantages of metaheuristic algorithms is that they are problem-independent and have a good approach to solve problems in different domains [29, 30].

There are a variety of meta-heuristic algorithms in the literature. One of the metaheuristic algorithms that have been introduced recently is the BMA that is used to solve complex computational problems [24]. BMA was successfully used for resolving the feature selection problem [31]. Also, the work [32] uses BMA for unmanned aerial vehicles flight route optimization problem.

In this section, we present in detail the proposed BBMA-MDS Approach. The proposed approach is named BBMA-MDS. The main steps of BBMA-MDS are showed in the flowchart of Figure 1.

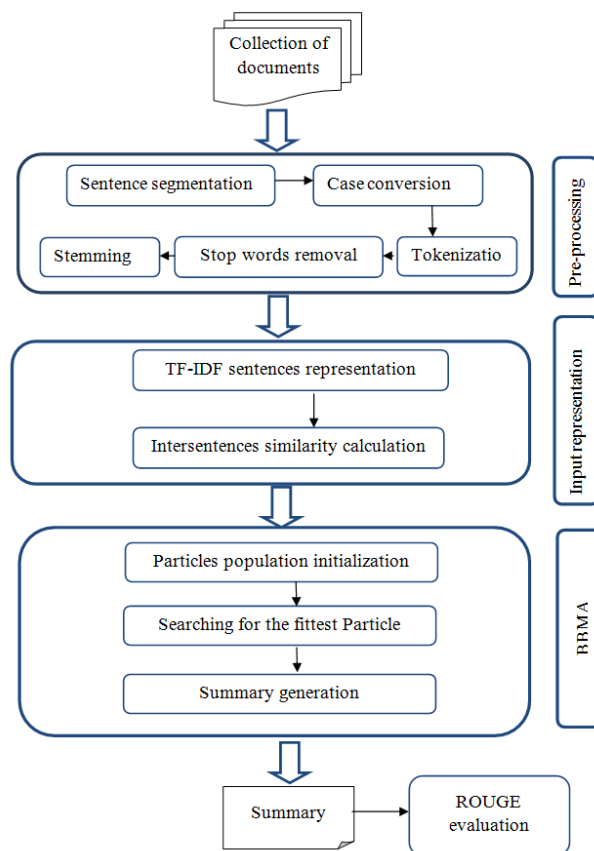


Figure 1. Main steps of BBMA-MDS

4.1 Pre-processing

Pre-processing is an indispensable phase for every text summarization process. The following treatments are performed in this work.

4.1.1 Sentence segmentation

For each document D in the collection of documents to be summarized, the text is segmented into sentences. The result

of this treatment is: $D=\{s_1, s_2, s_3, \dots, s_n\}$, where s_i represents the i^{th} sentence in the document D ; n is the number of sentences in document D .

4.1.2 Case conversion

The text in documents should be in the same form, lower-case or upper-case. This unification helps in reducing the vector space, which allows the optimization of computing power and time. In our study the lower-case is considered.

4.1.3 Tokenization

Tokenization is an important step for preparing documents for the step of removing stop words and stemming, this phase consists of dividing the sentences into tokens. The result of this step is; $s = \{t_1, t_2, t_3 \dots t_k\}$, where t_j represents the j^{th} token of the sentence s , and k is the number of tokens in the sentence s .

4.1.4 Stop words removal

The objective of this step is to exclude stop words such as "a," "the," "then," "before," and others, as they are commonly used but do not hold significant importance for the summarization task.

4.1.5 Stemming

In this process, words that come from the same root, are identified and grouped into a single group and are replaced with their root.

4.2 Input representation

4.2.1 Sentences representation

TF-IDF stands for Term Frequency-Inverse Document Frequency; it is the most widely used method for representing text in digital form. The input of BBMA-MDS is a set of sentences of all documents in the collection; each sentence is represented in TF-IDF form, where:

The weights of terms of each sentence in TF-IDF representation are given in formula (10).

$$W_{ij} = TF_{ij} * IDF_i \quad (10)$$

TF_{ij} represents term frequency of the term t_i in sentence s_j , it's given in formula (11).

$$TF_{ij} = \frac{freq_{ij}}{\|s_j\|} \quad (11)$$

where, $freq_{ij}$ are the occurrences number of the term t_i in the sentence s_j .

$\|s_j\|$ is the number of all terms in the sentence s_j .

Inverse Document Frequency (IDF), see Eq. (12), is the relation between the total number of sentences N in the collection and the number of sentences n_i containing the term t_i .

$$IDF_i = \log_{10} (N/n_i) \quad (12)$$

4.2.2 Inter-sentences similarity

After having represented sentences in TF-IDF form in Equation (10), we calculate the similarity between sentences using the following cosine similarity formula.

$$sim(s_i, s_j) = \frac{\sum_{k=1}^m W_{ik} * W_{jk}}{\sqrt{\sum_{k=1}^m W_{ik}^2} * \sqrt{\sum_{k=1}^m W_{jk}^2}} \quad (13)$$

The Eq. (13) is used in BBMA-MDS to calculate the fitness function.

4.3 Proposed Binary Biology Migration Algorithm (BBMA)

The MDS problem is binary in nature, while the original BMA has been designed for continuous problems. Therefore, the BMA requires a binarization process to adapt it to binary problems. Two main methods are used in the literature for metaheuristics binarization: transfer functions and the modified position equation method using new binary operators [33]. In the first method, the metaheuristic operates in continuous space and when we need to evaluate each particle's position a transfer function is applied to map the continuous vector to a binary one. In the second method, all operators in the position updating equations of the algorithm are transformed to binary operators using several technics, and then the particles are initialised and displaced in the binary space. The use of transfer function is widely used in binarization process due to their promising results obtained in solving many optimization problems before [34-36]. Hence, in this work, the transfer function binarization technic is used to propose a binary version of the BMA algorithm.

4.3.1 Solution encoding and population initialization

In the proposed approach for the MDS problem, a summary is a combination of a sub-set of sentences selected from the original CD . This problem is known as a combinatorial optimization problem which is of exponential complexity, what we motivated to use metaheuristics because are more adapted to these complex problems.

The algorithm starts with a set of solutions called initial particle's positions $P = (p_1, p_2, \dots, p_n)$, where N is the number of particles (swarm size) and p_i is the i^{th} particle in the swarm. Each particle p_i has a position p_i^t at a time t that represents a solution of the optimized problem. In the MDS problem, a solution (summary) is represented as S -sized binary vector, where S is the total number of sentences in the original documents, the $p_i^t[j]$ is the j^{th} element in the vector p_i^t . As our problem is to select or not select sentences in the summary (solution), each element $p_i^t[j]$ can take a binary value "zero" or "one"; the value "zero" means that the corresponding sentence (j^{th} sentence) is not selected in the solution, and the value "one" means that the corresponding sentence is selected in the summary. For example, in Figure 2, a particle's position encoding is presented. In this solution, the second, fifth, sixth and ninth sentences of the original CD are selected in the summary.

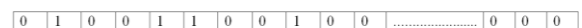


Figure 2. Particle's position encoding

4.3.2 Position updating

The BMA algorithm starts by generating a random population of N particles in continuous space. Each particle in the population updates his position at each iteration (displaced in the search space) using Eqs. (7) to (9) and the fitness function is called to evaluate the quality of the position until the maximum number of iterations.

It is impossible to use the BMA directly to tackle the MDS problem, because of the binary nature of MDS. Therefore, before the fitness evaluation of each position, a new binary position generated from the continuous position is necessary for the algorithm to be suitable for the MDS problem. Thus, a binary version of BMA (BBMA) is proposed based on the transfer function to tackle the MDS problem as a binary optimization problem. The transfer function is called in BBMA-MDS to convert the continuous position to binary position of particles, so that making the algorithm suitable for the MDS problem.

Several transfer functions are used in the literature; however, the sigmoid function is widely used and it proves promising results. So, in this approach, the sigmoid transfer function is used to BBMA-MDS for the position transformation. Eqs. (14) and (15) present the sigmoid function.

$$X_{sig} = \frac{1}{1 + e^{-x^{t+1}}} \quad (14)$$

$$X_b = \begin{cases} 1, & \text{Random} \leq x_{sig} \\ 0, & \text{Random} > x_{sig} \end{cases} \quad (15)$$

where X_b is the converted binary value of the real vector solution, and $Random$ is a random number used as the threshold.

4.3.3 Outlines of the binary biology migration algorithm for MDS problem

In the Algorithm 1 the pseudo-code of the proposed BBMA-MDS is presented.

Algorithm 1. Binary Biology Migration Algorithm

1. **Input:**
 2. Collection of documents to summarizing $CD = \{s_1, s_2, \dots, s_D\}$
 3. Number of particles (species) (NP)
 4. Maximum number of iterations (T)
 5. Switch probability (Pr)
 6. Maximum number of cycles (C)
 7. **Output:**
 8. Sum_{best} : Best summary of the Collection of documents
 9. **Begin**
 10. Initialize randomly the population $P = X_i$ ($i = 1, 2, \dots, NP$) in the continuous space
 11. **for** each particle i in P **do**
 12. Calculate the binary vector $XB_i(t)$ of the position $X_i(t)$ using Eqs. (14) and (15)
 13. Feasibility verification
 14. Generate the summary $Sum_i(t)$ that contains sentences of D that matching the “1” value in the binary vector $X_i(t)$
 15. Calculate the fitness value $F_i(t)$ of $Sum_i(t)$ using Eq. (6)
 16. **end for**
 17. S_1 = the fittest summary obtained by the particles of the population P
 18. $t=1$
 19. **while** ($t < T$)
 20. Calculate the step size $L(t)$ using Eq. (8)
 21. **for** each particle i in the population P
 22. (Migration phase)
 23. X_{best} = particle that have the best fitness from the population P
 24. **if** $rand < Pr$ **then** // $rand$ is a random value
 25. Calculate the new position $X_i(t+1)$ using Eq. (7)
 26. **else**
 27. Select two particles i and j randomly from the population P
 28. Calculate the new position $X_i(t+1)$ using Eq. (9)
 29. **end if**
 30. (Updating phase)
 30. Calculate the binary vector $XB_i(t+1)$ of the position $X_i(t+1)$ using Eqs. (14) and (15)
 31. Feasibility verification
 32. Generate the summary $Sum_i(t+1)$ that contains sentences of D that matching the “1” value in the binary vector $X_i(t+1)$
 33. Calculate the fitness value $F_i(t+1)$ of $Sum_i(t+1)$ using Eq. (6)
 34. **if** $F_i(t+1) > F_i(t)$ **then**
 35. $X_i(t) = X_i(t+1)$
 36. $Cycle(i) = 0$
 37. **else**
 38. $Cycle(i) = Cycle(i)+1$
 39. **if** $Cycle(i) \geq C$ **then**
 40. Generate randomly the position $X_i(t+1)$
 41. **end if**
 42. **end if**
 43. **end for**
-

-
44. Sum_t = the fittest summary obtained by all the population P in the iteration t
 45. $F_{best}(t+1)$ = the fitness of the best particle in the population P at iteration t
 46. **if** $F_{best}(t+1) > F_{best}(t)$ **then**
 47. $Sum_{best} = Sum_t$
 48. **end while**
 49. $t = t+1$
 50. **Output** the summary Sum_{best}
-

4.4 Summary generation

The final step of our approach is to produce the summary by retrieving the fittest particle after the maximum number of iterations. The particle is represented as a vector of D elements, each element corresponding to a sentence in the original CD . The summary is generated by comparing the particle vector with a pre-recorded list of sentences. If an element in the particle vector has a value of 1, the corresponding sentence in the list is included in the summary. Otherwise, the sentence is omitted out the summary.

5. EXPERIMENTS AND RESULTS

This section details the experimentation to assess the performance of our approach. It encompasses the used datasets, evaluation metrics, control parameters of the BBMA-MDS, and analysis of fitness function parameters. The section concludes with the presentation of the final results of our approach and a comparison with other works that have used the same datasets.

5.1 Dataset

The DUC (Document Understanding Conference) offers several datasets for text summarization. DUC datasets are centred on single-document and multi-document summarization problems. Each dataset contains a set of source documents in English together with Golden standard summaries created by human experts. To evaluate our approach, we must compare the summaries generated by BBMA-MDS approach with those proposed by the DUC. For the scope of this work, DUC2002 and DUC2004 datasets are used. Table 1 provides a short description of the used DUC datasets which are available in the link: <https://www-nlpir.nist.gov/projects/duc/data>

Table 1. Description of DUC2002 and DUC2004 datasets

Dataset PARAMETERS	DUC2002	DUC2004
Number of collections	60	100
Document per collection	10	10
Data source	duc.nist.gov	TREC
Summary length (word)	100	100

5.2 Evaluation measures

ROUGE is an abbreviation of "Recall Oriented Understudy for Gisting Evaluation"; it is a set of metrics used for automatic text summarization evaluation. The metrics compare the automatically generated summary with the Golden summary (summary generated by an expert human). A higher ROUGE score indicates that the automatically generated summary is more similar to the reference one. ROUGE-1.5.5 tool developed in the study [37] includes various ROUGE metrics such as ROUGE-N, ROUGE-L, ROUGE-S and ROUGE-SU.

ROUGE-N represents the overlap of N-grams between a system generated summary and a reference summary; it is calculated in (16) as follows:

$$ROUGE - N = \frac{\sum_{S \in \text{summ-ref}} \sum_{N\text{-gram} \in S} \text{Count}_{\text{Match}}(N - \text{gram})}{\sum_{S \in \text{summ-ref}} \sum_{N\text{-gram} \in S} \text{Count}(N - \text{gram})} \quad (16)$$

where $\text{Count}_{\text{Match}}(N - \text{gram})$ is the number of N-grams that are found both in the system summary and in a reference summary. For $\text{Count}(N - \text{gram})$ it corresponds to the number of N-grams in the reference summary.

In the following, we explain the formulas for measuring recall, precision, and F-measure for ROUGE-N. A summary generated by the proposed system is denoted as a candidate summary (C_{sum}). This latter will be compared with the reference summaries (R_{sum}).

The formulas (17), (18) and (19) give precision, recall and F-score respectively.

$$\text{Precision} = \frac{|C_{sum} \cap R_{sum}|}{|C_{sum}|} \quad (17)$$

$$\text{Recall} = \frac{|C_{sum} \cap R_{sum}|}{|R_{sum}|} \quad (18)$$

$$F - \text{score} = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \quad (19)$$

5.3 Controlling parameters

5.3.1 Algorithm parameters

In optimization algorithms, the values of control parameters vary depending on the application problem. Since there are no universal values that can be applied to all problems, it is necessary to conduct experiments to identify the most appropriate values for each problem.

To find the most adapted values of the parameters for our problem (Maximum number of iterations (T), Number of particles (NP) and Maximum number of cycles(C)) we have tested various combinations (T, NP and C) and calculated the fitness value for each one. The values of T, NP and C that yield the highest fitness value constitute the most appropriate combination. Table 2 gives the most appropriate values for T, NP and C that will be used in the remainder of this work. Table 2 also contains the value of Switch probability (Pr); it is a fixed value in the original algorithm.

Table 2. Algorithm's parameters

Parameter	Value
T	50
NP	100
C	10
Pr	0.5

5.3.2 Fitness function parameters

The fitness function is specifically designed to mirror the quality of the generated summary. As the fitness value increases, it signifies an enhancement in the overall quality of the summary. This observation aligns precisely with the results presented in Figures 3 and 4, and Tables 3 and 4, where higher fitness values correspond to improved ROUGE scores, indicating more effective and well-constructed summaries.

From the results of the experiments presented in Tables 3 and 4, our approach gives good results in most cases. Furthermore, its best results are obtained when the coefficients α , β and γ are fixed at the values 0.8, 0.1 and 0.1 respectively.

The results shown in Tables 3 and 4 suggest that setting α to 0.8, β to 0.1 and γ to 0.1 improve the ROUGE-1 and ROUGE-2 scores on the DUC2002 and DUC2004 datasets.

Table 3. ROUGE score on DUC2004 dataset.

α	β	γ	ROUGE-1	ROUGE-2
0.33	0.33	0.33	0.3992	0.1593
0.7	0.2	0.1	0.4175	0.1667
0.8	0.1	0.1	0.4379	0.1770
0.6	0.2	0.2	0.3923	0.1353
0.7	0.1	0.2	0.4144	0.1523
0.6	0.1	0.3	0.3564	0.1000
0.6	0.3	0.1	0.3987	0.1429

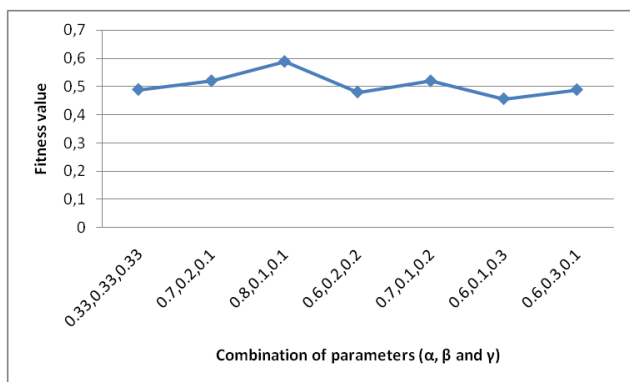


Figure 3. Variation of the fitness value vs. Combination of parameters (α , β and γ) on DUC2004

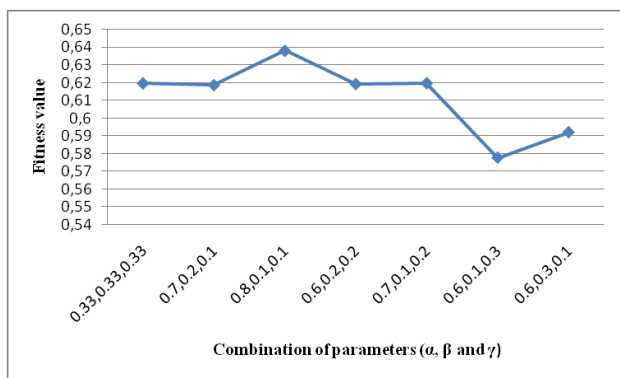


Figure 4. Variation of the fitness value vs. Combination of parameters (α , β and γ) on DUC2002

Our fitness function uses the weights α , β and γ respectively for the factors; coverage, cohesion and readability. To determine the most suitable combination of α , β and γ for our problem, several combinations of these weights have been tested. For each combination, we calculated the fitness value

and evaluated the generated summary using ROUGE-1 and ROUGE-2. These experiments were performed on the two datasets; DUC2002 and DUC2004. The results are presented in Figures 3 and 4 for the fitness values, along with Tables 3 and 4 for ROUGE-1 and ROUGE-2.

The increase in the ROUGE score with the weight of Coverage in the BBMA-MDS approach indicates that the algorithm places more emphasis on including relevant information from the input documents in the summary. As the weight of Coverage increases, the algorithm becomes more focused on ensuring that relevant information and the content coverage of all documents are well-represented in the generated summary. We conclude that the coverage is important to generate a good summary.

Table 4. ROUGE score on DUC2002 dataset

α	β	γ	ROUGE-1	ROUGE-2
0.33	0.33	0.33	0.4501	0.1993
0.7	0.2	0.1	0.4614	0.2291
0.8	0.1	0.1	0.4851	0.2619
0.6	0.2	0.2	0.4618	0.2249
0.7	0.1	0.2	0.4501	0.2192
0.6	0.1	0.3	0.4295	0.1867
0.6	0.3	0.1	0.4440	0.1979

6. RESULTS AND COMPARISON WITH OTHER WORKS

6.1 Results

The evaluation results of our approach using ROUGE-1, ROUGE-2 and ROUGE-SU4, with α set to 0.8, β set to 0.1 and γ set to 0.1, are summarized in Tables 5 and 6. The tables show the recall, precision and F-score for ROUGE-1 and ROUGE-2 on DUC2002 and DUC2004 datasets.

Table 5. Recall, precision and F-score for the BBMA-MDS algorithm on DUC-2002

	Rouge-1	Rouge-2	Rouge-Su-4
Average Recall	0.4842	0.2611	0.2830
Average Precision	0.4862	0.2628	0.2847
Average F-score	0.4851	0.2619	0.2838

Table 6. Recall, precision and F-score for the BBMA-MDS algorithm on DUC-2004

	Rouge-1	Rouge-2	Rouge-Su-4
Average Recall	0.4282	0.1726	0.1903
Average Precision	0.4489	0.1819	0.2000
Average F-score	0.4379	0.1770	0.1949

6.2 Comparison with other works

6.2.1 Comparison on DUC 2002

Comparison with classical algorithms. Our approach, BBMA-MDS, is compared with several classical algorithms such as; LexRank [18] where a similarity graph is created with sentences as nodes and edges connecting sentences with a cosine similarity above a certain threshold. The PageRank score of each sentence is then determined within this graph. The most highly ranked sentences are then selected to form the summary; TF-IDF weighting [5] in this work the score of each

sentence is calculated from the TF*IDF of its terms and the highest scoring sentences are chosen until the desired summary length is reached; JS-Greedy; KL-Greedy, and ICSI. These algorithms were implemented and evaluated using the DUC2002 dataset, as described in the study [38]. The comparison results are shown in Table 7.

Table 7. Performance comparison of BBMA-MDS with classical algorithms on DUC2002

Methods	ROUGE-1	ROUGE-2
TF-IDF	0.4072	0.1201
Lex Rank	0.4311	0.1388
KL-Greedy	0.3945	0.1125
JS-Greedy	0.4299	0.1455
ICSI	0.4434	0.1556
BBMA-MDS	0.4851	0.2619

The results show that our approach outperforms other methods in terms of ROUGE-1 and ROUGE-2 metrics. BBMA-MDS has the highest ROUGE-1 score of 0.4851 and the highest ROUGE-2 score of 0.2619.

Comparison with optimization approaches. In this part, our approach is compared with FBTS [28], FEOM [25] and many metaheuristic-based approaches implemented on the DUC-2002 dataset [28]; Firefly Algorithm with JS-Divergence, PSO using cosine similarity and JS-divergence as fitness functions, ant colony Optimization with cosine similarity, Genetic Algorithm with JS-divergence and cosine similarity.

Table 8. Performance comparison of BBMA-MDS with other metaheuristic-based approaches on DUC-2002

Methods	ROUGE-1	ROUGE-2
FA (JS -Divergence)	0.3262	0.1527
GA (JS-Divergence)	0.4117	0.1758
GA (Cosine Similarity)	0.3597	0.1374
PSO (JS-Divergence)	0.3759	0.1312
PSO (Cosine Similarity)	0.3235	0.1835
FEOM	0.4657	0.1249
ACO (Cosine Similarity)	0.3289	0.1589
FbTS (TRF, CF, RF)	0.4782	0.2295
BBMA-MDS (C, C, R)	0.4851	0.2619

The results in Table 8 and Figure 5 reveal that our approach, BBMA-MDS, has the highest ROUGE-1 score of 0.4851 and ROUGE-2 score of 0.2619, outperforming other methods. These results demonstrate the effectiveness of our approach in comparison to others.

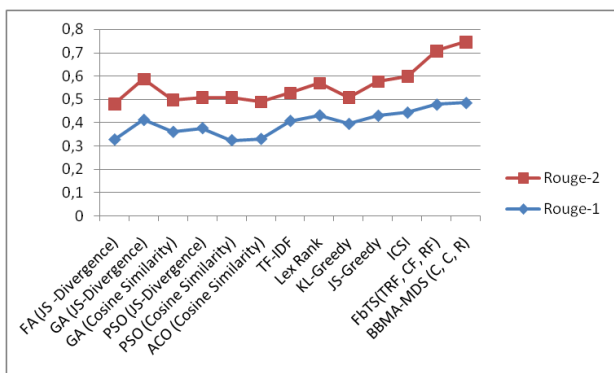


Figure 5. ROUGE-1 and ROUGE-2 comparison of BBMA-MDS with other works on DUC-2002 dataset

6.2.2 Comparison on DUC 2004

In this part our approach is compared with several meta heuristic-based approaches such as; Genetic algorithm with JS-divergence, ACO with cosine similarity, PSO with JS-divergence [28]. MCRMR-PSO with cosine similarity and MCRMR-SSO with cosine similarity, MMR-SE System 61 with centroid score and LexRank [22]. The results of the comparison are presented in Table 9 and Figure 6.

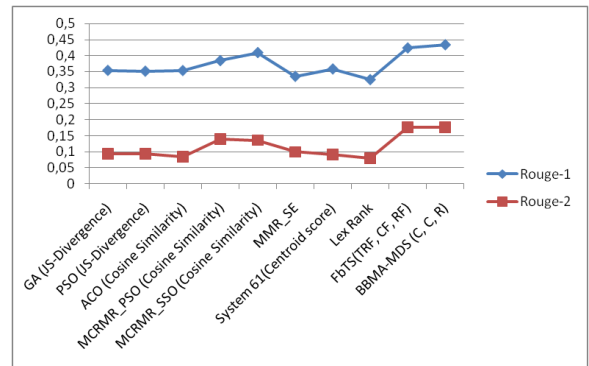


Figure 6. ROUGE-1 and ROUGE-2 comparison of BBMA-MDS with other works on DUC-2004 dataset

Table 9. Performance comparison of BBMA-MDS with other methods on DUC-2004 dataset

Methods	ROUGE-1	ROUGE-2
GA (JS-Divergence)	0.3546	0.0937
PSO (JS-Divergence)	0.3521	0.0935
ACO (Cosine Similarity)	0.3542	0.0837
MCRMR_PSO (Cosine Similarity)	0.385	0.139
MCRMR_SSO (Cosine Similarity)	0.410	0.136
MMR_SE	0.336	0.099
System 61 (Centroid score)	0.359	0.091
Lex Rank	0.326	0.079
FbTS (TRF, CF, RF)	0.4244	0.1764
BBMA-MDS (C, C, R)	0.4343	0.1770

Comparing the ROUGE-1 and ROUGE-2 scores from Table 9, our approach achieves the highest scores of 0.4343 and 0.1770 respectively, outperforming all other methods on DUC2004 dataset.

6.3 Interpretation and discussion of results

After conducting an analysis of the experimental results, it is clear that BBMA-MDS surpasses all the methods listed in Tables 7, 8, and 9. Notably, our approach outperforms FBTS, which achieved the best result in related works, in both the DUC2002 and DUC2004 datasets when assessed using ROUGE-1 and ROUGE-2 metrics. Specifically, compared with FBTS, BBMA-MDS demonstrate an improvement on DUC2002 of 0.0069 and 0.0324 in terms of ROUGE-1 and ROUGE-2 respectively. For DUC2004, the improvement is of 0.0099 and 0.0006 in ROUGE-1 and ROUGE-2 respectively.

When BBMA-MDS outperforms other methods on the ROUGE-1 and ROUGE-2 metrics, it indicates that BBMA-MDS is more effective at generating summaries that closely match the content of the source documents. Higher ROUGE-1 scores show that BBMA-MDS captures important words and maintains language fluency, while higher ROUGE-2 scores demonstrate its ability to maintain contextual coherence by capturing important word sequences. This superior

performance implies that BBMA-MDS produces more informative, comprehensive, and coherent summaries, making it a promising approach for real-world applications.

On the other hand, these outcomes on ROUGE metrics mean that the system achieves higher recall, precision, and F-scores. This indicates that BBMA-MDS can cover more relevant information from the source documents (higher recall), produce summaries with fewer unnecessary words (higher precision), and achieve a better balance between these two aspects (higher F-score). In summary, the superior recall, precision, and F-score results contribute to the overall effectiveness of BBMA-MDS in generating summaries that closely match the content of the reference summaries, as measured by the ROUGE evaluation.

These impressive outcomes can be attributed to the well-defined fitness function and the BBMA-MDS algorithm's ability to effectively balance between diversification and intensification.

7. CONCLUSION

A new BBMA-MDS metaheuristic-based algorithm for MDS is proposed in this work. This algorithm addresses the generation of extractive summaries as a binary optimization problem. After modeling the MDS as an optimization problem, a binarization step was added to adapt the BMA algorithm to the text summarization problem. The fitness function is proposed to measure the quality of the produced summary and is comprised of three elements: Coverage, Cohesion and Readability, each with different weights. The effectiveness of the proposed approach was evaluated by conducting several experiments on the DUC2002 and DUC2004 datasets. The results were evaluated using ROUGE scores. The results of the proposed BBMA-MDS showed better performance in terms of ROUGE-1 and ROUGE-2 compared to many baseline algorithms like TF-IDF, LexRank and several metaheuristic-based approaches such as FBTS and PSO.

We have to outline our main contributions:

- 1- We used BMA for the first time to resolve MDS problems,
- 2- The existing binarization in the literature is naïve, and to make more efficient the discretization of the continuous aspect of classical BMA, we attributed to it sigmoid function for the first time, transforming the real numbers to numbers between 0 and 1
- 3- We proposed a new objective function to assess the quality of the summary, with three features (coverage, cohesion and readability), and we attributed the three corresponding parameters or weights, not arbitrary but after finding the more appropriate ones to relevant results

The practical implications of the work on the scientific literature are:

- 1- the adaptation of the binarization to BMA more efficiently
- 2- the weights should be attributed scientifically to the features of the objective functions to get better findings of optimization problems.
- 3- This work could be used for scientific reviews to gain time and minimize efforts of the scientists in traditional preparation of the state of the art

The practical implications of the work on the real world are:

- 1- The improved summaries improve in their tour the efficiency and the precision of the business intelligence applied in enterprises to decision support
- 2- It could help in content analysis on the social media sites and discover the social trends
- 3- It may help in the preparation of brief news for journalists

Even we obtain better results in our work, it remains presenting certain limits:

- 1- The BBMA-MDS is only tested on DUC2002 and DUC2004 and not tested on other datasets
- 2- The BBMA-MDS is evaluated only by the ROUGE metrics

As perspectives of this work, we aim to:

- 1- Use a learning method to identify the best values for α , β , and γ , which represent the weights of the three factors of the fitness function, and make improvements to the BBMA-MDS algorithm to increase its capabilities in diversification and intensification.
- 2- The use of other metaheuristics or the combination of different bio-inspired algorithms is also envisaged in this context,
- 3- Create our own datasets in diversified contexts to test the efficiency of the solution to gain space for its generalizability,
- 4- Adaptation of the approach to produce automatically a scientific state of the art for scientists.

This work remains open to other improvements.

ACKNOWLEDGMENT

We are grateful to authors and our laboratories to support our researches and special gratitude come back to Drs Hichem Houassi, Hichem Rahab and Abdelaali Bekhouche for their help and advises.

REFERENCES

- [1] Ingwersen, P. (1992). *Information Retrieval Interaction*. London: Taylor Graham.
- [2] Babar, S.A., Patil, P.D. (2015). Improving performance of text summarization. *Procedia Computer Science*, 46: 354-363. <https://doi.org/10.1016/j.procs.2015.02.031>
- [3] Lloret, E., Palomar, M. (2012). Text summarisation in progress: A literature review. *Artificial Intelligence Review*, 37: 1-41. <https://doi.org/10.1007/s10462-011-9216-z>
- [4] Jezek, K., Steinberger, J. (2008). Automatic summarizing: (The state-of-the-art 2007 and new challenges). In *Proc. Znalosti*, pp. 1-12.
- [5] Luhn, H.P. (1958). The automatic creation of literature abstracts. *IBM Journal of Research and Development*, 2(2): 159-165. <https://doi.org/10.1147/rd.22.0159>
- [6] Alguliev, R.M., Aliguliyev, R.M., Hajirahimova, M.S., Mehdiyev, C.A. (2011). MCMR: Maximum coverage and minimum redundant text summarization model. *Expert Systems with Applications*, 38(12): 14514-14522. <https://doi.org/10.1016/j.eswa.2011.05.033>
- [7] Sharma, G., Sharma, D. (2022). Improving extractive text summarization performance using enhanced feature

- based RBM method. *Revue d'Intelligence Artificielle*, 36(5): 777-784. <https://doi.org/10.18280/ria.360516>
- [8] Khan, A., Salim, N., Kumar, Y.J. (2015). A framework for multi-document abstractive summarization based on semantic role labelling. *Applied Soft Computing*, 30: 737-747. <https://doi.org/https://doi.org/10.1016/j.asoc.2015.01.070>
- [9] Nallapati, R., Zhou, B., Gulcehre, C., Xiang, B. (2016). Abstractive text summarization using sequence-to-sequence rnns and beyond. *arXiv preprint arXiv:1602.06023*. <https://doi.org/10.48550/arXiv.1602.06023>
- [10] Erkan, G., Radev, D.R. (2004). Lexrank: Graph-based lexical centrality as salience in text summarization. *Journal of Artificial Intelligence Research*, 22: 457-479. <https://doi.org/10.1613/jair.1523>
- [11] Tomer, M., Kumar, M. (2020). Improving text summarization using ensembled approach based on fuzzy with LSTM. *Arabian Journal for Science and Engineering*, 45: 10743-10754. <https://doi.org/10.1007/s13369-020-04827-6>
- [12] Gehrmann, S., Deng, Y., Rush, A.M. (2018). Bottom-up abstractive summarization. *arXiv preprint arXiv:1808.10792*. <https://doi.org/10.48550/arXiv.1808.10792>
- [13] Nenkova, A., McKeown, K. (2012). A survey of text summarization techniques. *Mining text data*, 43-76. https://doi.org/10.1007/978-1-4614-3223-4_3
- [14] Wong, K.F., Wu, M., Li, W. (2008). Extractive summarization using supervised and semi-supervised learning. In *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)*, Manchester, UK, pp. 985-992. <https://aclanthology.org/C08-1124.pdf>
- [15] Rautray, R., Balabantaray, R.C. (2017). Cat swarm optimization based evolutionary framework for multi document summarization. *Physica A: Statistical Mechanics and Its Applications*, 477: 174-186. <https://doi.org/10.1016/j.physa.2017.02.056>
- [16] Pembe, F.C., GÜngör, T. (2007). Automated querybiased and structure-preserving text summarization on web documents. In *Proceedings of the International Symposium on Innovations in Intelligent Systems and Applications*, Istanbul.
- [17] Goldstein, J., Mittal, V.O., Carbonell, J.G., Kantrowitz, M. (2000). Multi-document summarization by sentence extraction. In *NAACL-ANLP 2000 workshop: automatic summarization*. <https://aclanthology.org/W00-0405.pdf>
- [18] Mihalcea, R., Tarau, P. (2004). Textrank: Bringing order into text. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, Barcelona, Spain, pp. 404-411. <https://aclanthology.org/W04-3252>
- [19] Filippova, K., Mieskes, M., Nastase, V., Ponzetto, S.P., Strube, M. (2007). Cascaded filtering for topic-driven multi-document summarization. In *Proceedings of the Document Understanding Conference*, Heidelberg, Germany, pp. 26-27.
- [20] Kupiec, J., Pedersen, J., Chen, F. (1995). A trainable document summarizer. In *Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, PaloAlto, CA, UCA, pp. 68-73. <https://doi.org/10.1145/215206.215333>
- [21] Zhou, L., Hovy, E. (2003). A web-trained extraction summarization system. In *Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, pp. 284-290. <https://aclanthology.org/N03-1037>
- [22] Verma, P., Om, H. (2019). MCRMR: Maximum coverage and relevancy with minimal redundancy based multi-document summarization. *Expert Systems with Applications*, 120: 43-56. <https://doi.org/10.1016/j.eswa.2018.11.022>
- [23] Selvan, R.S., Arutchelvan, K. (2021). Improved cuckoo search optimization algorithm based multi-document summarization model. In *2021 5th International Conference on Computing Methodologies and Communication (ICCMC) Erode, India*, pp. 735-739. IEEE. <https://doi.org/10.1109/ICCMC51019.2021.9418473>
- [24] Zhang, Q., Wang, R., Yang, J., Lewis, A., Chiclana, F., Yang, S. (2019). Biology migration algorithm: A new nature-inspired heuristic methodology for global optimization. *Soft Computing*, 23: 7333-7358. <https://doi.org/10.1007/s00500-018-3381-9>
- [25] Song, W., Choi, L.C., Park, S.C., Ding, X.F. (2011). Fuzzy evolutionary optimization modeling and its applications to unsupervised categorization and extractive summarization. *Expert Systems with Applications*, 38(8): 9112-9121. <https://doi.org/10.1016/j.eswa.2010.12.102>
- [26] Rautray, R., Balabantaray, R.C. (2018). An evolutionary framework for multi document summarization using Cuckoo search approach: MDSCSA. *Applied Computing and Informatics*, 14(2): 134-144. <https://doi.org/10.1016/j.aci.2017.05.003>
- [27] Selvan, R.S., Arutchelvan, K. (2021). An efficient social spider optimization algorithm based multi-document summarization model. In *2021 6th International Conference on Inventive Computation Technologies (ICICT) Coimbatore, India*, pp. 855-860. <https://doi.org/10.1109/ICICT50816.2021.9358521>
- [28] Tomer, M., Kumar, M. (2022). Multi-document extractive text summarization based on firefly algorithm. *Journal of King Saud University-Computer and Information Sciences*, 34(8): 6057-6065. <https://doi.org/10.1016/j.jksuci.2021.04.004>
- [29] Abdelaali, B., Hichem, H., Mahdaoui, R., Hichem, R., Makhlof, L. (2022). ADCSA-WSD: Adapted discrete crow search algorithm for word sense disambiguation. *Revue d'Intelligence Artificielle*, 36(1): 131-138. <https://doi.org/10.18280/ria.360115>
- [30] Ranganarayana, K., Rao, G.V. (2022). Modified ant colony optimization for human recognition in videos of low resolution. *Revue d'Intelligence Artificielle*, 36(5): 731-736. <https://doi.org/10.18280/ria.360510>
- [31] Javidi, M.M. (2021). Feature selection schema based on game theory and biology migration algorithm for regression problems. *International Journal of Machine Learning and Cybernetics*, 12(2): 303-342. <https://doi.org/10.1007/s13042-020-01174-8>
- [32] Zhang, Q.Y., He, X.Y., Dong, Y.Q., Jiang, S.Y., Song, H. (2021). Optimizing the flight route of UAV using biology migration algorithm. In *Journal of Physics:*

- Conference Series, 2025(1): 012052. <https://doi.org/10.1088/1742-6596/2025/1/012052>
- [33] Krause, J., Cordeiro, J., Parpinelli, R.S., Lopes, H.S. (2013). A survey of swarm algorithms applied to discrete optimization problems. In *Swarm Intelligence and Bio-Inspired Computation*, pp. 169-191. <https://doi.org/10.1016/B978-0-12-405163-8.00007-7>
- [34] Agrawal, P., Abutarboush, H.F., Ganesh, T., Mohamed, A.W. (2021). Metaheuristic algorithms on feature selection: A survey of one decade of research (2009-2019). *IEEE Access*, 9: 26766-26791. <https://doi.org/10.1109/ACCESS.2021.3056407>
- [35] Rahab, H., Haouassi, H., Laouid, A. (2023). Rule-based Arabic sentiment analysis using binary equilibrium optimization algorithm. *Arabian Journal for Science and Engineering*, 48(2): 2359-2374. <https://doi.org/10.1007/s13369-022-07198-2>
- [36] Hichem, H., Elkamel, M., Rafik, M., Mesaaoud, M.T., Ouahiba, C. (2022). A new binary grasshopper optimization algorithm for feature selection problem. *Journal of King Saud University-Computer and Information Sciences*, 34(2): 316-328. <https://doi.org/https://doi.org/10.1016/j.jksuci.2019.11.007>
- [37] Lin, C.Y. (2004). Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pp. 74-81.
- [38] Peyrard, M., Eckle-Kohler, J. (2016). A general optimization framework for multi-document summarization using genetic algorithms and swarm intelligence. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pp. 247-257. <https://aclanthology.org/C16-1024.pdf>.