Applied Intelligence
https://doi.org/10.1007/s10489-022-04149-0

Single document text summarization addressed with a cat swarm optimization approach

Dipanwita Debnath[1] · Ranjita Das[1] · Partha Pakray[2]

Accepted: 3 September 2022

Abstract
The availability of a tremendous amount of online information bringing about a broad interest in extracting relevant information in a compact and meaningful way, prompted the need for automatic text summarization. Hence, in the proposed system, the automated text summarization has been considered as an extractive single-document summarization problem, and a Cat Swarm Optimization (CSO) algorithm-based approach is proposed to solve it, whose objective is to generate good summaries in terms of content coverage, informative, anti-redundancy, and readability. In this work, input documents are pre-processed first. Then the cat population is initialized, where each individual (cat) in a binary vector is randomly initialized in the search space, considering the constraint. The objective function is then formulated considering different sentence quality measures. The Best Cat Memory Pool (BCMP) is initialized based on the objective function score. After that, individuals are randomly distributed for position updating to perform seeking/tracing mode operations based on the mixture ratio in each iteration. BCMP is also updated accordingly. Finally, an optimal individual is chosen to generate the summary after the last iteration. DUC-2001 and DUC-2002 data sets and ROUGE measures are used for system evaluation, and the obtained results are compared with the various state-of-the-art methods. We have achieved approximately 25% and 5% improvement on ROUGE-1 and ROUGE-2 scores on the datasets over the best existing method mentioned in this paper, revealing the proposed method's superiority. The proposed system is also evaluated considering the generational distance, CPU processing time, cohesion, and readability factor, reflecting that the system-generated summaries are readable, concise, relevant, and fast. We have also conducted a two-sample t-test, and one-way ANOVA test showing the proposed approach is statistically significant.

Keywords Optimization technique · Cat swarm optimization · Features scaling · Single document · Extractive text summarization · Automatic text summarization · Statistical analysis

1 Introduction

The text summarizer generates the shorter version of the text document (summaries), keeping the key concept intact [1, 2]. Thus, machine-generated summaries can provide a quick overview of the document that helps the reader to identify the document's usefulness in a short time, which further helps in quick decision-making [3]. However, Text Summarization (TS) can be categorized into many ways [4–7], such as: (i). Based on the input-output language, TS is classified into three groups, i.e., single, multi, and cross-lingual system [8]. The input-output is represented in the same language (one language) in a single language system. A multi-lingual system can operate in any language; however, the input and corresponding output should be in the same language. For example, if the system's input text is in Hindi, the system-generated summary should also be in Hindi. Input-output can be of two different languages in a cross-lingual system, i.e., if the given input text is in the English language, then the system can generate summaries in Hindi or any other language. (ii). Based

✉ Ranjita Das
ranjita.nitm@gmail.com

Dipanwita Debnath
ddebnath.nita@gmail.com

Partha Pakray
parthapakray@gmail.com

[1] National Institute of Technology Mizoram, Mizoram, 796012, India

[2] National Institute of Technology Silchar, Assam, 788010, India

Published online: 24 September 2022

on the type of input text, TS is classified into a single and multi-document system [9, 10]. The input text is a single document in a single documented system, such as a news article or a scientific research paper. In contrast, in the multi-document system, more than one document can be given as input text, for example, news articles related to the recent CoVID-19 pandemic published in several newspapers. (iii). Considering the type of summary, TS is classified as query-based, domain-specific and generic categories [11]. The whole document is considered for summary generation in the generic system. Only domain or query-specific data are considered in a domain or query-specific system. (iv). Based on extraction methodology, TS is classified as Extractive Text Summarization (ETS) and Abstractive Text Summarization (ATS) systems [12]. In the ETS system, summaries are generated using rule-based or machine learning algorithms, where some sentences or portions of the sentence are selected from the text itself, whereas, in the ATS system, key terms are identified, knowledge is extracted, and based on past and present knowledge related to the text, summaries are generated using natural language generation techniques [12].

In the past, plenty of work has been reported on TS, including rule-based [13], clustering-based [14], graph-based [15], neural network-based [16], and evolutionary-based [2, 17] approaches. Among these, evolutionary approaches, such as, swarm-based [3, 10, 18–20], FUZZY-based [21] Genetic Algorithm (GA) based [14, 22], Harmonic Search (HS) based [23, 24], and Differential Evaluation (DE) based [17, 25] approaches are majorly used to solve this TS problem. One such meta-heuristic optimization technique is Cat Swarm Optimization (CSO). The reason behind opting for CSO is:

1. CSO and its variations are applied on several optimization problems where it has proved its ability against algorithms like GA and PSO [26, 27].
2. CSO was tested on a vast number of classical functions, and different CSO-based applications belonging to other fields were analyzed, where it outperformed various optimization-based algorithms, namely dragonfly, butterfly, filter dependent, NSGA-II, GA, and DE [28–30].
3. Saha SK et al. [31] used CSO for optimal linear phase FIR filter design. They revealed that CSO convergences faster and possess the best convergence characteristics with the least execution time. They also demonstrated that CSO outperforms the GA, DE, and PSO algorithms in fast convergence and minimum execution time.
4. According to our knowledge, in the text summarization field, only a few works have been reported so far, and we found that using CSO, the obtained summaries are readable and relevant [10, 12]. This work is the extension of our previous work [12].

We are motivated by the evolutionary algorithm's remarkable ability to solve the ETS problem, and CSO is one of them. So, for summarizing a single document (in a single language, i.e., English), we have used the Modified Binary Cat Swarm Optimization technique and named it ETS-MBCSO. In the proposed system, we have modified the binary CSO [27] algorithm so that it does not get stuck into the local optima and can perform well in the text summarization problem. To justify our perception that modified CSO performs well, we have compared the ETS-MBCSO approach with classical CSO, GA, PSO, archive-based micro genetic-2 algorithm (AMGA-2), DE, and grey wolf optimization (GWO) based approaches and found that in the text summarization problem, ETS-MBCSO performs well compared to them. The proposed method performs the following steps: First, the input text is pre-processed. Relevant features are identified, and each document sentence is further represented as a feature vector (content covering and informative score). Then the population is initialized randomly, where each individual is encoded in a binary vector. The constraints are checked for each individual, and only feasible individuals are considered for further processing. Each feasible individual's fitness score is calculated utilizing the sentence feature vector, redundancy score, and feature scaling. Afterward, individuals are sorted based on their fitness score, and the best three individuals are stored in the Best Cat Memory Poll (BCMP). The individuals' positions are updated in each iteration considering the constraints and fitness score, and the BCMP is updated using the best three individuals. After the last iteration best individual of BCMP is used to create the summary.

The major contributions of this paper are enumerated below:

1. A modified binary cat swarm optimization approach is used for sentence-based optimal summary generation.
2. For uniform features distribution in the weighted sum based objective function and to avoid unintentional features dominance feature scaling is used [2, 32].
3. Functionalities such as objective function, position updating, and constraint are designed.
4. Showing the effectiveness of the functionalities like objective function formulation and child generation, a comparative analysis is also performed to show that the proposed system (ETS-MBCSO) performs well concerning classical CSO, GA, PSO, AMGA-2, DE, and GWO.

The rest of the paper is organized as follows: In Section 2 the detailed literature survey is presented along with an overview of the binary cat swarm optimization approach. In Section 3, the text summarization problem is defined considering the optimization aspects. The proposed system architecture is presented in Section 4, and the corpora and experimental setups are discussed in Section 5. Obtained results are discussed in Section 6, along with a comprehensive analysis of the system's performance and statistical analysis. Finally, in Section 7, we have concluded the paper with some insights into future work.

2 Literature survey

A brief discussion on meta-heuristic-based ETS approaches, the recent ETS approaches, and an overview of the cat swarm optimization algorithm are presented in this Section.

2.1 Meta-heuristic based ETS approaches

Researchers proposed several meta-heuristic-based ETS approaches for generating text summaries, most of which outperformed the existing TS (ETS and ATS) systems in terms of readability and ROUGE scores over the last decade. Some of these systems are reported in this Section.

In [5, 14, 15, 33], features and threshold-based multi-document summaries are extracted using GA where features such as TF-IDF score, sentence position, redundancy score, length of the sentences, sentence centrality, etc. are considered. Summary length is regarded as the threshold. M. Mendoza et al. used a memetic binary optimization algorithm integrating the population-based global and local search in MA-SingleDocSum [34] and global-best harmony search and a greedy local search procedure called as ESDS-GHS-GLO in [32] for generating automatic single document summaries. Thirty-one sentence features are studied and evaluated using the GA-based TS approach in [35]. Among these thirty-one features, the most relevant features reported are coverage, sentence position, sentence length, the sentence similarity with the title. Saleh and Kadhim [24] have developed MOEA/D, considering the two objectives, content coverage and diversity. In DPSO [36], Particle swarm optimization (PSO) is used as an underlying optimization technique for optimizing a single objective function, formulated by taking the mean of coverage and diversity-related features. ESDS-SMODE was proposed by N. Saini et al. [17], a self-organizing map incorporating the DE approach, which can automatically detect the number of clusters by optimizing two cluster validity indexes. From the clusters, sentences are selected after ranking. In the ESDocSum [37] approach, N. Saini et al. proposed a self-organizing map incorporating DE-based ETS approach where six objectives are optimized simultaneously. COSUM [2] is also a cluster optimization approach, using K-Mean clustering and adaptive DE-based optimization. Cat swarm optimization is utilized by R.Rautary et al. [10] to solve a multi-document text summarization problem. In [38], each sentence of the document is represented using TF-IDF scores based one-hot vector and then grouped according to their proximity measures. The cluster's quality is evaluated using silhouette index and GA, which helped to select the best approximate number of clusters. Finally, LDA (Latent Dirichlet Allocation) model is used to determine the sentences from the clusters. A decomposition-based multi-objective artificial bee colony algorithm is utilized in [39] to generate summaries, where three objective functions are optimized simultaneously. This approach was tested on ten documents from the DUC-2002 dataset and obtained a ROUGE-1 score of 0.553 and a ROUGE-2 score of 0.342. In ESDS-AMGA2 [40], an archive-based micro genetic algorithm-2 is used, where two objective functions are optimized, considering feature scaling and features representation as an essential functionality. Taner and Ali [41] used a graph-based extractive text summarization approach utilizing the maximum independent set and KUSH (a text processing toolkit) for generating generic summaries. Different weighted objective function schemes and similarity measures are implemented, compared, and analyzed by Sanchez-Gomez et al. in [42]. Deep learning based contextualized rewriting is performed to address the irrelevance, redundancy, and incoherence problem by Bao and Zhang [43]. Chettah and Draa [44], used decimal sentence encoding and weight features scoring incorporated discrete differential evolution approach to generate summaries.

2.2 Recent extractive text summarization approaches

A hybrid GA and PSO-based approach named PSOGA-BKSum is proposed in [45], where the objective function is formulated using four-sentence features: sentence position, similarity to the topic sentence, sentence length, and the number of proper nouns in a sentence. The comprehensive survey on TS done by WS El-Kassas et al. [46] reported that optimization-based TS needs high computation cost and time. A spider monkey optimization method is utilized for multi-document summarization, where both syntactic and semantic features are extracted and then enhanced using the softmax regression technique in [47]. Multi-document text summarization is proposed using a quantum-inspired genetic algorithm, where the objective function is formulated considering the summation of six features [48]. A Multi-Objective Artificial Bee Colony optimization approach is utilized as an underlying approach to show the importance of semantic similarity measure, and weight term score in [42]. In our previous work ESDS-MCSO [12], we have used two objective functions with non-dominated and crowding distance-based population sorting, where a similar cat swarm optimization approach is used as the underlying strategy. A pre-trained, encoder-only transformer language model (HiStruct+ mode) is used to formulate, extract, encode and inject hierarchical structure information explicitly into an extractive summarization model utilizing the relevant and content covering information such as the section titles and the hierarchical positions of sentences, etc. in [49]. A heuristic method is proposed that finds the required number of independent topics; then, important topic-based sentences are extracted using latent Dirichlet allocation; finally, several classifiers are used to generate a coherent summary in [50]. A graph-based summarization technique is proposed using features like the similarity among the summary sentences and the similarity between the text sentences in [51]. BERT embedding, logical regression, and similarity-based model are used for the biomedical document summarization in [52], showing certain contextual sentence features allow the classifier for sentence selection and rejection.

However, the existing TS approaches reported in Sections 2.1 and 2.2 suffer from at least one of the following drawbacks: (i) Poor ROUGE score [11, 35]. (ii) The TS problem is formulated as a single objective optimization problem, where the objective function is seen to be biassed towards some features [32, 53]. (iii) In many systems, the syntactic similarity measures are used [10, 34, 53]. (iv). Multi-objective optimization increases computation cost and time, as more number of functions are evaluated [12, 37]. All these drawbacks motivated us to find the optimal strategy.

2.3 Overview of cat swarm optimization

Cat Swarm Optimization (CSO) is a robust and powerful meta-heuristic optimization approach invented by Chu et al. [12, 26]. In this algorithm, every individual is designed to have the following parameters: position in the multi-dimensional space, velocities when it moves in any dimension, and fitness values. Two modes characterize the cat's position. In the seeking mode (SM), a cat rests, being alert. In contrast, it is modeled as a cat tracing the targets in the tracing mode (TM). CSO keeps the record of each cat's best position and the best-positioned cats at each interval until it reaches optimality. The conceptual steps of the CSO are shown in the Algorithm 1, and the seeking and tracing mode operations are further discussed below:

Seeking mode operation: when a cat/individual enters SM, the following steps are performed to update its position.

```
Input: Data and objective functions.
Output: The best individual.
Initialize C number of cats with their position and velocity.
Initialize best cat memory pool (BCMP) of size $|b|$ , $G_{max}$.
Evaluate the population using the fitness function and store the best cats into BCMP.
for iteration in range (0 to $G_{max}$) do
    for i in range (0 to C-1) do
        Assign TM/SM to each cat using MR.
        if $i^{\text{th}}$ cat is assigned to perform TM according to MR then
            Perform tracing mode operation.
        else
            Perform seeking mode operation.
    end for
    Update BCMP with the best cats.
end for
From BCMP, return the best cat.
Stop.
```

Algorithm 1 Classical Cat Swarm Optimization algorithm.

(i) making some replica-cat and updating each replica cat's position using mathematical operations (plus, minus). (ii) calculation of fitness score (FS) for each of them, (iii) if all the fitness scores are not equal, then calculate selecting probability of each replica-cat using the (1), where, FS_{cat} is the replica cat whose probability score is calculating, FS_{max} is the maximum fitted replica, FS_{min} is the minimum fitted replica and if its a maximization problem then $FS_b = FS_{max}$ otherwise $FS_b = FS_{min}$, and (iv) choose the best probable cat (including the current cat) to replace the current cat.

$$P_{cat} = \frac{FS_{cat} - FS_b}{FS_{max} - FS_{min}}. \tag{1}$$

Tracing mode operation: if a cat goes for TM, it first updates its velocity in all dimensions using (2). Note: if any velocity crosses the maximum allowed range, it is reset-ed to its maximum permitted value.

$$V_{i+1} = V_i + r_1 * c_1 * (X_{best} - X_i). \tag{2}$$

Where V_i is the velocity of the cat in one dimension, similarly velocity of all the dimensions is calculated to check possible moves (updated positions) in all the dimensions. Here, V_{i+1} is the updated velocity in the same dimension, r_1 is random value in the range [0,1], c_1 is a constant, X_{best} is the best-positioned cat in that dimension, and X_i is the current cat's position. After that, the current cat's position is updated using the velocity, as shown in (3).

$$X_{i+1} = X_i + V_{i+1} \tag{3}$$

3 Problem definition

We have formulated the sentence-based extractive single-document summarization problem as a constrained single objective optimization problem.

Let us consider a document D consisting of N sentences, $\{s_1, s_2, s_3, ..., s_N\}$. Our goal is to find the optimal summary consisting of some of the sentences $\{s_1, s_2, ..., s_m\}$ from the document D, such that it has the maximum fitness score and it is bounded by length.

We have taken the summary length limit as 100 words in this paper. So, we have formulated two constraints, (i) a maximum number of sentences possible in summary and (ii) summary length limit S_{max}, i.e., the summary should be less or equal to 100 words.

The objective function is designed considering three aspects of summarization, i.e., informative and content coverage, which should be maximum, and anti-redundancy, which should be minimum. We have converted the anti-redundancy function to maximization criteria for simplicity. Hence, utilizing these three aspects, the objective function/fitness function is formulated (OF_{score}), which is a maximization criterion. So, the problem is formulated as:

Maximize OF_{score}
Subject to the constraint:
$\sum_{s_i \epsilon S} l_{s_i} <= S_{max}$
Number of summary sentence <= maximum number of sentences possible.

4 Proposed method: ETS using modified binary CSO

This section discusses the proposed modified binary cat swarm optimization-based extractive text summarization approach, which we named ETS-MBCSO. The flow-chart of the ETS-MBCSO is shown in Fig. 1, and the Algorithm 2 shows the steps performed in the proposed system, which are further discussed in this Section.

4.1 Population initialization

Each cat/individual encoded in a binary vector represents a summary in this system. The length of the individual is equal to the total number of sentences present in the document, and each bit/value represents a sentence's presence or absence in it. For example, one of such cat can be represented as [0, 1, 0, 0, 0, 1, 0, 0, 1, 0], indicating 2nd, 6th and 9th sentence from the document consisting of ten sentences are present in the cat (summary). A subset of such feasible cats, satisfying both the constraints (which are discussed in Section 4.2), i.e., summary length limit and maximum sentence possible, are taken as the initial population.

```
Input: Single document
Output: Best cat and system-generated summary
Initialize cat population with the number of cat $|C|$ and cat length $|N|$.
Initialize BCMP of size $|b|$, $G_{max}$, SMP of size $|smp|$, MR, TMP of size (tmp) = (b*2) + 1. All these parameter values are mentioned in Section 5.5.
Calculate $OF_{score}$ for each cat & sort them in descending order.
Select the best b number of cats and store them in BCMP.
for iteration in range (0 to $G_{max}$) do
    Assign TM/SM mode to each cat according to MR.
    for i in range (0 to C-1) do
        if $i^{th}$ cat is in seeking mode(SM) then
            Create smp copies of $i^{th}$ cat and store them in SMP.
            Update SMP cats position by performing the bit-reversal operation.
            For each SMP cat; If it satisfies constraint then calculate $OF_{score}$ else put its $OF_{score}$ = 0.
            Store the current cat into SMP.
            Sort all the SMP cats, and update the $i^{th}$ cat's position.
        else
            Generate 2 children of $i^{th}$ cat for each BCMP cat using crossover & mutation operator of GA & store in TMP.
            For each TMP cat; If it satisfies constraint then calculate $OF_{score}$ else put its $OF_{score}$ = 0.
            Store the current $i^{th}$ cat into TMP.
            Sort all the TMP cats, and update the $i^{th}$ cat's position.
    end for
    Sort all the cats and update BCMP with the current $|b|$ number of best cats.
end for
From BCMP, return the best cat BCMP[0] and its corresponding summary.
Stop.
```

Algorithm 2 Proposed ETS-MBCSO.

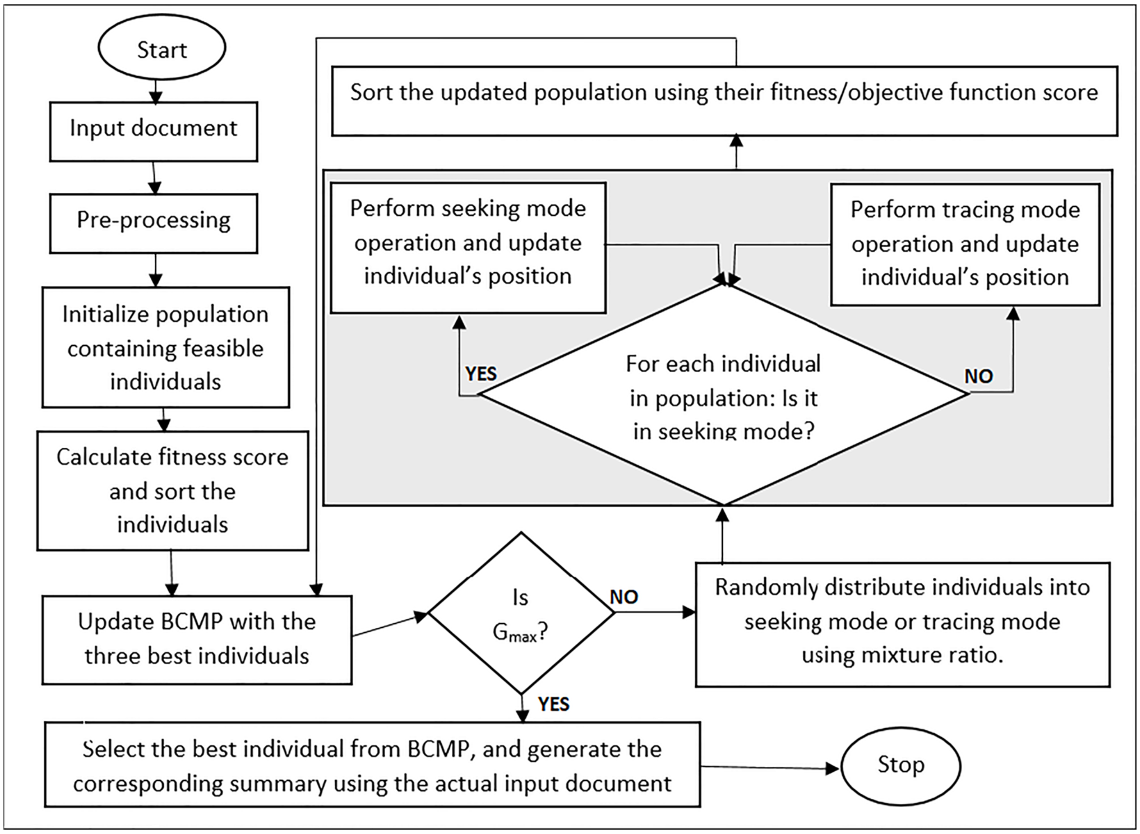

Start
Input document
Pre-processing
Initialize population containing feasible individuals
Calculate fitness score and sort the individuals
Update BCMP with the three best individuals
Is G_{max}?
NO
YES
Randomly distribute individuals into seeking mode or tracing mode using mixture ratio.
For each individual in population: Is it in seeking mode?
YES
NO
Perform seeking mode operation and update individual's position
Perform tracing mode operation and update individual's position
Sort the updated population using their fitness/objective function score
Select the best individual from BCMP, and generate the corresponding summary using the actual input document
Stop

Fig. 1 Flow-chart of ETS-MBCSO, where, G_{max} stands for a maximum number of generations

4.2 Constraint formulation

The constraint concerning these datasets (DUC-2001 and DUC-2002) is the summary length limit, i.e., a 100-word summary. However, we have used another constraint also, for reducing the number of function evolution. Along with the constraint summary length limit, we included another constraint called the maximum number of sentences possible in summary. For which the maximum number of sentences possible in summary based on their length is taken as a constraint. For example, a document has 100 sentences, and the required summary length is 100 word or less. For this, we have calculated sentence length and sorted them in ascending order. Now, suppose the first ten sentences in the sorted list generated 100 or fewer words summary and including the eleventh sentence generates 100 plus words summary. Then we have to take ten sentences as the constraint. i.e., a summary less than or equal to 100 words or ten sentences is feasible and valid. In doing so, we eliminate all the summaries directly, with 11 or more sentences, reducing their feasibility checking cost. However, the individuals' feasibility is checked considering the given constraint, i.e., summary length limit, if it satisfies the constraint "maximum number of sentences possible." Only feasible individuals are considered for objective function calculation or further processing in the proposed system. Therefore, the constraints are checked first after the initial population initialization and in each iteration. After that objective function score is calculated, infeasible individuals are scored as zero, making them least probable.

4.3 Objective/fitness function

After detailed research of existing literature, we observed three main aspects of text summarization, i.e., content coverage, informative, and anti-redundancy, which are the most important aspects to be covered while generating sentence-based extractive summaries. Content coverage is measured concerning various queries, such as the sentence similarity with the document's title, figure caption similarity with sentences, etc. Informative is calculated as the number of clue words, essential and unique terms present in a sentence, etc. Anti-redundancy is measured for assuring no/minimum redundancy in the summary where sentence-to-sentence similarity is considered. After a comparative study, we selected seven features in the proposed approach: key-term wise sentence length, TF−IDF score of 1-gram, 2-gram, & 3-gram, and sentence similarity with the document's title measured using cosine similarity and WMD similarity measure, and anti-redundancy score. Each sentence is represented as a feature vector (the first six features), which in detail are discussed below:

1. Key-term wise sentence length: For each sentence, we have considered the total number of words in a sentence after pre-processing as sentence length.
2. TF-IDF score of n-gram: TF-IDF score helps identify the relevant sentences based on their essential and unique terms. For each sentence, we have measured the 1-gram, 2-gram, and 3-gram TF-IDF scores.
3. Title similarity with a sentence: For measuring the sentence's similarity with the title of the document, we have used Cosine Similarity (CS) [54] and Word Mover Distance (WMD) [55] similarity. CS measures syntactic similarity, whereas WMD similarity measures semantic similarity. Mathematically, CS measures the cosine of the angle between two vectors projected in a multi-dimensional space [54] and WMD is a method that calculates similarity/dissimilarity between the sentences by calculating the distance between two sentences based in a meaningful way even when there are no common words present between them, i.e., semantic similarity. WMD method uses word2vec [56] and a bag of word representations of the sentences. It calculates the total similarity of two sentences by calculating the minimum traveling distance between the relative pair of sentences [55].

These text features are independent of each other. For example, the "key term wise sentence length" ranges from [0, 30], whereas the "Title similarity with a sentence" using WMD similarity measure ranges from [0, 1]. It is observable from the example that the "key term wise sentence length" feature is dominating the "Title similarity with a sentence" feature. Consequently, the result of the system will be biased towards the "key term wise sentence length" feature. For this reason, we have used feature scaling (min_max normalizer) for uniform feature distribution.

4.3.1 Feature scaling using min_max normalizer

Feature scaling helps to represent the feature score uniformly and normalizes all these values to a specific range. Therefore we have used min_max scaling here. Min_max normalizer re-scales all the feature values in a range of [0, 1] as shown in the (4), where, $F_{current}$ is the current feature value which is needed to scale/normalize, F_{max}, F_{min} are the maximum and minimum features score.

$$F_{new} = \frac{F_{current} - F_{min}}{F_{max} - F_{min}} \tag{4}$$

The detailed objective formulation is discussed below:

1. Summary coverage score (SC$_{score}$):

Coverage measures the extent to which the generated summary covers the content. Here, summaries' content coverage is measured in terms of each summary sentence's similarity with the document's title, as shown in (5).

$$SC_{score} = \sum_{i=1}^{M} \frac{\text{WMD}_{sim}\{S_i, T\} + CS_{sim}\{S_i, T\}}{2} \tag{5}$$

Where, S_i is the i^{th} sentence of the summary such that, i = 1, 2, 3, ...M, and M is the number of the sentence in the summary. T is the title/headline of the document.

2. Summary informative score (SI$_{score}$):

The summary's informative summary is measured in terms of knowledge, i.e., how much knowledge bearing each sentence is? We have considered sentence length and TF-IDF scores of 1, 2, and 3-gram features to measure the summary's informativeness in this work, as shown in (6). This objective is also should be maximized.

$$SI_{score} = \sum_{i=1}^{M} (\frac{S_i(L + TI_1 + TI_2 + TI_3)}{4}) \tag{6}$$

Where, L is the length of the i^{th} sentence, and TI_1, TI_2, and TI_3 are the TF-IDF score of 1-gram, 2-gram and 3-gram of the sentence. All these values are scaled values of range [0-1]

3. Summary anti-redundancy score (SR_{score}):

The sentence to sentence similarity/ dissimilarity score is calculated first as shown in the (7). Representing similarity by zero and dissimilarity by one, we have converted this minimization criterion (that is redundancy should be minimum) to maximization criterion (anti-redundancy should be null or minimum).

$$SM(S_i, S_j) = \begin{cases} 0, & \text{if } \sum_{i=1}^{M}\sum_{j=1}^{M} WMD_{sim}(S_i, S_j) \le 0.25 \text{ \& i=j} \\ 1, & \text{otherwise} \end{cases} \tag{7}$$

than, all these values are added up using the (8), where $|M|$ is the number of sentences in summary, SR_{score} is the summary redundancy score measured as the sum of all the summary sentences similarity/dissimilarity score measured using WMD. S_i, S_j are i^{th} and j^{th} sentence respectively.

$$SR_{score} = \frac{\sum_{i=1}^{M} \sum_{j=1}^{M} SM(S_i, S_j)}{2 * M * M} \tag{8}$$

Finally, the Objective/fitness function score (OF_{score}) is calculated by adding the content-coverage, informative, and anti-redundancy scores using the (9). This is a maximization criterion.

$$OF_{score} = (SC_{score} + SI_{score} + SR_{score}) \tag{9}$$

4.4 Best Cat Memory Poll (BCMP)

After forming the initial population and each iteration, the cats' fitness is measured using the objective function score, as shown in (9). After calculating each cat's objective function score, they are sorted in descending order. Chronologically, from the sorted list, the best b number of cats is used to update the BCMP. After trial and error, we have used $b = 3$ in our system. For example, if the size of BCMP is 3. then the best cat is stored in BCMP[0], the next best cat in BCMP[1], and the next best cat in BCMP[2]. These BCMP cats are further used to modify cats' position in tracing mode. After the last iteration, BCMP[0] is used for summary generation.

4.5 Cat's position updation

A cat performs seeking or tracing operations to update its position towards the food. After a cat enters any of these modes, it generates some replicas, allowing it to move further by following some rules for its position updating. After each replica cat updates their respective positions, they (including the current cat) are evaluated and sorted using the objective function scores after satisfying the constraint. And the best-positioned cat is considered as the updated current cat's position. For generating replicas following operations are performed:

4.5.1 Seeking mode operations for position updating

In SM, Seeking Memory Poll (SMP) is created first with smp copies of the current cat's replica. Then bit reversal operation is performed to update their position. After that, all the cats (current and all the updated replica cats) are evaluated (feasibility is checked using constraints, and objective function scores are calculated), and the best cat replaces the current cat. Bit values of a replica cat are updated by reverting the bit position from SRD to CDC. For example, the length of the individuals is ten sentences, i.e., they are of ten bits having the bit position [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]. Let a cat Cat_i having bit position [1, 1, 0, 0, 0, 1, 0, 0, 0, 1], is performing seeking mode operation, and two replica cats Cat_x, Cat_y are generated from Cat_i. Initially, they have the same bit values as Cat_i. Now, we applied the bit reversal technique to update their position in SM. The first two random values, CDC and SRD, are generated in a range [0-10(length of an individual)]. Suppose for Cat_x, SRD =5 and CDC=2, then updated bit position after reversal will be [7, 6, 5, 3, 4, 2, 1, 0, 9, 8]. So, Cat_x's position will be [0, 0, 1, 0, 0, 0, 1, 1, 1, 0]. Similarly, CDC and SRD have generated again for cat_y, let the new CDC = 4 and SRD =8. Then the updated bit position after reversal will be [0, 1, 2, 3, 8, 7, 6, 5, 4, 9]. So, Cat_y's position will be [1, 1, 0, 0, 0, 0, 0, 1, 0, 1]. Hence from the example, it is visible that all the replicas have an equal number of zeros and ones after position updating. This shows that we are checking the summaries of the same length (number of sentences). In this example, a subset of four sentenced length individuals is searched for position updating, which can be considered local optima searching.

4.5.2 Tracing mode for position updating

When a cat opts for performing TM operation, its new possible positions in n-dimensional space, i.e., child cats, are created first. Then the new possible positions, including the current position, are evaluated. Among them, the best position is considered the updated position of the current cat. For this purpose, the following steps are performed: The current cat and the BCPM cats participate in the matting poll. For each BCMP cat, two children (possible moves) are generated, i.e., for BCMP equals three, six cats are generated, and hence a total of seven cats are evaluated. For generating the new cat, we have used genetic algorithms [57] arithmetic crossover and mutation. For each BCMP cat, two children are generated; hence two times, these crossover and mutation are performed, where the first AND operation is performed which is the crossover operation. Mutation operation is performed then, i.e., 20% of the bits are randomly selected, and then bit inversion is performed. For example a cat cat_i (0 0 1 0 1 1 0 0 0 1) is performing TM operation and cat_{bi} (1 0 1 0 1 1 0 0 0 1) is one of the BCMP cat. Then after performing the AND operation, we obtained the offspring cat cat_c [0 0 1 0 1 1 0 0 0 1]. After that, bit inversion is applied. Here we have used 10-bit length individuals. Hence 20% of 10 is 2. So, two bits are randomly selected. Let the randomly selected bits be the 5th and 6th bit. Then after performing the mutation operation it will be cat_i [0 0 1 0 1 0 1 0 0 1].

4.6 Population for the next generation

The exact initial cats with their updated positions perform in consecutive iterations. In each iteration, each cat moves towards optimality. The BCMP also gets updated as mentioned in the Algorithm 2.

4.7 Termination criteria

The cat's position updation by performing seeking/tracing mode operations and BCMP updation continue until the maximum number of generations (G_{max}) is reached. The final BCMP contains the three best cats.

4.8 Summary generation

The proposed system is a single objective optimization system, so optimal cat exists. Hence, after the last iteration, the best cat is selected as a summary candidate.

5 Experimental setup

The experimental setup section presents the following things: (i). the dataset details, (ii). the pre-processing steps, (iii). the overview of ROUGE evolutionary metrics, (iv) list containing existing methods names, which are used for comparison, (v) parameter settings, (vi) system descriptions, (vii) description of other performance measures including (improvement obtained, generational distance, cohesion, CPU utilization, and ranking method), and (viii) statistical tests details.

5.1 Data-set description

We have used DUC-2001 and DUC-2002 data set from DUC[1] (Document Understanding Conferences) Corpus. From these data sets, 30 and 59 topics containing 309 and 567 unique news articles and their gold summary are used for evolutionary purposes. These actual summaries are 100 words.

5.2 Pre-processing

In the pre-processing module, we performed the following steps:

1. From each input document, we extracted the article's title and body as plain text.
2. Plain text and title are then segmented as sentences and tokenized. We further extracted:
    - N = total number of sentences,
    - Actual sentence length, which is the total of the word in each sentence,
3. Lower cased the plain text and title and noise, such as punctuation marks, stop words, HTML tag, hyperlink, etc., are removed.

5.3 ROUGE evolutionary metrics

For summary evaluation, we have used ROUGE matrices. Most existing systems provide ROUGE 1 and 2 recall scores; hence, we have used these scores for an evolutionary purpose. The recall score is calculated as the total number of overlapping n-grams divided by the n-gram in the reference/actual summary.

5.4 Comparing methods

The obtained results of the proposed approach, ETS-MBCSO, are compared with the various existing approaches, namely, PSOGA-BKSum [45], ESDS-MCSO [12], ESDS-AMGA2 [40], ESDocSum [37], ESDS-SMODE [17], COSUM [2], DE [36], FEOM [53], ETS-GA [35], MA-SingleDocSum [34], ESDS-GHS-GLO [32], DPSO [11]. These approaches are discussed in Section 2.1.

5.5 Parameter settings

The parameters used in the proposed framework are mentioned in this Section. These parameters along with their values are: number of cat $|C|$ = 20 or more (variable length), BCMP size (b) = 3, TMP size (tmp) = 7, G_{max} = 50, mutation probability = 0.2, SMP size (smp)= 11, MR = 0.5, CDC and SRD are two randomly generated number in a range (0 to (cat size-2)) for each iteration. However, it is also noticed that the system predicts most of the optimal cat in the iteration (20-25) approximately. We have used these parameters because, in most of the existing literature [2, 10, 36, 37], similar weight values are considered. Further, we applied trial and error before consideration of the values. Ten runs were conducted on each data set, and the results were averaged and presented.

5.6 System description

We have built four systems to show the importance of feature scaling and each objective function's weight. All these systems have used the same above-mentioned underlying strategy (ETS-MBCSO). They are: (i). System-1: the system used the same set of features (as mentioned in the proposed system) except the feature scaling. (ii). System-2: With features scaling, preference is given to the coverage and informative objective. Hence, the objective function is formulated as (2 * (coverage + informative) + anti-redundancy). (iii). System-3: With features scaling, priority is given to the anti-redundancy objective; hence,

[1] http://duc.nist.gov/

the objective function is formulated as ((coverage + informative) + (2 * anti-redundancy)), and (iv). System-4: Features scaling is used, and all the objectives are given equal preference. All these systems' functionalities are briefly described in Table 1, which are tested on ten documents from the dataset. The obtained results are discussed in the result analysis section.

The proposed approach(ETS-MBCSO) is also compared with the classical CSO [27], binary PSO [58], DE, GWO, AMGA2 [40] and GA [59] based extractive text summarization approach. The same methodology (population, objective function, termination condition, etc.) is used in designing all these algorithms for text summarization. Only the baseline systems' framework and child generation strategies are directly used.

5.7 Other performance measures

We have also evaluated the proposed system based on other performance measures to validate its superiority. These measures are Generational Distance (GD) [60], cohesion, readability, and CPU time (total time taken by the CPU to generate the summaries, i.e., process_time()). We have also analyzed the errors or possible reasons because of why the proposed system does not achieve 100% accuracy. These measures are briefly discussed below:

1. Improvement Obtained (IO): The percentage of improvement obtained by the proposed system over each existing system based on the ROUGE scores is calculated. Mathematically, IO is calculated using the (10), where PM is the proposed method's ROUGE score, and OM is the other method's ROUGE score.

$$IO = \frac{PM - OM}{OM} * 100 \tag{10}$$

2. Generational Distance (GD): GD scores help us predict the semantic similarity between obtained and actual summaries. With modification, we have used the GD formula [60]. Here we have calculated one-to-one semantic similarity using the WMD similarity measure. For example, if the actual summary has four sentences, we have checked how many of these sentences are unique and correctly predicted (semantically). Suppose two sentences are predicted; then GD is 0.5 or 50%. Mathematically, GD in percentage is calculated using the (11), where O is the number of unique sentences in the obtained summary, which are one-to-one mapped to the actual summary sentences, and A is the number of actual summary sentences.

$$GD = \frac{O}{A} * 100 \tag{11}$$

3. Cohesion: cohesion ensures readability by showing the similarity percentage between summary sentences. We used the WMD similarity measure and considered 50% and above similarity, as connected because 75% and above similar sentences are considered redundant sentences and are barely present in the optimal summaries. Let N be the number of sentences in the obtained summary. Then N-1 sentence pairs are checked, as only consecutive sentence pairs are checked; suppose for S_i, we have calculated the similarity score for the pair(S_i, S_{i+1}). Now let M be the total number of similarities found. Then mathematically, cohesion is measured as shown in (12)

$$Cohesion = \frac{M}{N-1} * 100 \tag{12}$$

4. Computational time (CPU time): It is the system's total time (in a second) taken to generate a summary for a given document. Time elapsed during sleep is not included. (Note: for the experimental purpose, we have used Lenovo IdeaPad 5i, Intel 11$^{\text{th}}$ Generation i5 Processor.
5. Ranking of methods (ROUGE scores): We have ranked different methods (including the proposed method). The ROUGE scores from Table 4 are used for this purpose. This ranking strategy is used by [36, 37], and calculated using the (13), where M is the number of methods, R_p is the number of times a method came in p^{th} position.

$$Ranking_{score} = \sum_{i=1}^{M} \frac{(M - p + 1)R_p}{M} \tag{13}$$

5.8 Statistical tests

1. t-test: For proving that the system is statistically significant i,e, it rejects the null hypothesis, we have conducted a two-sample t-test [61] at 5% confidence level. We have considered 5% because [9, 17, 37] also

Table 1 Systems description

System Name	System Description
System-1	All the three aspects are given equal preference, but without features scaling.
System-2	With features scaling, preference is given to the coverage and informative aspects.
System-3	With features scaling, preference is given to the anti-redundancy aspects.
System-4	All three aspects are given equal preference, and features scaling is also used.

used the same significance level. For this purpose, two hypotheses are considered. The NULL hypothesis states no significant differences between the mean values of the tested algorithms, and the alternate hypothesis states there are substantial differences. We have computed the sample means from two groups (group-1, group-2 contains the list of scores(ROUGE-1, ROUGE-2) of our approach and the exiting methods, respectively) and derived the conclusion for the population's means.

2. ANOVA test: ANOVA test [62] is also used to check the statistical significance. The same hypothesis of the t-test is considered. We have used the existing system's ROUGE scores with some variations to generate their respective mean and taken the mean of ten runs of the proposed system (ETS-MBCSO). All the existing systems mentioned in Section 5.4 are used for this purpose.

6 Result analysis

This Section discusses proposed and existing systems outputs in terms of ROUGE scores, systems performance based on various performance measures, obtained summary analysis, the system's convergence towards optimality, and statistical analysis.

6.1 Performance based on objective function formulation

Four systems are developed and tested on ten randomly selected documents from the data set: DUC-2001 and DUC-2002, respectively, to show the effectiveness of the proposed system's objective function formulation. In these systems, the same above-mentioned underlying strategy (ETS-MBCSO) is used, and only the fitness function is changed, considering each aspect's weightage and features scaling. Systems are briefly described in the system description Section 5.6. Results are shown in Table 2, which are the ROUGE score-wise average score. The ROUGE scores from Table 2 are used for visualization using box-plot, which are shown in Fig. 2. The result indicates that without features scaling, the system under-performs. System-4 is the best-performing system; the box plot shows that System-4 has the highest median value. However, we have analyzed these obtained summaries and found that in System-3, some of the sentences are redundant though they are content-related and informative. We have used System-4 as our underlying approach to generate summaries, which we called ETS-MBCSO.

6.2 System's performance concerning existing algorithms (classical CSO, GA, PSO, AMGA-2, DE, and GWO)

Based on Table 2, we identified that ETS-MBCSO (system-4) is the best-performing system among these systems. However, to know the performance of MBCSO, we have again compared the ETS-MBCSO with the binary CSO, GA, PSO, AMGA-2, DE, and GWO-based ETS approaches. Brief assumptions are mentioned in Section 5.6. These algorithms are tested on five documents from DUC-2001 by conducting ten runs for each. Average ROUGE-1(ROUGE-2) scores are reported and further averaged to generate a single score. Table 3, shows the obtained results and Fig. 3 shows the box-plotted values. From the results, it is visible that the ETS-MBCSO outperformed all these algorithms here. The box-plotted values from Fig. 3 show the ETS-MBCSO has the highest median value.

6.3 System's performance concerning existing state-of-the-art approaches

Table 4 presents the proposed systems and twelve other existing state-of-the-art systems ROUGE recall scores on DUC-2001 and DUC-2002 data sets. We also have box-plotted these ROUGE scores and illustrated them in Fig. 4. These scores are calculated by conducting ten runs on each document. The percentage of improvement is also presented in Table 5. From Tables 4, 5, and Fig. 4, it is visible that the proposed system outperformed all the existing approaches

Table 2 Table shows different system's performance

Methods	DUC 2001		DUC 2002	
	ROUGE 1	ROUGE 2	ROUGE 1	ROUGE 2
System-1	0.46556	0.20546	0.46543	0.22333
System-2	0.59488	0.31897	0.61843	0.30963
System-3	0.57251	0.30897	0.59455	0.29968
System-4	0.60879	0.31615	0.62265	0.32983

These systems are mentioned in Section 5.6

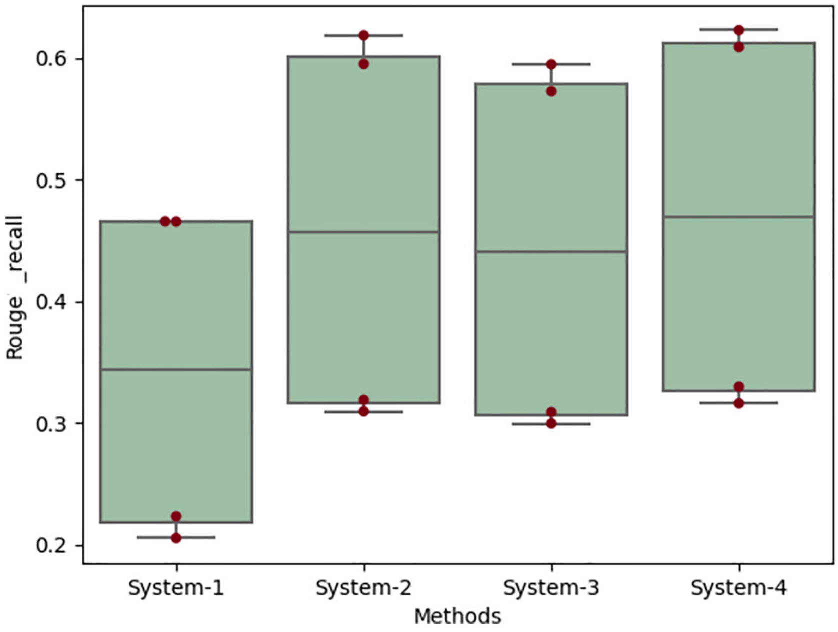


Fig. 2 Box-plot showing ROUGE scores of Table 2

mentioned in this paper on both the datasets when measured with ROUGE measures.

6.4 Performance analysis based on ROUGE scores and other evolutionary measures

We analyzed the performance using various parameters such as generational distance (GD), cohesion, readability, and CPU time to check the quality of the proposed system's obtained summaries and system performance. All these performance measures are described in the Experimental setup Section 5.7. We have used fifty randomly selected outputs. These outputs are taken from the previous ten runs of system execution. Utilizing these outputs, we have measured GD, CPU time, and cohesion, which are further averaged and shown in Table 6.

From the obtained results and analyzing the other performance measures (GD, CPU time, cohesion), it can be concluded that the proposed system performs well regarding ROUGE scores, GD, CPU time, and cohesion. The obtained result also ensures that the system-generated summaries are readable as the sentences are taken chronologically from the document. Further to show the goodness of generated summaries, we have

Table 3 Table shows the best system's performance against some similar algorithms

	Document_name	DOC-1	DOC-2	DOC-3	DOC-4	DOC-5	Average
GA	ROUGE-1	0.6154	0.5813	0.6053	0.6389	0.6777	0.62372
	ROUGE2	0.2973	0.2941	0.3099	0.3132	0.3623	0.31536
PSO	ROUGE-1	0.6219	0.6723	0.6414	0.6249	0.6838	0.64886
	ROUGE2	0.3863	0.3333	0.3429	0.3262	0.3317	0.34408
Classic CSO	ROUGE-1	0.6362	0.6328	0.6565	0.6875	0.6613	0.65486
	ROUGE2	0.2725	0.3401	0.3448	0.3125	0.3519	0.32436
AMGA-2	ROUGE-1	0.6033	0.6113	0.6324	0.6225	0.6877	0.63144
	ROUGE2	0.2873	0.2992	0.3201	0.3099	0.3523	0.31376
DE	ROUGE-1	0.6257	0.6586	0.6652	0.6893	0.7289	0.67354
	ROUGE2	0.2816	0.3371	0.3354	0.3554	0.4185	0.3456
GWO	ROUGE-1	0.6089	0.6551	0.6614	0.6348	0.6989	0.65182
	ROUGE2	0.3163	0.3442	0.3499	0.3185	0.3747	0.34072
MCSO(proposed)	ROUGE-1	0.6386	0.676	0.6715	0.6989	0.7667	0.69034
	ROUGE2	0.3265	0.3359	0.3104	0.3772	0.4352	0.35704

These algorithms are mentioned in Section 5.6

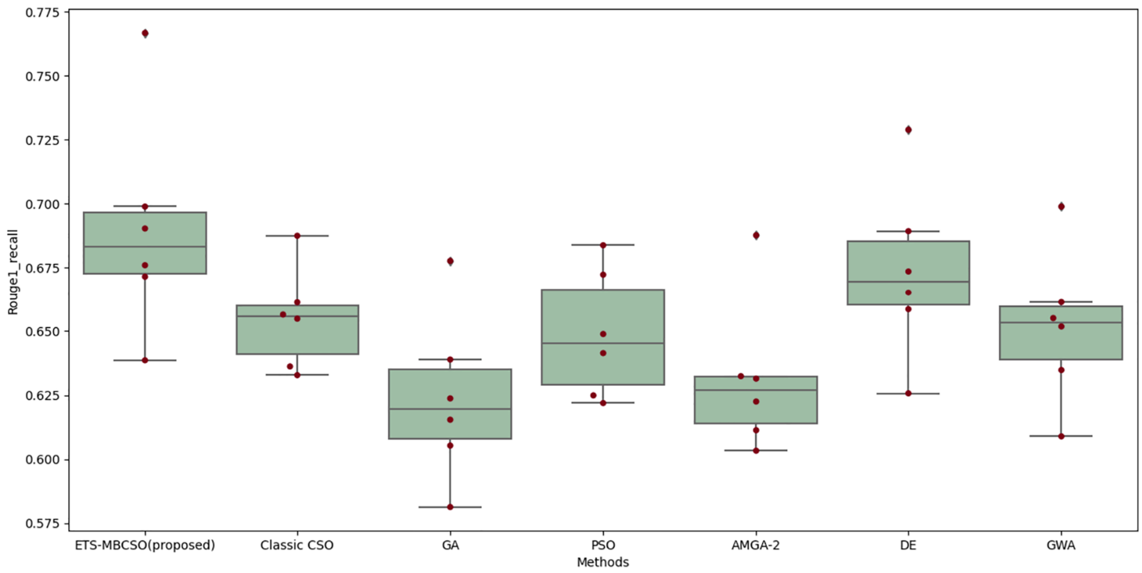


Fig. 3 Box-plot showing ROUGE scores of GA, PSO, classical CSO, and ETS-MBCSO

presented two actual vs. system-generated summaries In Fig. 5. The first summary pair is from document DUC-2001/d11b/ AP890403-0123, and the second summary pair is from document DUC-2001/d22d/AP880705-0109. Both the summaries are similar, except the actual summary has two extra sentences. In the second document, three sentences are similar out of five. However, though the obtained summaries are not exactly the same as actual summaries but, the obtained summaries are found to be readable and relevant.

We also observed that all other summaries are relevant and readable, similar to these two summaries. Though, some summaries are found with low ROUGE scores. One such summary is shown in Fig. 6. Figure 6 shows that three similar sentences are detected, but two sentences are entirely different. But, after analyzing such low ROUGE score summaries, we infrared that this obtained summary is still readable and relevant.

However, after analyzing the ROUGE scores, GD score, CPU time utilization, readability, and cohesion of the obtained summaries, the following observations were made:

1. Most of the obtained summaries are good regarding ROUGE scores, GD, and cohesion.

Table 4 Table showing ROUGE-recall scores

Methods	DUC-2001		DUC-2002	
	ROUGE-1	ROUGE-2	ROUGE-1	ROUGE-2
ETS-MBCSO (proposed)	0.65124	0.3228	0.67333	0.34985
PSOGA-BKSum [45]	0.5454	0.2625	0.5699	0.275
ESDS-MCSO [12]	0.51944	0.3042	0.53686	0.3042
ESDS-AMGA2 [40]	0.50769	0.29506	0.52581	0.31287
ESDocSum [37]	0.50236	0.29238	0.51662	0.28846
ESDS-SMODE [17]	0.45214	0.21450	0.49117	0.34132
COSUM [2]	0.47270	0.20120	0.49080	0.23090
DE [36]	0.47856	0.18523	0.46694	0.12368
FEOM [53]	0.47728	0.18549	0.46575	0.12490
ETS-GA [35]	0.45058	0.19619	0.48423	0.22471
MA-SingleDocSum [34]	0.44862	0.20142	0.48280	0.22840
ESDS-GHS-GLO [32]	0.45402	0.19565	0.47896	0.22138
DPSO [11]	0.39930	0.08320	0.41720	0.10260

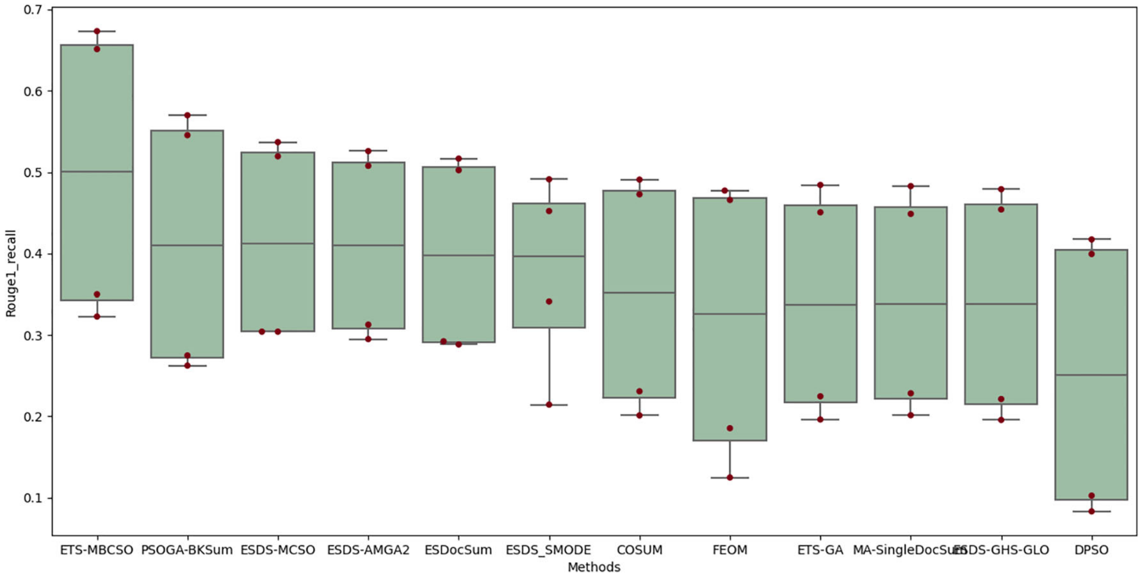


Fig. 4 Box-plot showing various methods(Proposed and existing methods)

2. All the system-generated summaries are readable because their sentences are taken chronologically as they were in the document.
3. Smaller document produces a better summary than a lengthy document.
4. The obtained summaries are extractive, i.e., the sentences are directly selected from the documents. So, 100% accuracy is reasonably not possible.
5. It is also observed that all the actual summaries are human-generated abstractive summaries. Hence they may contain a human error.

6.5 Ranking of text summarization approaches

We have also ranked the methods(proposed and existing methods) based on their ROUGE scores. We have ten methods, each having four values (ROUGE recall scores of both the data sets). Methods $ranking_{scores}$ are calculated using the (13) and presented in the Table 7. The obtained values are sorted in descending order, and methods are ranked. From Table 7, it can be interpreted that the proposed method (ETS-MBCSO) has been selected as rank 1.

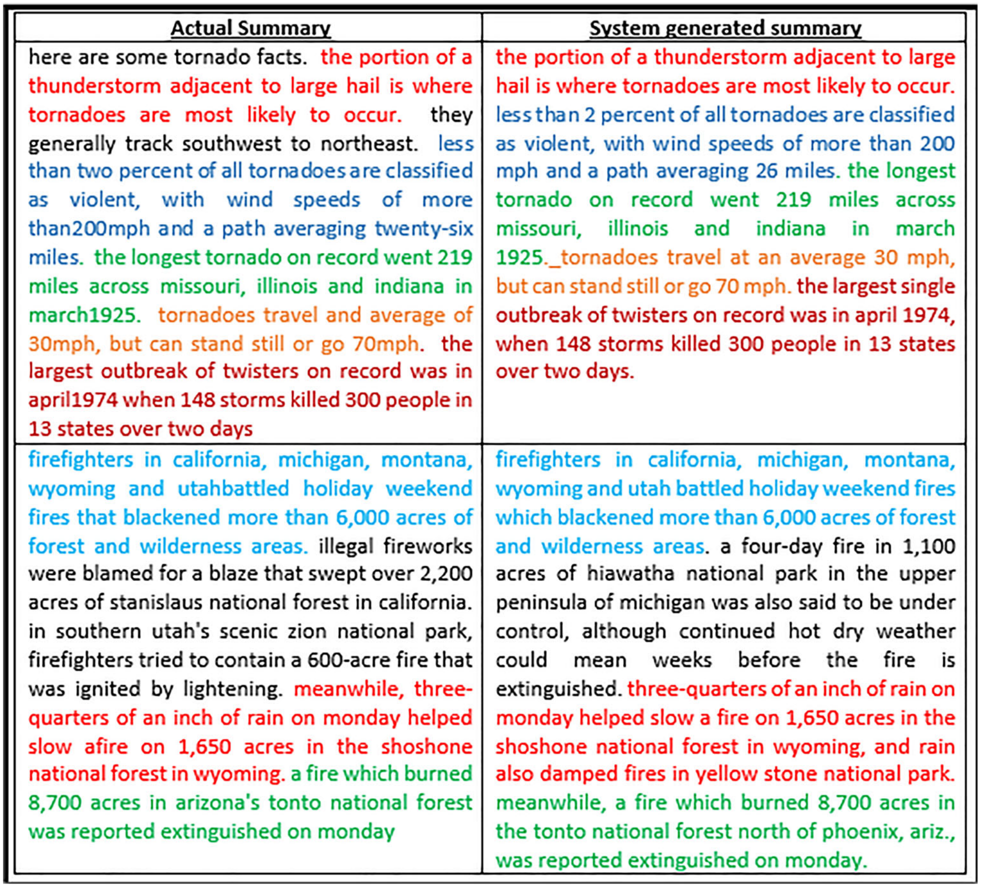
Table 5 Table shows the percentage of improvement as mentioned in the Section 5.4

Methods	DUC-2001		DUC-2002	
	ROUGE-1	ROUGE-2	ROUGE-1	ROUGE-2
ETS-MBCSO (proposed)	XXX	XXX	XXX	XXX
PSOGA-BKSum	19.40594059	22.97142857	18.14879803	27.21818182
ESDS-MCSO	25.37347913	6.114398422	25.42003502	15.00657462
ESDS-AMGA2	28.27462825	9.39951629	28.05504096	11.82063741
ESDocSum	29.63611753	10.40426842	30.33370756	21.28198017
ESDS-SMODE	44.0350334	50.48951049	37.08695564	2.499121059
COSUM	37.77025598	60.43737575	37.19030155	51.51580771
DE	36.08324975	74.2698267	44.20053968	182.8670763
FEOM	36.4482065	74.02555394	44.56897477	180.1040833
ETS-GA	44.5337121	64.53437994	39.05169031	55.68955543
MA-SingleDocSum	45.1651732	60.26213881	39.46354598	53.17425569
ESDS-GHS-GLO	43.43861504	64.98849987	40.58167697	58.03143915
DPSO	63.09541698	287.9807692	61.39261745	240.9844055

Table 6 Average generational distance (GD), CPU time, Cohesion, and ROUGE 2 score of the summaries are shown

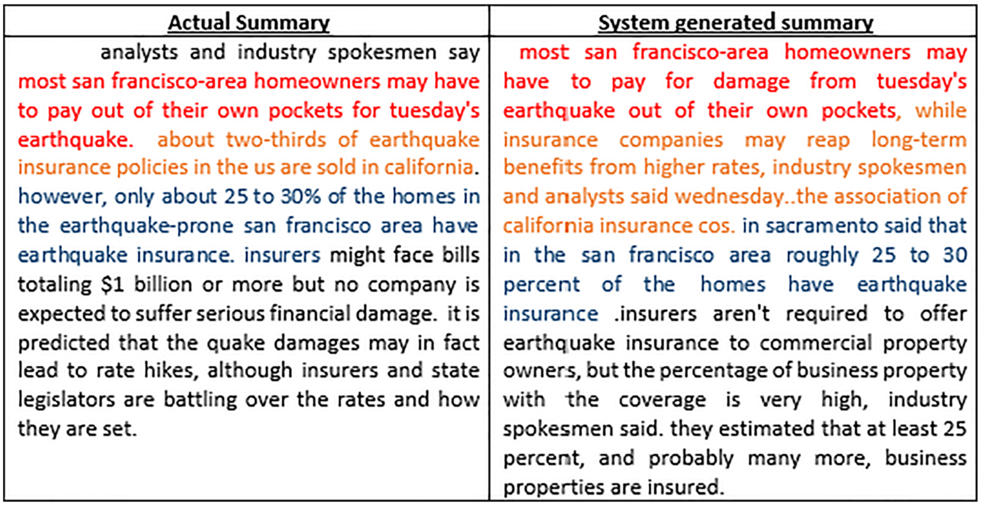
Data-set name	GD	Cohesion	CPU Time (in sec)	ROUGE-2 score
DUC-2001	0.592	0.624	70.83	0.6997150997150997
DUC-2002	0.637	0.645	65.50	0.6320593268682259

Fig. 5 Example of good-quality summary: Two actual vs. obtained summaries are presented



Actual Summary	System generated summary
here are some tornado facts. the portion of a thunderstorm adjacent to large hail is where tornadoes are most likely to occur. they generally track southwest to northeast. less than two percent of all tornadoes are classified as violent, with wind speeds of more than200mph and a path averaging twenty-six miles. the longest tornado on record went 219 miles across missouri, illinois and indiana in march1925. tornadoes travel and average of 30mph, but can stand still or go 70mph. the largest outbreak of twisters on record was in april1974 when 148 storms killed 300 people in 13 states over two days	the portion of a thunderstorm adjacent to large hail is where tornadoes are most likely to occur. less than 2 percent of all tornadoes are classified as violent, with wind speeds of more than 200 mph and a path averaging 26 miles. the longest tornado on record went 219 miles across missouri, illinois and indiana in march 1925._tornadoes travel at an average 30 mph, but can stand still or go 70 mph. the largest single outbreak of twisters on record was in april 1974, when 148 storms killed 300 people in 13 states over two days.
firefighters in california, michigan, montana, wyoming and utahbattled holiday weekend fires that blackened more than 6,000 acres of forest and wilderness areas. illegal fireworks were blamed for a blaze that swept over 2,200 acres of stanislaus national forest in california. in southern utah's scenic zion national park, firefighters tried to contain a 600-acre fire that was ignited by lightening. meanwhile, three-quarters of an inch of rain on monday helped slow afire on 1,650 acres in the shoshone national forest in wyoming. a fire which burned 8,700 acres in arizona's tonto national forest was reported extinguished on monday	firefighters in california, michigan, montana, wyoming and utah battled holiday weekend fires which blackened more than 6,000 acres of forest and wilderness areas. a four-day fire in 1,100 acres of hiawatha national park in the upper peninsula of michigan was also said to be under control, although continued hot dry weather could mean weeks before the fire is extinguished. three-quarters of an inch of rain on monday helped slow a fire on 1,650 acres in the shoshone national forest in wyoming, and rain also damped fires in yellow stone national park. meanwhile, a fire which burned 8,700 acres in the tonto national forest north of phoenix, ariz., was reported extinguished on monday.

Fig. 6 An example of low-quality obtained summary



Actual Summary	System generated summary
analysts and industry spokesmen say most san francisco-area homeowners may have to pay out of their own pockets for tuesday's earthquake. about two-thirds of earthquake insurance policies in the us are sold in california. however, only about 25 to 30% of the homes in the earthquake-prone san francisco area have earthquake insurance. insurers might face bills totaling $1 billion or more but no company is expected to suffer serious financial damage. it is predicted that the quake damages may in fact lead to rate hikes, although insurers and state legislators are battling over the rates and how they are set.	most san francisco-area homeowners may have to pay for damage from tuesday's earthquake out of their own pockets, while insurance companies may reap long-term benefits from higher rates, industry spokesmen and analysts said wednesday..the association of california insurance cos. in sacramento said that in the san francisco area roughly 25 to 30 percent of the homes have earthquake insurance .insurers aren't required to offer earthquake insurance to commercial property owners, but the percentage of business property with the coverage is very high, industry spokesmen said. they estimated that at least 25 percent, and probably many more, business properties are insured.

Table 7 Ranking of different methods based on their ROUGE-1 and ROUGE-2 scores

Methods	Rp													Ranking_score	Rank
Methods	1	2	3	4	5	6	7	8	9	10	11	12	13	Ranking_score	Rank
ESDS-MBCSO (proposed)	4													4	1
PSOGA-BKSum		2			1	1								3.154	4
ESDS-MCSO		1	2	1										3.384	2
ESDS-AMGA2			2	2										3.231	3
ESDocSum				1	3									2.846	5
ESDS_SMODE		1				2				1				2.461	6
COSUM							2	2						2	7
DE						1					1	2		1.154	9
FOEM							1				2	1		1.154	10
ETS-GA								1	2		1			1.462	11
MA-SingleDocSum							1	1	1			1		1.839	8
ESDS-GHS-GLO									1	3				1.307	12
DPSO													4	0.308	13

6.6 Possible reasons behind achieving better results with respect to existing state-of-the-art approaches

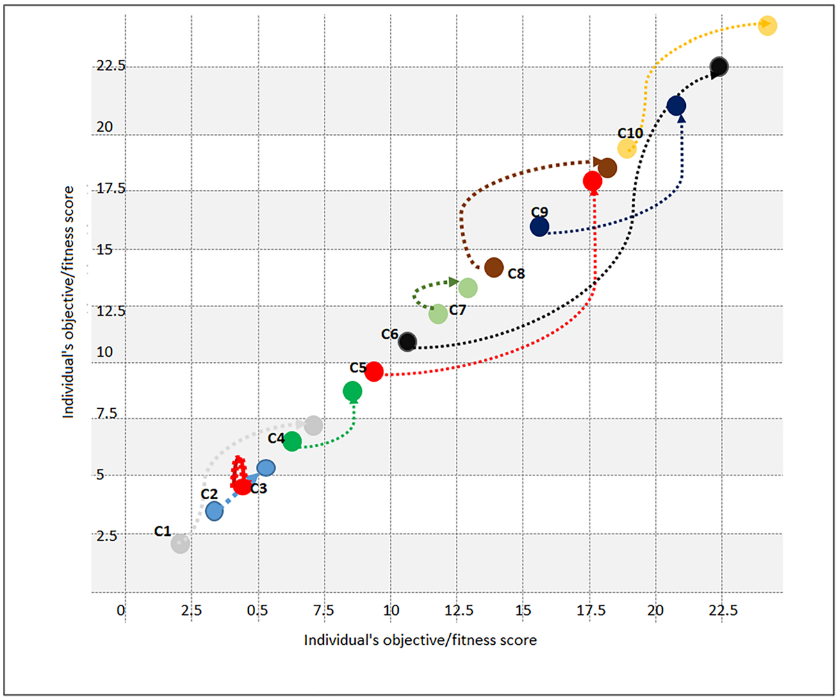
After visualizing the results, we found that our approach outperformed the existing state-of-the-art systems. Here, we have enumerated some of the possible reasons behind achieving better results concerning the current methods are:

1. Modules are designed independently, such that, in need, any of these modules can be modified easily without much altering the other.
2. The pre-processing module's outputs are clean query and text, which are the most suitable inputs for performing feature extraction.
3. The constraint formulation (maximum number of sentences in summary from the summary length limit constraint) reduced the number of function evaluations as the summaries containing more sentences than the constraint value are never included or evaluated in the process [12]
4. The minimum feature set is used to represent each sentence. Which are further scaled in a range [0-1] for uniformity. In contrast, most of the existing system used the weighted sum of features directly in their system [12, 32, 34, 45].
5. Both WMD and cosine similarity measure is used. Hence, we improved our result as compared to [2, 5, 37] where they have considered only syntactic similarity.
6. Our method is fully unsupervised compared to many existing methods [17], where actual summaries are used for system summary generation.
7. In the proposed system, three prime aspects of summarization are considered for the objective formulation, along with functionalities like features scaling and semantic and syntactic similarity measure. In contrast, the systems like FEOM, DE, MA-SingleDocSum used weighted features score as the objective function [34, 36, 53] and in ESDS-SMODE [17] and COSUM [2], objective functions used are clustering indices to detect the number of clusters presence.
8. In our approach, we used CSO as an underlying optimization algorithm that performs better as compared to many evolutionary optimization algorithms as reported by [26, 28] etc. Though CSO had the only drawback of sticking to the local optimums, we modified its seeking and tracing mode operation such that it no longer sticks to local optimums.

To show that cat updates their position in multi-directional space in iterations, we represented the initial vs. final cat population's position in Fig. 7. These are the iteration-wise objective function score of each cat of the document DUC-2001/d04a/FT923-5089 . Ten individuals (cat) is found to be feasible in this execution, which we have marked with ten different colors and marked as C1, C2 to C10. The arrow shows the moves of each cat (the same color represents the initial and final positions for each cat). It can be observed that each cat either updated its position towards optimality or remained constant but did not downgrade its position. All the cats expect C3 have updated their position. Only C3 remained in its position still the last iteration. For this document, the yellow-colored cat is considered an optimal cat with an objective function score of 24.3 (marked with yellow), which we have further used to generate the summary.

We have also presented iteration wise ROUGE-1 and ROUGE-2 scores for the same document (DUC-2001/d04a/FT923-5089) in Fig. 8 and fifty documents

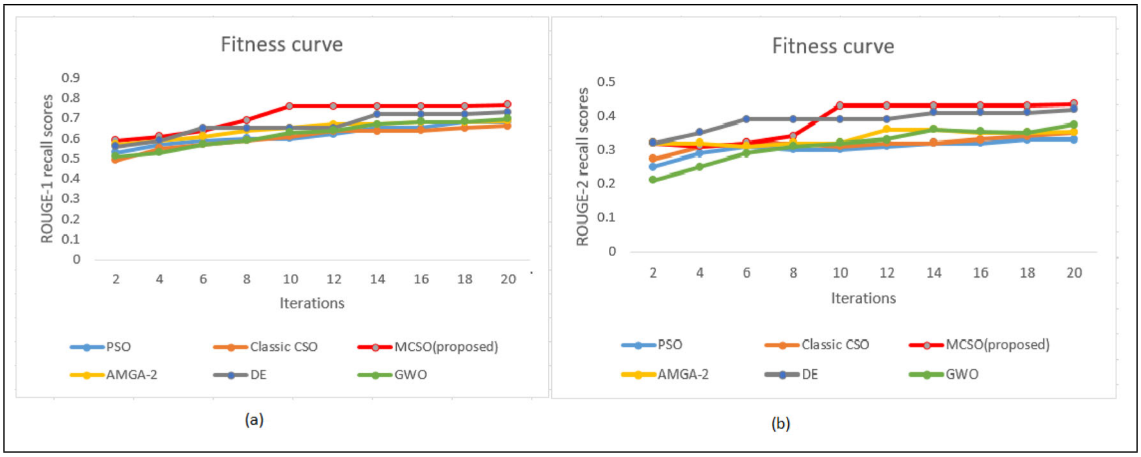
Fig. 7 An example of the initial VS final iteration's cat position



iteration wise average ROUGE-1 and ROUGE-2 scores of Fig. 9. After each iteration, we have taken the best individual from BCMP and calculated its ROUGE scores with respect to the actual summary in Fig. 8. Similarly, we have generated iteration-wise ROUGE-1 and ROUGE-2 scores from fifty documents and then averaged their scores. These averaged scores concerning all the existing and proposed approaches (classical CSO, binary PSO, DE, GWO, AMGA2, GA, and the proposed approach) are presented in Fig. 9. In the Figs. 8 and 9, Figure (a). presents the ROUGE-1 scores and Figure (b). presents ROUGE-2 scores from Figs. 8 and 9, it can be observed that with iteration, the ROUGE scores are improving; however, in some cases, they are also decreasing. Even if the objective function score is higher than the previous iteration, their ROUGE scores are decreased. The objective function scores are based on some aspects of summarization, but the actual summaries are human-generated. Hence, they are not accurate. From these figures, we have observed the proposed system convergences at 10th iteration, whereas others converge after the 14th or after that.

6.7 Statistical significant test results

To prove that the obtained results are statistically significant, we have conducted some statistical tests discussed in



(a)

(b)

Fig. 8 Fitness curve with the ROUGE scores (Figure (a). ROUGE-1 scores and Figure (b). ROUGE-2 scores) VS number of iterations for a single document obtained by various approaches

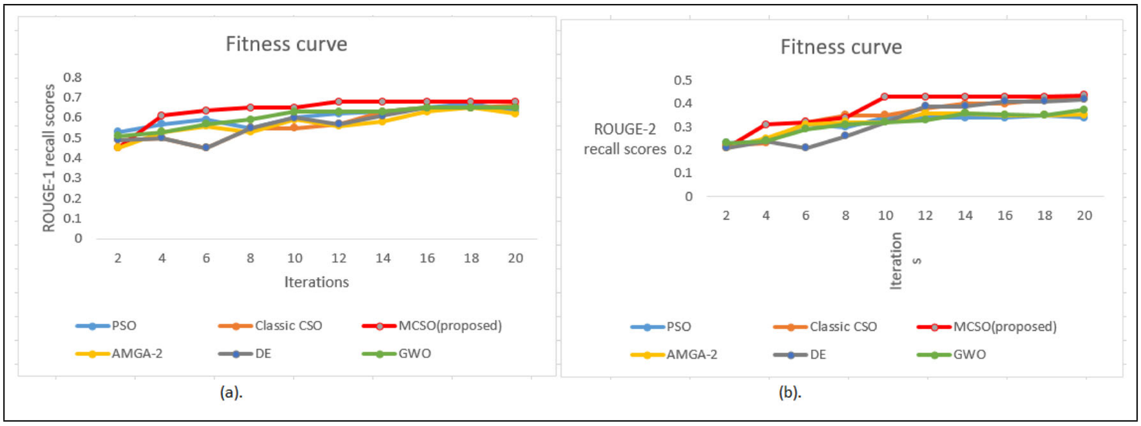


Fig. 9 Iteration-wise average ROUGE scores(Figure (a). ROUGE-1 scores and Figure (b). ROUGE-2 scores) of fifty documents obtained by existing and proposed approaches

Section 5.8. For performing these tests, we have conducted ten runs. We have considered ten runs for other existing systems by varying the given ROUGE score such that the average score is their reported score. The p-values obtained from the two-sample t-test and the one-way ANOVA test are shown in Table 8. In Table 8, we have also applied the Mann-Whitney test, which is also known as the Wilcoxon rank sum test, such that null hypothesis $\mu 0 = 0.0$, an alternative hypothesis is two-sided, i.e.,$\mu \neq \mu 0$. Test statistic W is 16, p-value obtained is 0.028571.

The 95% confidence interval is between 0.0001 – 0.0004, and the sample estimate of the difference in location obtained μ is 0.00020008. These values are lower than 0.05; hence, it suggests that one or more methods are significantly different, further rejecting the null hypothesis.

7 Conclusion

This paper has presented an automatic summary generation technique utilizing the Modified Binary Cat Swarm Optimization (ETS-MBCSO) approach. The proposed approach is generic. Relevant objective functions are identified for a text, and a set of binary vectors representing feasible summaries are chosen as the initial cat population. Then in each iteration, cat positions are updated by performing seeking or tracing mode operation, and BCMP is updated using the current three best cats. Finally, the optimal cat is chosen as a summary representative after all the iterations. The obtained result proves the proposed system's efficiency compared with the various state-of-the-art methods. Also, the system-generated summaries are found readable and cohesive. Statistical tests are also conducted. In all these tests, we have obtained p-values lower than 0.05, rejecting the NULL hypothesis and suggesting that the proposed system is statistically better and more significant.

However, based on the output, it can be concluded that there is still ample scope for improvement in the results, which can be considered for our future work. Further, the authors will study the effects on the approach's performance by using other sentence scoring schemes, similaritydissimilarity measures, etc. The objective function formulation can be considered a future direction of work. We planned to apply a multi-objective optimization algorithm to solve the summarization problem by considering each objective independently or grouping a subset of the objective functions. We also plan to use the

Table 8 P-values based on two-sample t-test, one way ANOVA test

Dataset	Comparing measures	Two sample t-test	One way ANOVA test
DUC-2001	ROUGE-1	0.00013318	3.80E-22
DUC-2001	ROUGE-2	0.00043209	1.11E-16
DUC-2002	ROUGE-1	0.00026698	4.81E-18
DUC-2002	ROUGE-2	0.00012185	3.42E-29

proposed approach to different data sets, including single document, multi-document, multi-lingual, figure summary, and scientific document summarization.

References

1. Radev DR, Joseph MT, Gibson B, Muthukrishnan P (2016) A bibliometric and network analysis of the field of computational linguistics. J Assoc Inform Sci Technol 67(3):683–706
2. Alguliyev RM, Aliguliyev RM, Isazade NR, Abdi A, Idris N (2019) Cosum: text summarization based on clustering and optimization. Expert Syst 36(1):e12340
3. Binwahlan MS, Salim N, Suanmali L (2009) Swarm based text summarization. In: 2009 International association of computer science and information technology-spring conference. IEEE, pp 145–150
4. Ježek K, Steinberger J (2008) Automatic text summarization (the state of the art 2007 and new challenges). In: Proceedings of Znalosti. Citeseer, pp 1–12
5. Alguliev RM, Aliguliyev RM (2005) Effective summarization method of text documents. In: The 2005 IEEE/WIC/ACM international conference on web intelligence (WI'05). IEEE, pp 264–271
6. Jones KS et al (1999) Automatic summarizing: factors and directions. Advances in automatic text summarization, 1–12
7. Nenkova A, McKeown K (2012) A survey of text summarization techniques. In: Mining text data. Springer, pp 43–76
8. Akter S, Asa AS, Uddin MP, Hossain MD, Roy SK, Afjal MI (2017) An extractive text summarization technique for bengali document (s) using k-means clustering algorithm. In: 2017 IEEE International conference on imaging, vision & pattern recognition (icIVPR). IEEE, pp 1–6
9. Alguliev RM, Aliguliyev RM, Isazade NR (2013) Multiple documents summarization based on evolutionary optimization algorithm. Expert Syst Appl 40(5):1675–1689
10. Rautray R, Balabantaray RC (2017) Cat swarm optimization based evolutionary framework for multi document summarization. Physica A: Stat Mech Applic 477:174–186
11. Alguliev RM, Aliguliyev RM, Hajirahimova MS, Mehdiyev CA (2011) Mcmr: maximum coverage and minimum redundant text summarization model. Expert Syst Appl 38(12):14514–14522
12. Debnath D, Das R, Pakray P (2021) Extractive single document summarization using multi-objective modified cat swarm optimization approach: Esds-mcso. Neural Comput Applic, 1–16
13. Radev DR, Hovy E, McKeown K (2002) Introduction to the special issue on summarization. Comput Ling 28(4):399–408
14. Kogilavani A, Balasubramanie P (2010) Clustering based optimal summary generation using genetic algorithm. In: 2010 International conference on communication and computational intelligence (INCOCCI). IEEE, pp 324–329
15. Zhao P, Han J (2010) On graph query optimization in large networks. Proc VLDB Endowm 3(1-2):340–351
16. Wan X, Yang J, Xiao J (2007) Manifold-ranking based topic-focused multi-document summarization. In: IJCAI, vol 7, pp 2903–2908
17. Saini N, Saha S, Jangra A, Bhattacharyya P (2019) Extractive single document summarization using multi-objective optimization: exploring self-organized differential evolution, grey wolf optimizer and water cycle algorithm. Knowl-Based Syst 164:45–67
18. Nayyar A, Le D-N, Nguyen NG (2018) Advances in swarm intelligence for optimizing problems in computer science. CRC Press
19. Nayyar A, Nguyen NG (2018) Introduction to swarm intelligence. Advances in Swarm Intelligence for Optimizing Problems in Computer Science, 53–78
20. Asgari H, Masoumi B, Sheijani OS (2014) Automatic text summarization based on multi-agent particle swarm optimization. In: 2014 Iranian conference on intelligent systems (ICIS). IEEE, pp 1–5
21. Abbasi-ghalehtaki R, Khotanlou H, Esmaeilpour M (2016) Fuzzy evolutionary cellular learning automata model for text summarization. Swarm Evol Comput 30:11–26
22. Fattah MA, Ren F (2008) Automatic text summarization. World Acad Sci Eng Technol 37(2):192
23. Shareghi E, Hassanabadi LS (2008) Text summarization with harmony search algorithm-based sentence extraction. In: Proceedings of the 5th international conference on soft computing as transdisciplinary science and technology, pp 226–231
24. Saleh HH, Kadhim NJ (2016) Extractive multi-document text summarization using multi-objective evolutionary algorithm based model. Iraqi J Sci 57(1C):728–741
25. Zamuda A, Lloret E (2020) Optimizing data-driven models for summarization as parallel tasks. J Comput Sci 42:101101
26. Chu S-C, Tsai P-W, Pan J-S (2006) Cat swarm optimization. In: Pacific Rim international conference on artificial intelligence. Springer, pp 854–858
27. Sharafi Y, Khanesar MA, Mohammad T (2013) Discrete binary cat swarm optimization algorithm. In: 2013 3rd IEEE international conference on computer, control and communication (IC4). IEEE, pp 1–6
28. Ahmed AM, Rashid TA, Saeed SAM (2020) Cat swarm optimization algorithm: a survey and performance evaluation. Computational intelligence and neuroscience
29. Panda G, Pradhan PM, Majhi B (2011) Iir system identification using cat swarm optimization. Expert Syst Appl 38(10):12671–12683
30. Skoullis VI, Tassopoulos IX, Beligiannis GN (2017) Solving the high school timetabling problem using a hybrid cat swarm optimization based algorithm. Appl Soft Comput 52:277–289
31. Saha SK, Ghoshal SP, Kar R, Mandal D (2013) Cat swarm optimization algorithm for optimal linear phase fir filter design. ISA Trans 52(6):781–794
32. Mendoza M, Cobos C, León E (2015) Extractive single-document summarization based on global-best harmony search and a greedy local optimizer. In: Mexican international conference on artificial intelligence. Springer, pp 52–66
33. He Y-X, Liu D-X, Ji D-H, Yang H, Teng C (2006) Msbga: a multi-document summarization system based on genetic algorithm. In: 2006 International conference on machine learning and cybernetics. IEEE, pp 2659–2664
34. Mendoza M, Bonilla S, Noguera C, Cobos C, León E (2014) Extractive single-document summarization based on genetic operators and guided local search. Expert Syst Appl 41(9):4158–4169
35. Vázquez E, Arnulfo Garcia-Hernandez R, Ledeneva Y (2018) Sentence features relevance for extractive text summarization using genetic algorithms. J Intell Fuzz Syst 35(1):353–365
36. ARM (2009) A new sentence similarity measure and sentence based extractive technique for automatic text summarization. Expert Syst Appl 36(4):7764–7772
37. Saini N, Saha S, Chakraborty D, Bhattacharyya P (2019) Extractive single document summarization using binary differential evolution: optimization of different sentence quality measures. PloS one 14(11):e0223477

38. Hernández-Castañeda Á, García-Hernández RA, Ledeneva Y, Millán-Hernández CE (2020) Extractive automatic text summarization based on lexical-semantic keywords. IEEE Access 8:49896–49907
39. Sanchez-Gomez JM, Vega-Rodríguez MA, Pérez CJ (2020) A decomposition-based multi-objective optimization approach for extractive multi-document text summarization. Appl Soft Comput 91:106231
40. Debnath D, Das R, Pakray P (2020) Extractive single document summarization using an archive-based micro genetic-2. In: 2020 7th International conference on soft computing & machine intelligence (ISCMI). IEEE, pp 244–248
41. Uçkan T, Karcı A (2020) Extractive multi-document text summarization based on graph independent sets. Egypt Inform J 21(3):145–157
42. Sanchez-Gomez JM, Vega-Rodríguez MA, Pérez CJ (2021) The impact of term-weighting schemes and similarity measures on extractive multi-document text summarization. Exp Syst Applic 169:114510
43. Bao G, Zhang Y (2021) Contextualized rewriting for text summarization. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol 35, no 14, pp 12544–12553
44. Chettah K, Draa A (2021) A discrete differential evolution algorithm for extractive text summarization. In: 2021 International conference on innovations in intelligent systems and applications (INISTA). IEEE, pp 1–6
45. Thi TTTNB, Dinh TT, Hoai NT et al (2021) A hybrid pso-ga for extractive text summarization. In: Proceedings of the 35th Pacific Asia conference on language, information and computation, pp 742–751
46. El-Kassas WS, Salama CR, Rafea AA, Mohamed HK (2021) Automatic text summarization: a comprehensive survey. Exp Syst Applic 165:113679
47. Wilson PK, Jeba J (2022) A developed framework for multi-document summarization using softmax regression and spider monkey optimization methods. Soft Comput, 1–16
48. Mojrian M, Mirroshandel SA (2021) A novel extractive multi-document text summarization system using quantum-inspired genetic algorithm: Mtsqiga. Exp Syst Applic 171:114555
49. Ruan Q, Ostendorff M, Rehm G Histruct+: Improving extractive text summarization with hierarchical structure information. arXiv:2203.09629
50. Roul RK (2021) Topic modeling combined with classification technique for extractive multi-document text summarization. Soft Comput 25(2):1113–1127
51. Belwal RC, Rai S, Gupta A (2021) A new graph-based extractive text summarization using keywords or topic modeling. J Ambient Intell Humaniz Comput 12(10):8975–8990
52. Xie T, Zhen Y, Li T, Li C, Ge Y (2021) Self-supervised extractive text summarization for biomedical literatures. In: 2021 IEEE 9th international conference on healthcare informatics (ICHI). IEEE, pp 503–504
53. Song W, Choi LC, Park SC, Ding XF (2011) Fuzzy evolutionary optimization modeling and its applications to unsupervised categorization and extractive summarization. Expert Syst Appl 38(8):9112–9121
54. Nguyen HV, Bai L (2010) Cosine similarity metric learning for face verification. In: Asian conference on computer vision. Springer, pp 709–720
55. Kusner M, Sun Y, Kolkin N, Weinberger K (2015) From word embeddings to document distances. In: International conference on machine learning, PMLR, pp 957–966
56. Goldberg Y, Levy O word2vec explained: deriving mikolov et al.'s negative-sampling word-embedding method, arXiv:1402.3722
57. Srinivas M, Patnaik LM (1994) Adaptive probabilities of crossover and mutation in genetic algorithms. IEEE Trans Syst Man Cybern 24(4):656–667
58. Khanesar MA, Teshnehlab M, Shoorehdeli MA (2007) A novel binary particle swarm optimization. In: 2007 Mediterranean conference on control & automation. IEEE, pp 1–6
59. Houck CR, Joines J, Kay MG (1995) A genetic algorithm for function optimization: a matlab implementation. Ncsu-ie tr 95(09):1–10
60. Zhenan H, G YG (2015) Visualization and performance metric in many-objective optimization. IEEE Trans Evol Comput 20(3):386–402
61. Ruxton GD (2006) The unequal variance t-test is an underused alternative to student's t-test and the mann–whitney u test. Behav Ecol 17(4):688–690
62. Ruxton GD, Beauchamp G (2008) Time for some a priori thinking about post hoc testing. Behav Ecol 19(3):690–693

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.