Metaheuristic Aided Improved LSTM for Multi-document Summarization: A Hybrid Optimization Model

Sunilkumar Ketineni and Sheela J*

Department of School of Computer Science and Engineering, VIT-AP University, Amaravathi, Andhra Pradesh, India-522237
E-mail: sheela.j@vitap.ac.in
**Corresponding Author*

Received 01 February 2023; Accepted 27 April 2023;
Publication 24 October 2023

Abstract

Multi-document summarization (MDS) is an automated process designed to extract information from various texts that have been written regarding the same subject. Here, we present a generic, extractive, MDS approach that employs steps like preprocessing, feature extraction, score generation, and summarization. The input text goes preprocessing steps such as lemmatization, stemming, and tokenization in the first stage. After preprocessing, features are extracted, including improved semantic similarity-based features, term frequency-inverse document frequency (TF-IDF-based features), and thematic-based features. Finally, an improved LSTM model will be proposed to summarize the document based on the scores considered under the objectives such as content coverage and redundancy reduction. The Blue Monkey Integrated Coot Optimization (BMICO) algorithm is proposed in this paper

Journal of Web Engineering, Vol. 22_4, 701–730.
doi: 10.13052/jwe1540-9589.2246

for fine-tuning the optimal weight of the LSTM model that ensures precise summarization. Finally, the suggested BMICO's effectiveness is evaluated, and the outcome is successfully verified.

Keywords: Multi-document summarization, LSTM, score generation, BMICO, optimization.

Nomenclature

Abbreviation	Description
ATS	Automatic text summarizationa
BM	Blue Monkey
BMO	Blue Monkey optimization
BMICO	Blue Monkey integrated Coot optimization
DUC	Document Understanding Conferences
FbTS	Firefly-based text summarization
GA	Genetic algorithm
IR	Information retrieval
KNN	K-nearest neighbor
LSTM	Long short-term memory
MDS	Multi-document summarization
ML	Machine learning
MOABC/D	Multi-objective artificial bee colony algorithm based on decomposition
MDS	Multi-document summarization
MTSQIGA	Multi-document text summarization system using a quantum-inspired genetic algorithm
NLP	Natural language processing
NLP	Natural language processing
PSO	Particle swarm optimization
QIGA	Quantum-inspired genetic algorithm
ROUGE	Recall-oriented understudy for gisting evaluation
SVM	Support vector machine
TF-IDF	Term frequency-inverse document frequency

1 Introduction

Nowadays, the amount of knowledge available online is vast and continues to expand daily. This implies that there are more digital documents than ever before. The amount of data makes it challenging to find the most crucial details about a given subject, even though Internet users find these details quickly [1]. The difficulty of gathering valuable information has grown due to the abundance of information available on the Internet, including news stories published on websites. Online forums as well as social networks have also taken over as the most often used media for people to discuss their experiences. Therefore, having an automated summary method is crucial so that one can rapidly select the most crucial and prominent information. Automatic summarize techniques have been used in search engines, websites, media, and other sorts of online evaluations [2, 3]. To summarize documents and news for deeper analysis, a summarize task is used by educational institutions, media outlets, the military, and political entities [4].

Text summarization is the method of taking a text's key concepts and distilling them into manageable pieces [5, 6]. ATS, which condenses texts while retaining their key elements, can aid in the efficient processing of this constantly expanding text collection. Automatic text summarization's major goal is to identify a subset of information that includes all of the data from the larger set [7]. It can be categorized according to several criteria, such as input, goal, communication, and output [8, 9]. Any automatic abstractive summarization must pay close attention to the relevancy and redundancy of the article while summarizing it [10, 11]. Based on specific criteria, automated text summarization systems are regularly divided into many classes. A system can simultaneously fit into multiple categories. Text summarization can be divided into single-document and multi-document categories based on the number of relevant papers [12–14]. Also, based on their intended use, text summarization algorithms can be divided into two groups: general and query-oriented [15–17].

Depending on the context, many English words can be used in a variety of ways. Pronouns and other co-reference elements in the sentences are some other language-related problems that are frequently hard to identify. The text seems to be an unstructured data type, which means that it may be written in a variety of ways depending on the language of origin. The majority of current research in this area is focused on creating a text summarizing system that might simultaneously handle each of these issues [18, 19]. In this way, this

paper proposes a multi-document summarization model with the following contributions:

- Improved LSTM is introduced for summarization, based on the scores generated under the objectives like content coverage and redundancy reduction.
- The Blue Monkey integrated Coot optimization (BMICO) algorithm is proposed to tune the optimal weights of the LSTM.

The organization of this paper is as follows: the introduction is given in Section 1, and the literature review is covered in Section 2. An MDS architectural explanation is offered in Section 3. Section 4 discusses feature extraction and preprocessing. Section 5 explains score generation and summarization. A suggested BMICO algorithm is revealed in Section 6. Section 7 concludes with a statement.

2 Review on Literature Works

2.1 Literature Review

In 2020, Sanchez-Gomez et al. [1] suggested a multi-objective artificial bee colony algorithm based on decomposition (MOABC/D) as a solution to the integrative multi-document text summarization issue. To make use of multi-core systems, the MOABC/D method had an asynchronous similar design built. Document understanding conference (DUC) datasets were used for the experiments, and ROUGE measures were used to assess the outcomes. The acquired results enhanced the ROUGE-1, ROUGE-2, and ROUGE-L scores reported in the scholarly literature while also indicating a very good speedup.

In 2021, Mojrian et al. [15] proposed a multi-document text summarization system using the quantum-inspired genetic algorithm (MTSQIGA) method, a revolutionary technique to multi-document text summarization that draws out key lines from a variety of source documents to produce the summary. To find the optimum solution, the recommended generic summarizer employs a modified quantum-inspired genetic algorithm (QIGA) to offer extractive summarization as a binary optimization. This approach's objective function was crucial in maximizing the six phrase scoring measures that were composed of a concatenation of criteria for coverage, relevancy, and repetition. The recommended QIGA employs a self-adaptive quantum rotation gate along with a tailored quantum measurement, based on the grade and size of the summary, to ensure the development of a summary within

a defined length limit. On benchmark datasets from DUC 2005 and 2007, the suggested scheme was assessed using ROUGE standard metrics. It also shows the potential effectiveness of our suggested technique when used to do text summarization tasks using a genetic algorithm influenced by quantum mechanics.

In 2020, Bidoki et al. [20] offered a semantic evaluation framework for extractive multi-document summarizer models using a concatenation of statistical, machine learning (ML)-based, and graph-based frameworks. This was an unsupervised system with no linguistic restrictions. To derive the semantic form of a word from a collection of provided documents, the suggested framework uses the word2vec approach. Each phrase is extended uniquely by using the most intriguing and least repetitive phrases related to the text's core idea. The task of word meaning disambiguation was implicitly carried out by sentence expansion, which also adjusts the logical densities to the primary subject of each phrase. A creative grouping strategy was put forth to pinpoint the documents' most crucial subjects. It groups texts in accordance with the number of groups and the beginning centroids that it selects on its own.

In 2020, Alqaisi et al. [21] suggested a system that uses evolutionary and clustering-based multi-objective optimization techniques. The key themes in the text were found using the clustering-based technique, and three goals were optimized using the evolving multi-objective optimization technique focused on coverage, diversity/redundancy, and relevance. The suggested system's performance was assessed using the TAC 2011 and DUC 2002 datasets. The ROUGE assessment measure was used to compare the experimental outcome.

In 2021, Lamsiyah et al. [8] offered a method to describe sentences in texts and user queries utilizing embedding vectors that reflect the syntactic and semantic links between its constituents by leveraging transfer learning via pre-trained sentence embedding models (words, phrases). Additionally, a selection of phrases was retrieved based on relevance to the question by linearly combining BM25 and the semantic similarity function. The chosen sentences were then re-ranked using the maximal marginal relevancy criteria, which maintains query relevance and reduces redundancy. The suggested approach was unsupervised, easy to use, effective, and didn't need any labelled training data for text summarization.

In 2020, Wei Li et al. [22] offered a new abstractive multi-document summary paradigm by first transforming documents into something like a semantic link network of concepts and actions, and then selecting essential

ideas and activities while keeping semantic coherence, summarizing the semantic link network. The suggested methodology greatly outperforms appropriate state baselines, according to experiments on benchmark datasets, and the semantic link network is crucial for describing and comprehending documents.

In 2021, Rajendra Kumar Roul et al. [5] offered a fresh approach employing topic modeling and classification technology to synthesize a corpus of documents into a coherent summary. The stochastic aspect of latent Dirichlet assignment is handled via a novel suggested method that was developed to determine the precise number of themes that are present in a corpus of documents. The results indicate that, in comparison to existing text summarization methods, the suggested technique was more effective.

In 2021, Minakshi Tomer et al. [11] suggested a swarm intelligence-based system inspired by nature, viz. multi-document text summarization using the firefly approach. A topic relation factor, cohesiveness factor, and readability factor were all utilized in a new fitness feature. The effectiveness of the suggested method was contrasted with various existing algorithms drawn from nature, such as particle swarm optimization (PSO) and the genetic algorithm (GA). The suggested approach outperforms the others that have been tested. Table 1 shows the current MDS scheme's features and difficulties.

2.2 Review

Numerous sentences in the resulting summary still include syntax problems, despite the SLN incorporating the greedy selection and sentences-over-generation features in our approach. The majority of cases are caused by inaccurate event extraction and event relation extraction, so more efficient methods must be created to enhance the efficiency of summarization. The following list includes some of the main issues with the current framework. In the MOABC/D method, the search procedure proceeds very slowly in comparison. QIGA [15] is a low-cost method, but it adds more statistical features together. In KNN [20], short texts pose a significant problem and may have a negative impact on the system's overall performance. In the k-medoid clustering method [21], Arabic is a language that makes it harder to identify proper nouns, titles, and abbreviations. The greedy search method [8] needs to investigate transfer learning capabilities for summary creation from trained models. Semantic link network summarization [22] is good at conveying and comprehending document semantics, but here improved performance is needed. In the SVM method [5], it is quite difficult to maintain high

Table 1 Analysis of existing techniques for multi-document summarization

Author [Citations]	Methods	Features	Limitations
Jesus M. Sanchez-Gomez et al. [1]	MOABC/D	Excellent rates for efficiency and speed	As far as the search process goes, it moves pretty slowly
Mohammad Mojrian et al. [15]	QIGA	Low cost	Add more statistical measures together
Mohammad Bidoki et al. [20]	KNN	Recall, F-measures scores, and precision were high	Short texts pose a significant problem and may have a negative impact on the system's overall performance
Rana Alqaisi et al. [21]	k-medoid clustering method	The F-measure yields the greatest value	Arabic is a language that makes it harder to identify proper nouns, titles, and abbreviations
Salima Lamsiyah et al. [8]	Greedy search method	Outcomes for all performance metrics (R-1, R-2, and R-SU4) have improved	Need to investigate transfer learning capabilities for summary creation from trained models
Wei Li, Hai Zhuge et al. [22]	Semantic link network summarization	Good at conveying and comprehending document semantics	Need for improved performance
Rajendra Kumar Roul et al. [5]	SVM	Improve performance and accuracy	It is quite difficult to maintain high performance when summarizing a big corpus of documents
Minakshi Tomer et al. [11]	FbTS	F-score and precision obtained high values	It covers problems like having a lot of redundant documents

performance when summarizing a big corpus of documents. FbTS [11], covers problems like having a lot of redundant documents.

3 Proposed Multi-document Summarization: Architectural Layout

This paper presents a new improved LSTM-based multi-document summarization model that creates the summary by retaining the characteristics of

the document with respect to specific text-based features. Thereby, the model includes the following steps:

- Preprocessing
- Feature extraction
- Score generation and summarization.

Also, the proposed technique tries to select the key phrases that address the main concepts of the original text by minimizing the redundant data in the output summary. Figure 1 shows the proposed MDS method.

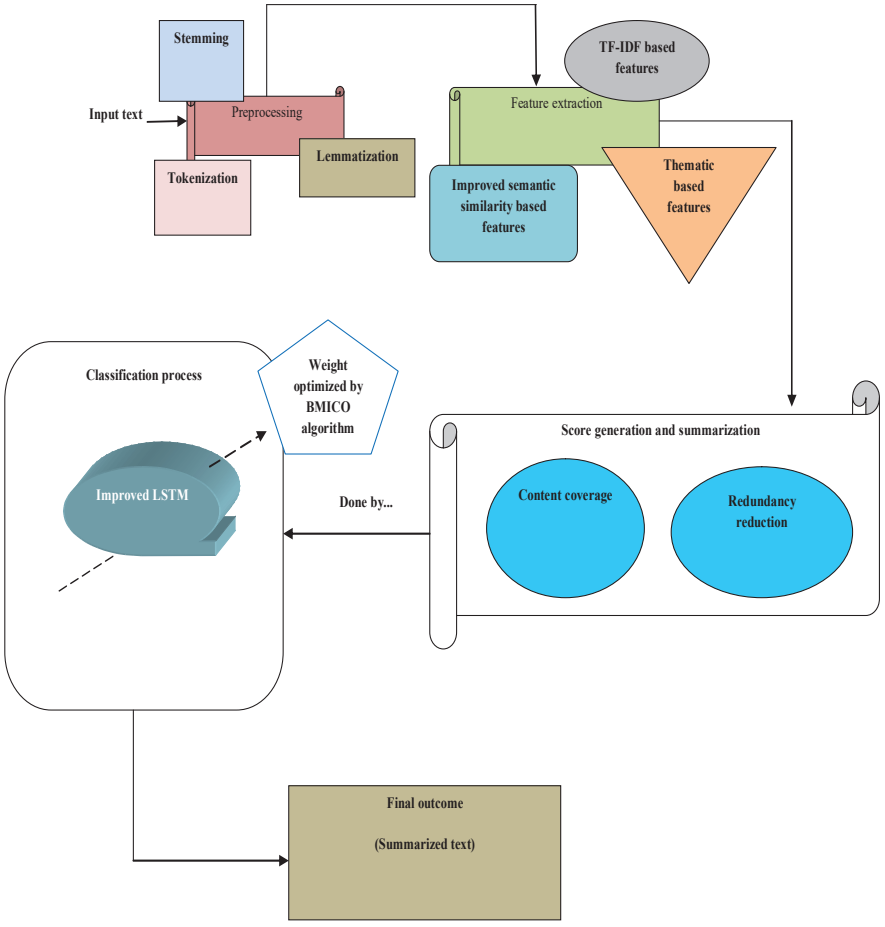

Figure 1 Multi-document summarization architectural layout.

Preprocessing: This is the first step, where the input document will do the processing of lemmatization, stemming, and tokenization.

Feature extraction: After preprocessing, features including improved semantic similarity-based features, TF-IDF-based features, and thematic-based features will be extracted.

Score generation and summarization: Metaheuristic-aided improved LSTM will be used for the summarization of the document based on the scores evaluated under the consideration of constraints like content coverage and redundancy reduction (objective function). This will be evaluated during the weight optimization of the ILSTM model. For this, a new BMICO is introduced in this work. The workflow of the suggested strategy will be more thoroughly explained in the upcoming sections.

4 Description of Preprocessing and Feature Extraction

4.1 Preprocessing

To represent the words and sentences more accurately, preprocessing tries to make words less ambiguous and inconsistent. In this step, the input document I^T will go through preprocessing steps like lemmatization, stemming, and tokenization in this initial step. The preprocessed document is referred to as P^T.

Lemmatization: The computational process of identifying a word's lemma depending on its actual intent is called lemmatization [23]. Finding the "lexical headword" or core word form of a given word is the aim of lemmatization. For the many tasks of NLP, such as keyword identification or IR, it has been discovered that some lemmatization preparation is particularly important for highly inflected languages. Lemmatization reduces the number of words that must be processed.

Stemming: Using stemming [24], words having to match starting letters would be retrieved for each word after it had been searched through a dictionary. At each stage, the words that are the closest in meaning are first chosen from the list of terms, and then they are categorized. When a word is designated as a stop word, it is instantly dropped. The singular or fundamental word formation or the masculinity of the word would be taken into account while counting the frequency of the terms unless it is plural or derived from the verb.

Tokenization [25]: Tokenization seeks to divide the document into manageable pieces, such as sentences, paragraphs, and words. The examination of morphological patterns is closely related to this activity. As a result, it is a challenging task. This tokenization strategy is helpful when there are punctuation mistakes.

4.2 Feature Extraction

After the preprocessing phase, features including improved semantic similarity-based features, TF-IDF-based features, and thematic-based features are extracted from P^T, and the extracted features are indicated as E^T.

Improved semantic similarity: A similarity among textual units (such as sentences) can be determined using a variety of measurements, including the Euclidean distance, cosine similarity, and Jaccard correlation. However, the most widely used measure is cosine similarity. A measure of similarity between two non-zero vectors within an inner product space that calculates the cosines of their angles is known as cosine similarity [26]. We may use it to compare sentences because we will be expressing our sentences as a collection of vectors. Cosine similarity is one of the most popular text similarity metrics, which adds a large computational burden to tasks involving document interpretation. The conventional cosine similarity formula is depicted in Equation (1). As per the proposed approach, improved cosine similarity is depicted in Equation (2), where the weighted mean is multiplied which ensures additive weightage in calculating the similarity among the vectors.

$$\cos(e, f) = \frac{e \cdot f}{||e|| \cdot ||f||} \tag{1}$$

$$\cos(e, f) = \frac{\left(\sum_{i=1}^{n} e_i \cdot f_i\right)^2}{\sqrt{\sum_{i=1}^{n} e_i^2 \sqrt{\sum_{i=1}^{n} f_i^2}}} \times Mean_w \tag{2}$$

where, $e, f =$ vectors, $e_i^2, f_i^2 =$ vector length, $Mean_w = \frac{\sum_{i=1}^{n} w_i x_i}{\sum_{i=1}^{n} w_i}$, weighted mean, $n =$ term count.

TF-IDF-based features: It is possible to quantify the significance or relevancy of string forms (words, phrases, lemmas, etc.) in a document among a group of documents using the TF-IDF [27] metric, which is utilized in the disciplines of IR and ML. The score of any phrase in any document is

determined by multiplying the TF and IDF for the particular words provided in Equation (3).

$$TF - IDF[word, I^T] = \{ TF[word, I^T] * IDF[word] \} \tag{3}$$

Thematic-based features: The creation of topic signatures or important keyword identification and weighting are required for thematic features. A user's information demand is described in one or even more paragraphs in the narrative. This makes it possible for us to calculate the accompanying sentence features, in which each feature evaluates how closely the subject distribution F of the phrase resembles the topic distribution of the "query":

$g(F, N) =$ cluster narrative, $g(F, CT) =$ cluster title, $g(F, T) =$ document title, $g(F, C) =$ cluster centroid vector, $g(F, D) =$ document term vector, and $E^T = [\cos \ \ TF - IDF \ g(F, D)]$.

5 Improved LSTM-based Summarization with Optimal Tuning of Weights

The input given to the improved LSTM is E^T and the output will be the generated summary. To enhance the performance of summary generation, the weights of them are optimally tuned by the new BMICO algorithm based on the objectives of content coverage and redundancy reduction.

5.1 Objective Function (Score) and Input Solution to the BMICO Algorithm

Score Level 1: By using the most pertinent sentence, the created summary should encompass most of the document collection's content. First, the content coverage criterion is addressed by the objective function $\tau_{\text{cov}}(E)$. A cosine similarity among sentences l_i and the group of sentences in Q indicated by the mean vector H is used to determine the content coverage for the sentence $l_i \in L$. The objective function in Equation (4) should be maximized:

$$\tau_{\text{cov}}(E) = \sum_{i=1}^{n} sim(l_i, H) \cdot e_i \tag{4}$$

Score level 2: The redundancy reduction criteria are addressed by the objective function $\tau_{\text{Re\_}R}(E)$. It is necessary to define a unique binary decision variable f_{ij}. The sentences l_i and l_j have a connection to this variable.

A cosine similarity among each pair of phrases $l_i, l_j \in L$, $sim(l_i, l_j)$, should be minimized in Equation (5). The same thing happens when the objective function listed below is maximized:

$$\tau_{\text{Re\_}R}(E) = \frac{1}{\left(\sum_{i=1}^{n-1} \sum_{j=i+1}^{n} sim(l_i, l_j).f_{ij}\right) \cdot \sum_{i=1}^{n} e_i} \tag{5}$$

The final score as the multi-objective function is defined as per Equation (6).

$$ob = \min\left(\frac{1}{\{\tau_{\text{cov}}(E), \tau_{\text{Re\_}R}(XE)\}}\right) \tag{6}$$

5.2 Improved LSTM Classifier for Summarization

LSTM [28] is frequently utilized as a sequence-to-sequence based approach to construct text summarization models. Figure 2 represents the LSTM model.

Encoder: The encoder has two stages: a forward encoder and a backward encoder. When text is transformed into a vector, a forward encoder reads it from the front. The backward encoder, on the other hand, reads a sequence vector from the back. They are written as in Equation (7):

$$I_t = LSTM(e_t, I_{t-1}) \tag{7}$$

where, I_t = hidden state for encoder at time step t, e_t = input. LSTM formulas are depicted in Equations (8)–(13).

$$u_t = \omega(P_u e_t + Q_u I_{t-1} + Z_u) \tag{8}$$

$$x_t = \omega(P_x e_t + Q_x I_{t-1} + Z_i) \tag{9}$$

$$\hat{s}_t = \tanh(P_s e_t + Q_s I_{t-1} + Z_s) \tag{10}$$

$$s_t = u_t \times s_{t-1} + x_t \times \hat{s}_t \tag{11}$$

$$T_t = \omega(P_T e_t + Q_T I_{t-1} + Z_T) \tag{12}$$

$$I_t = T_t \times \tanh(s_t) \tag{13}$$

In the above equations, u_t is the forget gate, x_t is the input gate, T_t is the output gate, s_t is the newly upgraded vector or context vector, $\hat{s}_t$ is the candidate vector from new context, and $P \in R^{I \times d}, Q \in R^{I \times I}, Z \in R^I$ are the trainable parameters for weight matrix. Here, $\overline{W} = \{P, Q, Z\}$ and

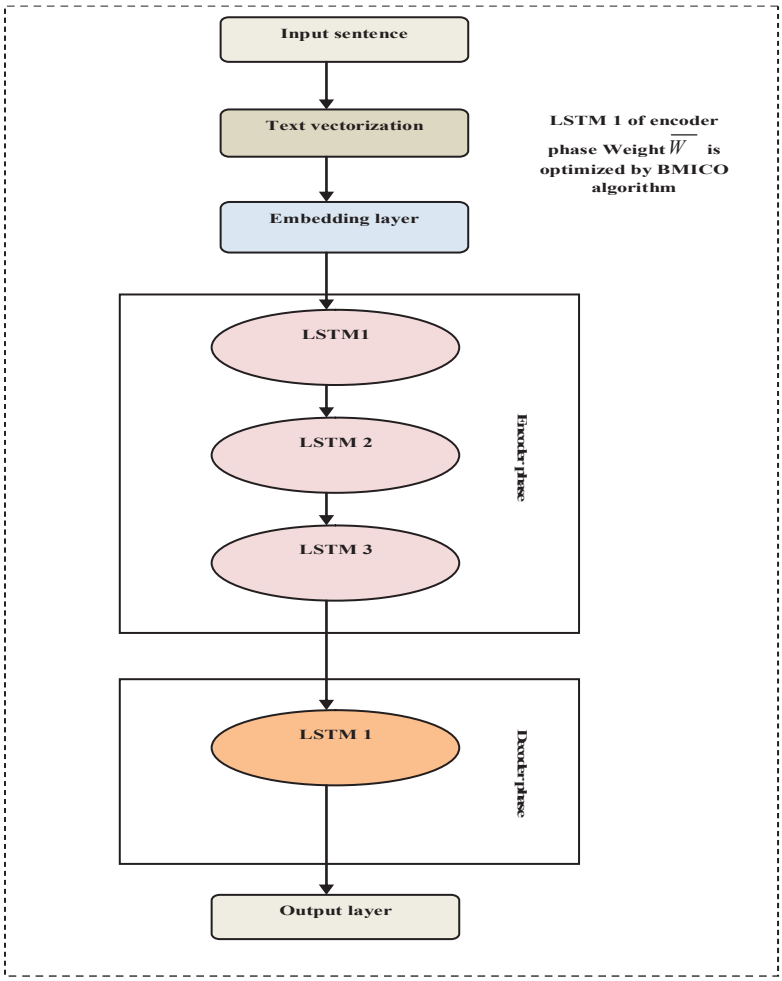

Figure 2 Improved LSTM classifier.

the weight matrix was optimally tuned by the proposed hybrid optimization algorithm BMICO by fixing the objective (score) given in Equation (6).

Decoder: The decoder procedure reads the input data that have been transformed into vector form and have undergone the encoding process to generate English words. The decoder's expression is the same as that of the encoder with various inputs, which is written as in Equation (14).

$$L_t = LSTM(f_{t-1}, s_{t,y_{t-1}}) \tag{14}$$

where, L_t = hidden state, f_{t-1} = old time step output, s_t = context vector. The conventional binary cross entropy loss function [29] of LSTM is defined in Equation (15).

$$LoS = \frac{1}{output^{Size}} \sum_{i=1}^{output^{Size}} m_i \cdot \log \hat{m}_i + (1 - m_i) \cdot \log(1 - \hat{m}_i) \quad (15)$$

As per the proposed logic, the improved LSTM binary cross entropy loss function is defined in Equation (16)

$$LoS = \frac{1}{output^{Size}} \sum_{i=1}^{output^{Size}} m_i \cdot \log \hat{m}_i + \frac{(1 - m_i) \cdot \log(1 - \hat{m}_i)}{w_i} \quad (16)$$

where, $\hat{m}_i$ = *i*th scalar value in the model outcome, m_i = target value, w_i = weight function, which is calculated by Gaussian map function in Equation (17). The LSTM output will be the generated summaries.

$$V_{c+1} = \begin{cases} 0, & V_c = 0 \\ \left(\frac{1}{V_c}\right) \bmod(1), & V_c \neq 0 \end{cases} \quad (17)$$

6 Proposed Blue Monkey Integrated Coot Optimization (BMICO) Algorithm

BMICO appears to be the obvious solution to this kind of weight optimization issue. The proposed algorithm is a combination of the Coot [30] and BMO [31] algorithms. In this, the conceptual procedure of solution update is processed by the BMO principle. The mathematical model of the BMICO algorithm is as follows. This algorithm begins with the initial population $(\overline{W}) = \{\overline{W}_1, \overline{W}_2, \ldots \overline{W}_N\}$. The target function evaluates this random population numerous times, and a target value is established as $(\vec{G}) = \{G_1, G_2, \ldots, G_n)$. The population is randomly generated in the slight space using the formula (18):

$$Coot^{Pos}(i) = rand(1, e) * (U^B - L^B) + L^B \quad (18)$$

where, e = problem dimension, $Coot^{Pos}(i)$ = Coot position, U^B, L^B = upper as well as lower bound of search space, which is determined by Equation (19)

$$L^B = [L_1^B, \ldots, L_e^B], U^B = [U_1^B, \ldots, U_d^B] \quad (19)$$

The random movement to this side and that side: To carry out this movement, we move the Coot in the direction of a random place determined by the formula (20) in a search area

$$M = rand(1, e) * (U^B - L^B) + L^B \tag{20}$$

The search space is explored by this Coot movement in many areas. If the algorithm gets trapped inside the local optimal, this movement will allow it to escape. Coot's new location is determined as per formula (21).

$$Coot^{Pos}(i) = Coot^{Pos}(i) + T \times J2 \times (M - Coot^{Pos}(i)) \tag{21}$$

where, $J2$ = random number among (0, 1), T is calculated according to Equation (22), where, F = current iteration, I^I = maximum iteration

$$T = 1 - F \times \left(\frac{1}{I^T}\right) \tag{22}$$

Proposed chain movement: If the algorithm is stuck inside the local optima, it can move in a way that allows it to escape. After first determining the distance vector between them, we can also move the Coot towards the opposing Coot by roughly half the distance between them. Formula (23) was applied to determine the Coot's new position:

$$Coot^{Pos}(i) = 0.5 \times (Coot^{Pos}(i-1) + Coot^{Pos}(i)) \tag{23}$$

where, $Coot^{Pos}(i-1)$ is the second Coot. According to the proposed approach, the Coot new position update is done by BMO, where, Z_i^{ch} is the weight of the leader child, $rate_{i+1}^{ch}$ = child power rate, and D is the distance, which is calculated in Equation (25). Here, p_1, p_2 is the first point coordinate, q_1, q_2 is the second point coordinate.

BMO equation: $Z_{i+1}^{ch} = Z_i^{ch} + rate_{i+1}^{ch} * rand.$

Updated equation:

$$Coot^{Pos}(i) = Z_i^{ch} + \frac{0.5 \times (Coot^{Pos}(i-1) + Coot^{Pos}(i))}{rate_{i+1}^{ch} * rand} \times D \tag{24}$$

$$D = \sqrt{(p_2 - p_1) + (q_2 - q_1)} \tag{25}$$

Changing position in accordance with the group leaders: QoS should be achieved. The group is often led by a few Coots in the front, and the remainder

of the Coots must move closer and adjust their position in accordance with the group's leaders. Equation (26) is used for implementing this movement.

$$G = 1 + (iMODLc) \tag{26}$$

where i represents the current Coot index, G represents the leader index, and Lc represents the leader count. $Coot(i)$ wants to update its location depending on the leader's g. Formula (27) uses the chosen leader to determine the Coot's subsequent position.

$$\begin{aligned} Coot^{Pos}(i) = Lead^{Pos}(g) + 2 \times J1 \times \cos(2J\pi) \\ \times (Lead^{Pos}(g) - Coot^{Pos}(i)) \end{aligned} \tag{27}$$

where $Coot^{Pos}(i)$ represents the current Coot position, $Lead^{Pos}(i)$ represents the chosen leader position, and $J1$, J represents the random number which was estimated using the sine map function according to the suggested method. The sine map is a dynamic system and is defined as: $z_{c+1} = \sin(\pi z_c)$.

Leader movement: Leaders must reposition themselves in the context of the objective to guide the group toward the goal (the ideal region). It is advised to update the leader's position using formula (28).

$$Lead^{Pos}(i) = \begin{cases} I \times J3 \times \cos(2J\pi) \times OB - Lead^{Pos}(i) + OB, J4 < 0.5 \\ I \times J3 \times \cos(2J\pi) \times OB - Lead^{Pos}(i) - OB, J4 \geq 0.5 \end{cases} \tag{28}$$

where, $OB =$ best position and $J3$, $J4 =$ random number. Here, $I = 2 - F \times (\frac{1}{I^T})$, F is a current iteration and I^T is a maximum iteration. The pseudo-code of BMICO is presented below:

Algorithm 1 Blue Monkey integrated Coot optimization (BMICO)

Initialize the Coot population randomly
Initialize parameter $P = 0.5$, Lc, $Count_{coot}$ (Coot count)
$Count_{coot} = Count_{pop} - Count1$
Select Coot leader randomly
Calculate Coot and leader fitness
Find the best leader or Coot as OB
While end criterion is not satisfied
  Calculate T, I parameter

If $rand < P$
$J, J1, J3$ are the random numbers along the problem dimension
else
$J, J1, J3$ are random numbers
end if
For $i = 1$ to $Count_{coot}$
calculate parameter of G
If $rand > 0.5$
Update Coot position by Equation (27) with sine map randomization
else
If $rand < 0.5i \sim= 1$
update Coot position by Equation (21)
end if
end if
calculate Coot fitness
if $Coot^{fitness} < lead^{fitness}(g)$
$temp = lead(g)$
$lead(g) = coot$
$coot = temp$
end if
for leader count
update leader position by Equation (30.1)
else
update leader position by Equation (30.2)
end for
if $lead^{fitness} < OB$
$temp = OB$
$OB = lead$
$lead = temp$
end if
end for
$I^T = I^T + 1$
end while

7 Results and Discussion

7.1 Simulation Procedure

Python was used as the implementation tool to carry out the proposed model. The used dataset was given in [32]. The proposed Blue Monkey integrated Coot optimization (BMICO) work was assessed using various performance

metrics, including F-measure, precision, recall, ROUGE, ablation research, and statistical analysis, to demonstrate its efficacy. The suggested BMICO model was contrasted with traditional methods including cat and mouse based optimization (CMBO), bald eagle search (BES), Spider Monkey optimization (SMO), and Blue Monkey optimization (BMO), respectively. By changing the learning percentages to 60, 70, 80, and 90, the efficacy of the methods was evaluated.

7.2 Dataset Description

The data available here for each past DUC workshop include:

1. Documents
2. Summaries, results, etc.
    - manually created summaries
    - automatically created baseline summaries
    - submitted summaries created by the participating groups' systems
    - tables with the evaluation results
    - additional supporting data and software.

7.3 Performance Assessment on Precision

Figure 3 portrays the proposed BMICO model-based multi-document summarization in contrast to previously used techniques. By comparing the suggested model to the other existing approaches, the precision analysis

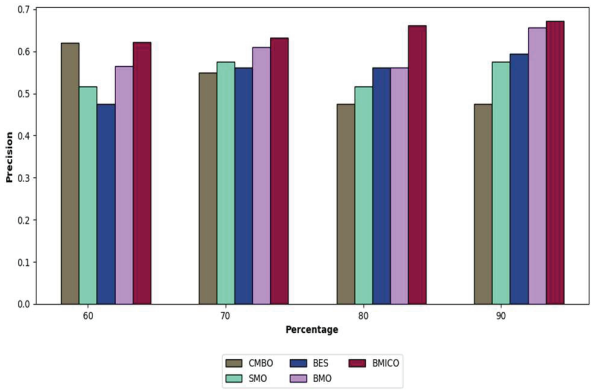

Figure 3 Analysis of the proposed BMICO methodology's performance vs. traditional methods in terms of precision.

revealed that it possessed a high level of precision. This figure indicates that the learning percentage is rising at the same time that the proposed method's precision is improving. Under a 60% learning rate, the implemented technique obtains a precision of 62.18%, which is quite high in contrast to the established techniques such as CMBO (62.01%), SMO (51.66%), BES (47.5%) and BMO (56.44%).

The recommended technique produces the maximum precision (63.15%), in the learning percentage of 70%, compared to the CMBO, SMO, BES, and BMO's precision of 54.86%, 57.52%, 56.15%, and 61.01%, respectively. Simultaneously, the proposed method has obtained improved precision by 66.15% at 80% of the learning percent, outperforming CMBO (47.5%), SMO (51.66%), BES (56.15%), and BMO (56.11%). Finally, the adopted model's precision is 67.12% when accounting for learning rates of 90%, which is significantly higher than the precision of other techniques like CMBO = 47.5%, SMO = 57.44%, BES = 59.44%, and BMO = 65.64%. As a result, the dataset findings have shown that our proposed BMICO method is the more accurate system for multiple document summarization.

7.4 Performance Assessment on F-measure

Figure 4 depicts the performance of the suggested BMICO technique in relation to the F-measure. Though the suggested technique is preferable

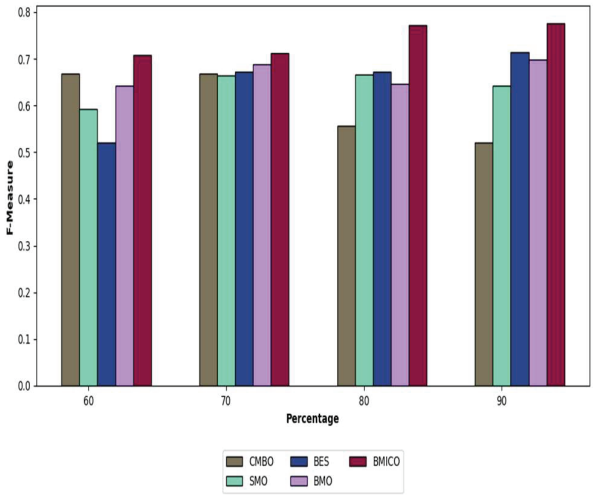

Figure 4 Analysis of the proposed BMICO methodology's performance vs. traditional methods in terms of F-measure.

to the other established approaches, the proposed classifier has the highest F-measure, i.e. approximately above 70% across all learning percentages. While analyzing the proposed work, it achieved the maximal F-measure as 77.50% (in the 90% of the learning percentage), although it is superior to CMBO (52.10%), SMO (64.20%), BES (71.46%), and BMO (69.82%).

Once more, the suggested model demonstrated its suitability for multi-document summarization. The proposed model gained an F-measure of 70.82% after examining 60% of the learning rate, meanwhile, the lowest F-measure is 52.10%, followed by SMO at 59.32% and BMO at 64.15%. The suggested model's F-measure in the 70% learning rate is 66.82%, whereby it is much preferable to the F-measures of the conventional approaches, CMBO = 66.82%, SMO = 66.45%, BES = 67.14%, and BMO = 68.82%. This highlights that the suggested BMICO method document summarization method is a fairly beneficial and effective summary system.

7.5 Performance Assessment on Recall

The recall measure is calculated to give the suggested BMICO technique some extra value. Recall data for the proposed model in comparison to traditional methods are displayed in Figure 5. Additionally, the recall for the suggested model is 84.26% (in 80% of the learning percentage), the CMBO algorithm obtained the lowest recall at 62.85%, followed by SMO at 72.91%

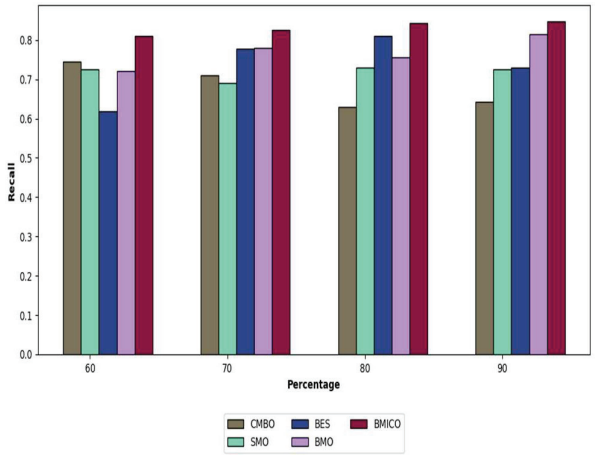

Figure 5 Analysis of the proposed BMICO methodology's performance vs. traditional methods in terms of recall.

and BES at 81.02%. Likewise, the suggested model gained a recall of 84.62% at a 90% learning rate; this is much greater than the recalls of the CMBO (64.26%), SMO (72.5%), BES (73.05%), and BMO (81.42%), respectively.

While the proposed BMICO model achieves a recall of 81.06%, the other extant techniques, such as CMBO, SMO, BES, and BMO, reached recall as 74.42%, 72.5%, 61.87%, and 72.15% in 60% of learning percent. The suggested method's recall rate is 82.55% (at 70% learning rate), which is extremely higher than that of more widely used techniques like CMBO = 71.05%, SMO = 68.98%, BES = 77.85%, and BMO = 78.04%. The general outcomes show that the suggested BMICO methodology has greatly enhanced multi-document summarization performance.

7.6 Performance Assessment on ROUGE

Figure 6 displays the performance of the proposed BMICO method with conventional methods with respect to ROUGE. The empirical framework with high ROUGE is quite effective for summarizing multiple documents. The proposed BMICO work is the sole classifier that has been gained with the greatest ROUGE. The suggested model receives the ROUGE at 77.57% (in the 80% of the learning rate), while the smallest ROUGE is seen in the CMBO algorithm at 67.68%, accompanied by SMO at 69.62% and BMO at 71.50%.

According to the 90% of learning percentage, the proposed BMICO model has the ROUGE as 78.35%, although it is preferable to the extant

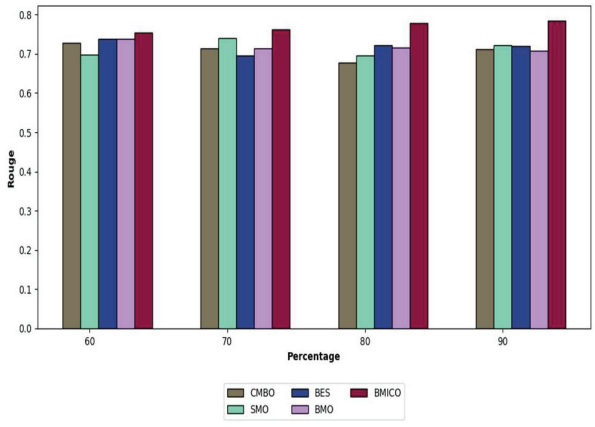

Figure 6 Analysis of the proposed BMICO methodology's performance vs. traditional methods in terms of ROUGE.

models like CMBO (71.16%), SMO (72.28%), BES (72%), and BMO (70.76%). The ROUGE obtained by CMBO, SMO, BES, and BMO is 72.84%, 69.77%, 73.86%, and 73.80% (in 60% of learning percentage), which is relatively low when compared to the suggested approach (75.47%). Finally, the proposed work has a ROUGE of 76.20%, which is substantially greater than CMBO, SMO, BES, and BMO, respectively, in 70% learning percentage. It is worthwhile acknowledging that our BMICO technique provides better results compared to the other four peer systems, as shown by the ROUGE outcomes.

7.7 Ablation Study

Table 2 shows the examination of the ablation study using various metrics of the suggested BMICO model in contrast to traditional methods. The proposed BMICO work intimates its capability for summarization of multi documents. The proposed model obtain a precision of 63.15%, the model without features is 30.55%, the model with conventional LGDIP is 34.87%, and the model without optimization is 46.94%. The F-measure of the model without features, the model with conventional LGDIP, the model without optimization, and the proposed method is 39.80%, 51.92%, 54.69%, and 71.14%, respectively.

The ROUGE of the model without features = 51.78%, the model with conventional LGDIP = 56.37%, the model without optimization = 62.60%, and the proposed BMICO = 76.20%. The recall of the proposed BMICO model is 82.55%, a model without features, 55.40%, a model with conventional LGDIP, 56.37%, and a model without optimization, 62.60%. The suggested BMICO method is built on a hybrid optimization model representation of the document that provides interpretability, controllability, and traceability while accessing documents across a variety of purposes, like document summarization.

Table 2 Ablation study

	Proposed Without Features	Proposed with Conventional LGDIP	Proposed Without Optimization	BMICO
ROUGE	0.517819	0.534282	0.536396	0.762024
Precision	0.305502	0.348745	0.469498	0.631538
Recall	0.554054	0.563707	0.626082	0.825512
F-measure	0.398033	0.519204	0.546952	0.711429

Table 3 Statistical analysis

Method	Standard Deviation	Mean	Maximum	Median	Minimum
CMBO	0.010374	0.60659	0.650145	0.601487	0.600464
SMO	0.005448	0.604066	0.615322	0.601247	0.600743
BES	0.004248	0.60136	0.628567	0.600022	0.600022
BMO	0.015768	0.623225	0.65167	0.612436	0.603198
BMICO	0.00642	0.600171	0.630763	0.597082	0.597012

7.8 Statistical Analysis

Despite having a stochastic nature, the optimization process is exposed to repeated runs in order to ascertain the final results in terms of statistical metrics. In Table 3, the statistical analysis of the proposed BMICO model is compared to the traditional approaches such as CMBO, SMO, BES, and BMO. Standard deviation, maximum, minimum, mean, and median are five separate case scenarios used to analyze it. When compared to the conventional models, such as CMBO = 0.60659, SMO = 0.604066, BES = 0.60136, and BMO = 0.623225, the mean obtained by the suggested work is 0.600171, which is incredibly low. The proposed BMICO technique scored the median value of 0.597082, whilst the BMO approach achieved the maximum median of 0.612436, followed by CMBO at 0.601487 and SMO at 0.601247.

Consequently, the proposed BMICO model's minimum value is 0.597012 which is greater than the values of CMBO (0.600464), SMO (0.600743), BES (0.600022), and BMO (0.603198). Additionally, the suggested BMICO approach has a standard deviation and maximum of 0.00642 and 0.630763, respectively. The analysis shows that the suggested BMICO strategy is suitable for multi-document summarization.

7.9 Analysis of Classifiers

The study of classifiers for the suggested BMICO methodology is summarized in Table 4 along with other widely used techniques including ATSDL [33], DL [34], NN, RNN, CNN, and LSTM. As a result of this investigation, the proposed BMICO model attained higher ROUGE, precision, recall, and F-measure. The proposed BMICO approach outperformed more traditional approaches, such as ATSDL = 45.58%, DL = 55.89%, NN = 35.48%, RNN = 47.98%, CNN = 45.96%, and LSTM = 54.69%, with a highest F-measure of 71.14%. ATSDL and NN perform the worse in terms of precision measure (34.92% and 34.95%, respectively), meanwhile, the proposed methodology performs the best (82.55%).

Table 4 Analysis on classifiers

	ATSDL [33]	DL [34]	NN	RNN	CNN	LSTM	BMICO
ROUGE	0.555251	0.613717	0.455187	0.53556	0.523784	0.536396	0.762024
Precision	0.349202	0.453867	0.349524	0.375663	0.353063	0.469498	0.631538
Recall	0.600463	0.630914	0.49731	0.591358	0.603906	0.626082	0.825512
F-measure	0.4558	0.558992	0.354851	0.479823	0.459661	0.546952	0.711429

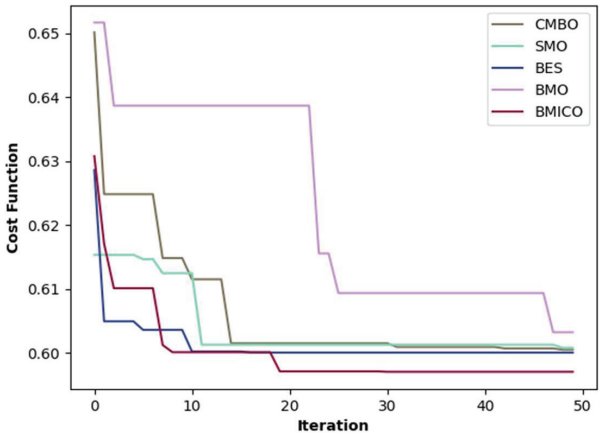

Figure 7 Convergence study of the proposed BMICO work vs. conventional models.

In addition, the adopted model's ROUGE is 76.20%, which is significantly higher than the ROUGE of existing models like ATSDL (55.52%), DL (61.37%), NN (45.51%), RNN (53.55%), CNN (52.37%), and LSTM (53.63%). The proposed model's recall is 82.55%, whilst it is superior to ATSDL, DL, NN, RNN, CNN, and LSTM. By the findings depicted in Table 4, our suggested BMICO method beats all other current approaches for multi-document text summarization, i.e. it performs far superior to other state-of-the-art methods based on F-measure, ROUGE, recall, and precision assessment metrics.

7.10 Convergence Analysis

In Figure 7, the convergence study of the suggested BMICO strategy over the existing approaches is shown by adjusting the iterations 10, 20, 30, 40, and 50, respectively. This figure shows that the proposed BMICO method has a low error rate and converges faster than other existing methods. According to the suggested BMICO model, it reached a greater level of convergence in the first iteration and began to converge in the 10th iteration ($\sim$0.6032). Once

Table 5 Analysis on predicted summary

Input text	Summarized Output
The California earthquake of Oct. 17, 1989 caused extensive damage. Despite heavy insurance losses, insurance company stocks posted gains as investors bet on increases in insurance rates producing a long-term increase in profits. In Washington, the White House appeared hyperactive, anxious to show its responsiveness to the disaster. In Sacramento, Governor Deukmejian called a special session of the legislature to deal with the crisis. Critics accused him of blaming others for the collapse of freeways and called on him to get on with reconstruction and a temporary increase of the gasoline tax so hundreds of thousands of commuters could get to work.	Iraq invades Kuwait the world reacts to northern California earthquake of October

more, convergence occurred at iteration 20, and it remained stable up to the 50th iteration with the same convergence value (0.5908). Additionally, the BES method yields a convergence value of 0.6298 (in the final iteration). The convergence values for the BMO and SMO methods are ∼0.6089 and 0.6054 at iteration 50. The convergence study reveals that the suggested BMICO model converges more quickly than the existing methods, allowing for the reduction of error and the creation of multi-document summaries that are extremely accurate.

7.11 Predicted Summary

Numerous organizations profit from prediction, which has proven difficult and expensive. Table 5 shows the predicted summary analysis. Think of a earthquake predicting its demand for the following year using current data and Excel. Prediction is getting cheaper as it gets simpler across different countries. Consider how creating basic forecasts using spreadsheets become simpler and more affordable with the use of Microsoft Excel.

8 Conclusion

We describe an extractive MDS method that makes use of preprocessing, feature extraction, score generation, and summarization. The input goes through preprocessing steps such as lemmatization, stemming, and tokenization in the

first stage. After preprocessing, features are extracted, including improved semantic similarity-based features, TF-IDF-based features, and thematic-based features. In this step, improved LSTM will be used to obtain the best scores by choosing objectives such as content coverage and redundancy reduction. The Blue Monkey integrated Coot optimization (BMICO) algorithm was proposed in this paper for optimization strategy. Finally, the suggested BMICO's effectiveness was evaluated, and the outcome was successfully verified.

References

[1] Jesus M. Sanchez-Gomez a, Miguel A. Vega-Rodríguez, Carlos J. Pérez, "A decomposition-based multi-objective optimization approach for extractive multi-document text summarization", Applied Soft Computing Journal, vol. 91, 2020.

[2] Taner Uçkan, Ali Karcı, "Extractive multi-document text summarization based on graph independent sets", Egyptian Informatics Journal, vol. 21, 2020.

[3] Tran, NT., Nghiem, MQ., Nguyen, N.T.H. et al. ViMs: a high-quality Vietnamese dataset for abstractive multi-document summarization. Lang Resources and Evaluation 54, 893–920 (2020). https://doi.org/10.1007/s10579-020-09495-4.

[4] Khaleghi, Z., Fakhredanesh, M. and Hourali, M. MSCSO: Extractive Multi-document Summarization Based on a New Criterion of Sentences Overlapping. Iran J Sci Technol Trans Electr Eng 45, 195–205 (2021). https://doi.org/10.1007/s40998-020-00361-1.

[5] Roul, R.K. Topic modeling combined with classification technique for extractive multi-document text summarization. Soft Comput 25, 1113–1127 (2021). https://doi.org/10.1007/s00500-020-05207-w.

[6] Min Yang, Xintong Wang, Yao Lu, Jianming Lv, Ying Shen, Chengming Li, "Plausibility-promoting generative adversarial network for abstractive text summarization with multi-task constraint", Information Sciences, vol. 521, 2020.

[7] Salima Lamsiyah, Abdelkader El Mahdaouy, Bernard Espinasse, Saïd El Alaoui Ouatik, "An unsupervised method for extractive multi-document summarization based on centroid approach and sentence embeddings", Expert Systems with Applications, vol. 167, 2021.

[8] Lamsiyah, S., El Mahdaouy, A., Ouatik El Alaoui, S. et al. Unsupervised query-focused multi-document summarization based on transfer

learning from sentence embedding models, BM25 model, and maximal marginal relevance criterion. J Ambient Intell Human Comput (2021). https://doi.org/10.1007/s12652-021-03165-1.

[9] Alireza Ghadimi, Hamid Beigy, "Deep submodular network: An application to multi-document summarization", Expert Systems With Applications, vol. 152, 2020.

[10] Gao, Y., Meyer, C.M. and Gurevych, I. Preference-based interactive multi-document summarisation. Inf Retrieval J 23, 555–585 (2020). https://doi.org/10.1007/s10791-019-09367-8.

[11] Minakshi Tomer, Manoj Kumar, "Multi-document extractive text summarization based on firefly algorithm", Journal of King Saud University – Computer and Information Sciences, 2021.

[12] Hou Pong Chan, Irwin King, "A condense-then-select strategy for text summarization", Knowledge-Based Systems, vol. 227, 2021.

[13] Ramesh Chandra Belwal, Sawan Rai, Atul Gupta, "Text summarization using topic-based vector space model and semantic measure", Information Processing and Management, vol. 58, 2021.

[14] Srivastava, A.K., Pandey, D. and Agarwal, A. Extractive multi-document text summarization using dolphin swarm optimization approach. Multimed Tools Appl 80, 11273–11290 (2021). https://doi.org/10.1007/s11042-020-10176-1.

[15] Mohammad Mojrian, Seyed Abolghasem Mirroshandel, "A novel extractive multi-document text summarization system using quantum-inspired genetic algorithm: MTSQIGA", Expert Systems With Applications, vol. 121, 2021.

[16] Shirin Akther Khanam, Fei Liu, Yi-Ping Phoebe Chen, "Joint knowledge-powered topic level attention for a convolutional text summarization model", Knowledge-Based Systems, vol. 228, 2021.

[17] Jesus M. Sanchez-Gomez, Miguel A. Vega-Rodríguez, Carlos J. Pérez, "The impact of term-weighting schemes and similarity measures on extractive multi-document text summarization", Expert Systems With Applications, vol. 169, 2021.

[18] Jesus M. Sanchez-Gomez, Miguel A. Vega-Rodríguez, Carlos J. Pérez, "A decomposition-based multi-objective optimization approach for extractive multi-document text summarization", Applied Soft Computing Journal, 2020.

[19] Leonhard Hennig and Berlin, "Topic-based Multi-Document Summarization with Probabilistic Latent Semantic Analysis", International Conference RANLP 2009 – Borovets, Bulgaria.

[20] Mohammad Bidoki, Mohammad R. Moosavi, Mostafa Fakhrahmad, "A semantic approach to extractive multi-document summarization: Applying sentence expansion for tuning of conceptual densities", Information Processing and Management, vol. 57, 2020.

[21] R. Alqaisi, W. Ghanem and A. Qaroush, "Extractive Multi-Document Arabic Text Summarization Using Evolutionary Multi-Objective Optimization With K-Medoid Clustering, "in IEEE Access, vol. 8, pp. 228206–228224, 2020, DOI: 10.1109/ACCESS.2020.3046494.

[22] W. Li and H. Zhuge, "Abstractive Multi-Document Summarization Based on Semantic Link Network, "in IEEE Transactions on Knowledge and Data Engineering, vol. 33, no. 1, pp. 43–54, 1 Jan. 2021, DOI: 10.1109/TKDE.2019.2922957.

[23] Lucie Skorkovska, "Application of Lemmatization and Summarization Methods in Topic Identification Module for Large Scale Language Modeling Data Filtering", DOI: 10.1007/978-3-642-32790-2_23, 2012.

[24] Marzieh Berenjkoub, Razieh Mehri, Hadi Khosravi Farsani, Mohammad Ali Nematbakhsh, "A method for stemming and eliminating common words for Persian text summarization", DOI: 10.1109/NLPKE.2009.5313836, 2009.

[25] R. Alqaisi, W. Ghanem and A. Qaroush, "Extractive Multi-Document Arabic Text Summarization Using Evolutionary Multi-Objective Optimization With K-Medoid Clustering, "in IEEE Access, vol. 8, pp. 228206–228224, 2020, DOI: 10.1109/ACCESS.2020.3046494.

[26] Sahar Sohangir and Dingding Wang, Improved sqrt-cosine similarity measurement, Sohangir and Wang J Big Data (2017) 4:25, DOI: 10.1186/s40537-017-0083-6, 2017.

[27] Bijoyan Das and Sarit Chakraborty, "An Improved Text Sentiment Classification Model Using TF-IDF and Next Word Negation", 2022.

[28] Puruso Muhammad Hanunggul and Suyanto Suyanto, "The Impact of Local Attention in LSTM for Abstractive Text Summarization", 2019 International seminar on information technology and intelligent systems (ISRITI).

[29] Ruby, Usha, and Vamsidhar Yendapalli. "Binary cross entropy with deep learning technique for image classification." *Int. J. Adv. Trends Comput. Sci. Eng* 9.10 (2020).

[30] Iraj Naruei and Farshid Keynia, "A new optimization method based on COOT bird natural life model ", Expert Systems With Applications, vol. 183, 2021.

[31] Maha Mahmood and Belal Al-Khateeb, "The blue monkey: A new nature inspired metaheuristic optimization algorithm", Periodicals of Engineering and Natural Sciences, vol. 7, no. 3, 2019.
[32] https://duc.nist.gov/data.html.
[33] Song, S., Huang, H. and Ruan, T. Abstractive text summarization using LSTM-CNN based deep learning. Multimed Tools Appl 78, 857–875 (2019). https://doi.org/10.1007/s11042-018-5749-3.
[34] Kasimahanthi Divya, Kambala Sneha, Baisetti Sowmya, G Sankara Rao,"Text Summarization using Deep Learning", International Research Journal of Engineering and Technology (IRJET), vol. 7, 2020.

Biographies

Sunilkumar Ketineni received his M.Tech degree in JNTUK. He is currently pursing a Ph.D. in VIT-AP University, Andhra Pradesh, India. Areas of interest are natural language processing and deep learning. He has published five research papers in international journals and conferences of repute in data mining and NLP.

Sheela J has served VIT, Andhra Pradesh as Assistant Professor in the School of Computer Science and Engineering (SCOPE). She was a faculty

member at KITS, Warangal before joining VIT, Andhra Pradesh. She graduated with a B.Eng. from Sri Krishna College of Technology, Coimbatore affiliated to the Anna University University, Chennai, and obtained a Master of Engineering from Anna University Campus, with the third rank from Anna University, Coimbatore. She got her Ph.D. from National Institute of Technology, Tiruchirappalli, India. She cleared the National Eligibility Test for Lectureship by Tamil Nadu in 2016. Before beginning the Ph.D. program, Sheela worked as an Assistant Professor in the Hindustan College of Engineering and Technology. She has 6 years of teaching experience as an Assistance Professor in Anna University and Hindustan College of Engineering and Technology, Coimbatore. She has published over 20 research papers in international journals and conferences of repute in data mining and NLP.