



Applied Linguistics with Mixed Leader Optimizer Based English Text Summarization Model

Hala J. Alshahrani¹, Khaled Tarmissi², Ayman Yafoz³, Abdullah Mohamed⁴, Manar Ahmed Hamza^{5,*},
Ishfaq Yaseen⁵, Abu Sarwar Zamani⁵ and Mohammad Mahzari⁶

¹Department of Applied Linguistics, College of Languages, Princess Nourah Bint Abdulrahman University,
P.O. Box 84428, Riyadh, 11671, Saudi Arabia

²Department of Computer Sciences, College of Computing and Information System, Umm Al-Qura University,
Saudi Arabia

³Department of Information Systems, Faculty of Computing and Information Technology, King Abdulaziz University,
Jeddah, Saudi Arabia

⁴Research Centre, Future University in Egypt, New Cairo, 11845, Egypt

⁵Department of Computer and Self Development, Preparatory Year Deanship, Prince Sattam bin Abdulaziz University,
AlKharj, Saudi Arabia

⁶Department of English, College of Science & Humanities, Prince Sattam bin Abdulaziz University, AlKharj, Saudi Arabia

*Corresponding Author: Manar Ahmed Hamza. Email: ma.hamza@psau.edu.sa

Received: 29 July 2022; Accepted: 23 November 2022

Abstract: The term ‘executed linguistics’ corresponds to an interdisciplinary domain in which the solutions are identified and provided for real-time language-related problems. The exponential generation of text data on the Internet must be leveraged to gain knowledgeable insights. The extraction of meaningful insights from text data is crucial since it can provide value-added solutions for business organizations and end-users. The Automatic Text Summarization (ATS) process reduces the primary size of the text without losing any basic components of the data. The current study introduces an Applied Linguistics-based English Text Summarization using a Mixed Leader-Based Optimizer with Deep Learning (ALTS-MLODL) model. The presented ALTS-MLODL technique aims to summarize the text documents in the English language. To accomplish this objective, the proposed ALTS-MLODL technique pre-processes the input documents and primarily extracts a set of features. Next, the MLO algorithm is used for the effectual selection of the extracted features. For the text summarization process, the Cascaded Recurrent Neural Network (CRNN) model is exploited whereas the Whale Optimization Algorithm (WOA) is used as a hyperparameter optimizer. The exploitation of the MLO-based feature selection and the WOA-based hyperparameter tuning enhanced the summarization results. To validate the performance of the ALTS-MLODL technique, numerous simulation analyses were conducted. The experimental results signify the superiority of the proposed ALTS-MLODL technique over other approaches.

Keywords: Text summarization; deep learning; hyperparameter tuning; applied linguistics; multi-leader optimizer



This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

1 Introduction

The web resources available on the Internet (for example, user reviews, websites, news, social networking sites, blogs, etc.) are gigantic sources of text datasets. Further, the text dataset is available in other forms, such as the archives of books, news articles, legal documents, journals, scientific papers, biomedical documents, etc. [1]. There is a dramatic increase experienced in the volume of text data available on the Internet and other archives. Consequently, a user consumes considerable time to find the required information [2]. As a result, it becomes crucial to summarize and condense the text resource to make it meaningful. However, manual summarization is highly challenging and consumes a lot of time and effort [3]. It is challenging for human beings to manually summarize huge volumes of textual datasets. The automated Text Summarization (ATS) process can be a major conclusion to this dilemma. The most important goal of the ATS system is to generate a low-volume summary that encompasses all the major ideas from the input document [4].

The automated Text Summarization (ATS) process is challenging in nature. When human beings try to summarize a text, the content is completely read and understood, and the key points are prepared. However, this is not the case in terms of the ATS process due to insufficient language processing and human knowledge capabilities in computers. So, automated text summarization is a difficult process [5]. ATS system is categorized into single- or multiple-document summarization schemes. Initially, the system generates a summary for a single document, producing the summaries for a cluster of documents. The ATS system aims to employ the text summarization methodologies such as hybrid, extractive and abstractive to achieve the outcomes [1]. Amongst these, the extractive method selects the main sentences from the input text and uses them to produce the summary. The hybrid method incorporates both extractive and abstractive methodologies [6]. The ATS system is primarily used in text analytics and mining applications like question answering, information retrieval, information extraction, etc. This system has been utilized with information retrieval techniques to improve the search engine's abilities [7].

Due to the wide accessibility of the internet, a considerable number of study opportunities are available in the field of ATS with Natural Language Processing (NLP), especially based on statistical Machine Learning (ML) techniques. The primary aim of the ATS approach is to generate a summary similar to a human-generated summary [8]. Nonetheless, in many cases, both the readability and the soundness of the generated summary are unacceptable. This is because the summary doesn't encompass each semantically-consistent feature of the data. Most of the recent TS approaches do not have an individual point-of-view over the semantics of the words [9]. To efficiently overcome the issues present in the existing TS methods, the current study presents a new architecture through the Deep Learning approach.

The current study introduces an Applied Linguistics-based English Text Summarization using a Mixed Leader-Based Optimizer with deep learning (ALTS-MLODL) model. The presented ALTS-MLODL technique aims to summarize the text documents in the English language. To accomplish the objective, the proposed ALTS-MLODL technique pre-processes the input documents and primarily extracts a set of features. Next, the MLO algorithm is used for the productive selection of the extracted features. The Cascaded Recurrent Neural Network (CRNN) model is exploited with Whale Optimization Algorithm (WOA) as a hyperparameter optimizer for the text summarization process. Exploiting the MLO-based feature selection and the WOA-based hyperparameter tuning enhanced the summarization results. Numerous simulation analyses were conducted to observe the better outcomes of the proposed ALTS-MLODL technique.

The rest of the paper is organized as follows. Section 2 provides a brief overview of text summarization approaches. Next, Section 3 introduces the proposed model and Section 4 offers performance validation. Finally, Section 5 draws the concluding remarks of the study.

2 Existing Text Summarization Approaches

Wan et al. [10] devised an innovative structure to address the tasks by extracting many summaries and ranking them in the targeted language. At first, the authors extracted many candidate summaries by presenting numerous methods to improve the quality of the upper-bound summary. Then, the author designed a novel ensembling ranking technique to rank the summaries of the candidates using the bilingual features. Song et al. [11] presented an LSTM-CNN-related ATS framework (ATSDL) that constructed innovative sentences by exploring the finely-grained fragments against the established sentences, like semantic phrases. Unlike the existing extraction methods, the ATSDL approach had two main phases: extracting phrases from source sentences and developing the text summaries using DL techniques. D'silva et al. [12] explored the summary extraction domain and proposed an automated text summarization method with the help of the DL technique. The authors implemented the method in the Konkani language since it is considered a language with fewer resources. Here, the resources correspond to the availability of inadequate resources, namely speakers, data, experts and tools in the Konkani language. The presented method used a fast-text pre-trained word embedding Facebook-based method to receive the vector representations for sentences. Afterwards, a deep multi-layer perceptron method was used as a supervised binary classifier task for auto-generating the summaries utilizing the feature vectors.

Maylawati et al. [13] introduced an ideology by combining the DL and SPM methods for a superior text summarization outcome. In the text summarization process, generating readable and understandable summaries is significant. SPM, as a text representation-extraction method, can maintain the meaning of the text by showing interest in the order of words' appearance. DL is a famous and powerful ML approach broadly utilized in several data mining research works. It employs a descriptive research method that gathers every fact about the data based on DL and SPM techniques for text summarization. Here, the NLP technique is used as knowledge, whereas DL and the SPM techniques are applied for text summarization since it is the main issue that needs to be sorted. Khan et al. [14] focused on the extraction-based summarization process with the help of K-Means clustering and Term Frequency-Inverse Document Frequency (TFIDF). This study reflected the ideology of true K and utilized the value of K to split the sentences that belonged to the input document to arrive at the concluding summary.

Lin et al. [15] proposed a simple yet effective extraction technique in the name of the Light Gradient Boosting Machine regression method for Indonesian files. In this method, four features were derived such as the TitleScore, PositionScore, the semantic representation similarity between the title of the document and the sentence and the semantic representation similarity between a sentence's cluster center and the sentence. The author described a formula to calculate the sentence score as its objective function. Zhao et al. [16] presented a Variational Neural Decoder text summarization approach (VND). This method presented a series of implicit variables by integrating the modified AE and the modified RNN to be utilized in capturing the complicated semantic representation at every decoding stage. It involved a variational RNN layer and a standard RNN layer. These two network layers produced a random hidden state and a deterministic hidden state. The author used two such RNN layers to establish the dependency between the implicit variables and the adjacent time steps.

3 The Proposed Text Summarization Model

In this article, a new ALTS-MLODL technique has been introduced as an effectual text summarization process. The presented ALTS-MLODL technique aims to summarize the text documents in the English language. Fig. 1 depicts the working process of the ALTS-MLODL approach.

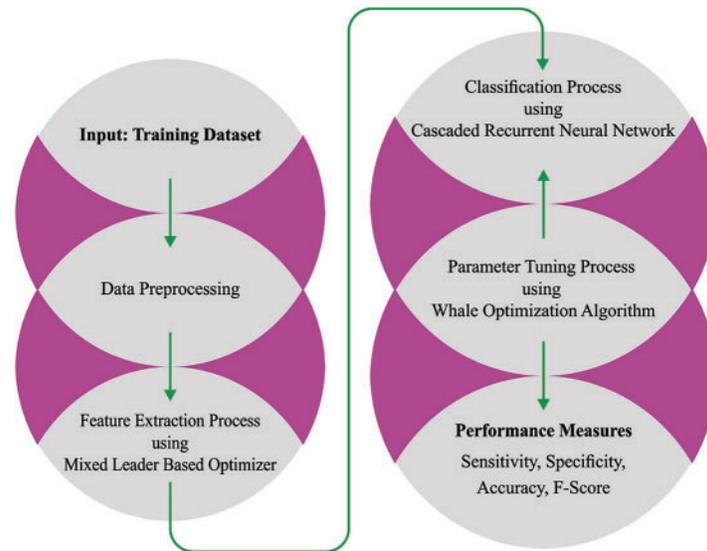


Figure 1: Working process of the ALTS-MLODL approach

3.1 Pre-Processing and Feature Extraction

At first, the proposed ALTS-MLODL technique pre-processes the input documents and primarily extracts a set of features. The basic concept of the proposed model is to create an input text document so it can be processed in multiple phases [17]. It transmits the input document since it contains important data. The presented method involves the following pre-processing sequencing functions: stemming process, letter normalization, tokenization and the removal of stop-words. During the feature extraction procedure, the sentences of special significance or importance are chosen in the group of features with a coherent summary to demonstrate the main issue of the offered documents. The pre-processed input document is represented as a vector of features. It contains elements that are employed to signify the summarized sentences. In this study, 15 features were extracted in total. All the features provide a disparate value. The maximum score values indicate a lesser occurrence of the features, whereas the minimal values correspond to a higher occurrence of the features from the sentence. To extract a summary, every sentence is ordered based on the score of words from all the sentences.

With the help of distinct features such as the term cue words, frequency and phrase, and the lexical measure, the words are identified by the scores. At this point, the pre-processed input document is represented as a vector of features. These elements are employed to signify the summary of the sentences described from F1 to F14.

3.2 Feature Selection Using MLO Algorithm

In this stage, the MLO algorithm is used to select the extracted features effectively. MLO is a population-based optimization technique that randomly produces a specific number of potential solutions for optimization problems [18]. Then, the MLO approach upgrades the presented solution through the iteration method. After the repetitions, the MLO method offers an appropriate quasi-optimum solution. The major concept used in guiding and updating the population is that a new member is utilized as the population leader and is generated by blending the optimal members of the population using an arbitrary member.

The population of the MLO approach is described as a matrix named ‘population matrix’ as given below.

$$X = \begin{bmatrix} X_1 \\ \vdots \\ X_i \\ \vdots \\ X_N \end{bmatrix}_{N \times m} = \begin{bmatrix} x_{1,1} & \dots & x_{1,d} & \dots & x_{1,m} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ x_{i,1} & \dots & x_{i,d} & \dots & x_{i,m} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ x_{N,1} & \dots & x_{N,d} & \dots & x_{N,m} \end{bmatrix}_{N \times m} \quad (1)$$

In Eq. (1), X denotes the population, X_i indicates the i -th population member, N shows the number of population members, m indicates the number of problem parameters, $x_{i,d}$ indicates the value of the d -th problem parameter that is recommended by the i -th population member.

All the members of the population matrix are considered viable solutions for the set value of the problem variable. Consequently, a value is attained for the objective function based on the values indicated by all the members of the problem parameter, which is shown below.

$$F = \begin{bmatrix} F_1 \\ \vdots \\ F_i \\ \vdots \\ F_N \end{bmatrix}_{N \times 1} = \begin{bmatrix} F_1(X_1) \\ \vdots \\ F_i(X_i) \\ \vdots \\ F_N(X_N) \end{bmatrix}_{N \times 1} \quad (2)$$

In Eq. (2), F refers to an objective function vector, and F_i indicates an objective function value for the i -th population member.

During all the iterations, the population member that provides the optimal value for the objective function is regarded as the optimum population member. An arbitrary member is produced as given below as a possible solution.

$$x_d^R = x_d^{lo} + r \times (x_d^{hi} - x_d^{lo}) \quad (3)$$

In Eq. (3), x_d^R refers to the d -th dimension of an arbitrary member, x_d^{lo} denotes the lower limit of the d -th problem parameter, x_d^{hi} shows the upper limit of the d -th problem parameter, and r indicates an arbitrary value within $[0, 1]$. In all the iterations, the optimum member is upgraded after which a random value is generated. Then, the mixed leader of the population is upgraded based on two members, as given below.

$$x_d^{ML} = (1 - I_M) \times x_d^{best} + I_M \times x_d^R \quad (4)$$

$$I_M = \begin{cases} (1 - \frac{t}{T}), & t < \frac{T}{2} \\ 0, & otherwise \end{cases} \quad (5)$$

Now, x_d^{ML} refers to the d -th dimension of a mixed leader, I_M indicates the mixed index, x_d^{best} denotes the d -th dimension of the optimum member, T shows the iteration counter, and T indicates the maximal number of iterations.

The population matrix, upgraded in the MLO, is inspired by the mixed leader of the population, as given below.

$$dx_{i,d} = \begin{cases} r \times (x_d^{ML} - x_{i,d}), & F_{ML} < F_i \\ r \times (x_{i,d} - x_d^{ML}), & else \end{cases} \quad (6)$$

$$x_{i,d}^{new} = x_{i,d} + dx_{i,d} \quad (7)$$

$$X_i = \begin{cases} X_i^{new}, & F_i^{new} < F_i \\ X_i, & else \end{cases} \quad (8)$$

Now, $dx_{i,d}$ indicates the displacement value for the i -th population member in the d -th dimension, F_{ML} denotes the objective function value of the mixed leader, $x_{i,d}^{new}$ denotes the d -th dimension of the newly-recommended location for the i -th population member, and F_i^{new} indicates the objective function value for the newly-recommended location of the i -th population member.

3.3 Text Summarization Using Optimal CRNN Model

For the text summarization process, the WOA approach is exploited in this study with the CRNN model. CRNN is a deep-cascaded network that contains a front-end network to extract the word mappings [19], a back-end network to exploit the deep semantic contexts and a CNN to extract the CNN features. Fig. 2 demonstrates the infrastructure of the CRNN method.

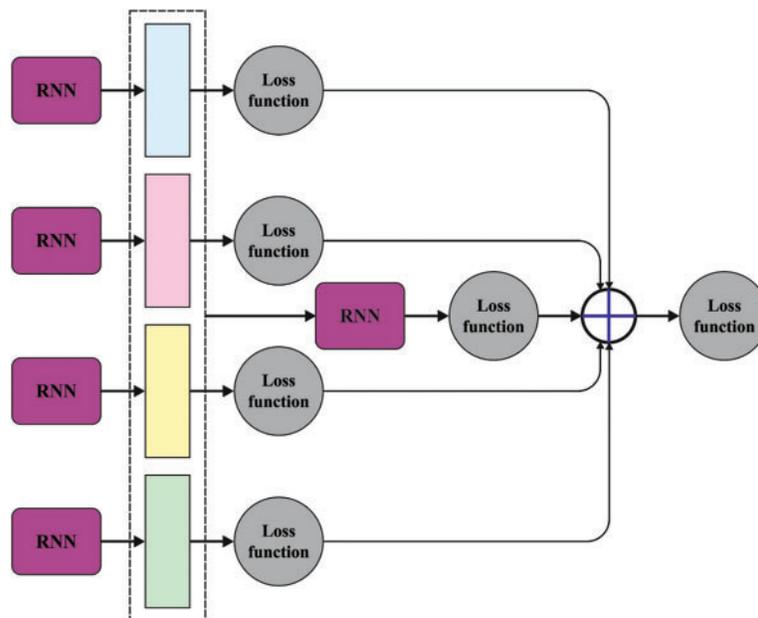


Figure 2: Structure of the CRNN approach

Like Google NIC, the front-end network adapts the encoder-decoder structure to extract the visual language interaction approach from the forward direction. The suggested method contains the output layers, the word input layers, the dense embedding layers and the recurrent layers. To learn the word mappings efficiently, the authors directly optimized the amount of probability for an accurate representation as determined in (9) through stochastic gradient descent.

$$\log p(w|I) = \sum_{t=0}^N \log p(w_t|I, w_0, \dots, w_{t-1}) \quad (9)$$

In Eq. (9), I represent a CNN feature, and $w_{0:t-1}$ denotes the past textual context. w_t indicates the predicted word, w shows an accurate representation and N represents the length of the sentence.

In the case of a front-end network, two incorporated layers are employed to implement a one-hot vocabulary into a dense-word expression. This might enhance the semantic meaning of the words. Furthermore, an SGRU is also designed to map the deep words.

The SGRU approach generates all the recurrent units to capture the dependency of the distinct time scales adaptively. It contains two hidden states with a gating unit to modulate the data flow inside the unit instead of employing individual memory cells. The activation h_t of every gate at time t is a linear integration of the candidate activation h'_t and the preceding activation h_{t-1} as given below.

$$h_t = (1 - z_t) h_{t-1} + z_t h'_t \quad (10)$$

In Eq. (10), z_t represents the way the content is updated by the unit, which is determined as follows.

$$z_t = \sigma(W_z \cdot x_t + U_z h_{t-1}) \quad (11)$$

In Eq. (11), σ denotes the sigmoid function, and x_t corresponds to the input unit. The h'_t candidate activation is formulated as follows.

$$h'_t = \tan h(W \cdot x_t + U(r_t \odot h_{t-1})) \quad (12)$$

r_t indicates a set of reset gates and \odot denotes the element-wise multiplication. r_t reset gate is calculated based on the update gate as follows

$$r_t = \sigma(W_r \cdot x_t + U_r h_{t-1}) \quad (13)$$

In Eq. (13), W_z, W_r, W, U_z, U_r, U denote the project matrices that map distinct parts into respective feature spaces. The activation of the SGRU model is executed by the *softmax* layer to obtain a likelihood distribution of the following words.

For optimal fine-tuning of the hyperparameter values, the WOA method is utilized. WOA is a recent metaheuristic approach that is stimulated by the social behaviours of humpback whales [20]. This approach starts with the arbitrary generation of a set of N solutions 'TH' that signifies the solution for the provided problem. Next, for all the solutions $TH_i, i = 1, 2, \dots, N$, the objective function is calculated, and the optimum solution is defined as TH^* . Then, all the solutions are upgraded via the bubble-net algorithm or the encircling algorithm. In the bubble-net algorithm, the existing solution TH_i is upgraded through the shrinking-encircling algorithm, where the value of a gets reduced as follows:

$$a = a - a \frac{g}{g_{\max}}. \quad (14)$$

In Eq. (14), g and g_{\max} denote the existing iteration and the maximal amount of iterations correspondingly. Further, the solution TH_i gets upgraded through the encircling method as follows.

$$TH_i(g+1) = TH_i(g) - A \odot D, A = 2a \odot r_1 - a, \quad (15)$$

$$D = |C \odot TH^*(g) - TH_i(g)|, B = 2r_2, \quad (16)$$

Here, D denotes the distance between TH^* and TH_i during the g -th iteration. r_1 and r_2 denote the arbitrary numbers and \odot denote the element-wise multiplication. Furthermore, the value of a gets reduced between $[2, 0]$ with an increasing number of iterations.

The solution TH_i gets upgraded based on the spiral method, which in turn stimulates the helix-shaped movement around TH^* as follows.

$$TH_i(g+1) = D' \odot e^{bl} \odot \cos(2\pi l) + TH_i(g), D' = |TH^*(g) - TH_i(g)|, \quad (17)$$

In Eq. (17), $l \in [-1, 1]$ and b indicate the random variable and the constant values utilized for determining the shape of the logarithmic spiral.

Furthermore, the solution in the WOA gets upgraded based on the spiral-shaped path and shrinking path as follows.

$$TH_i(g+1) = \begin{cases} TH^*(g) - A \odot D & \text{if } r_3 \geq 0.5 \\ D' \odot e^{bl} \odot \cos(2\pi l) + TH_i(g) & \text{otherwise} \end{cases} \quad (18)$$

In Eq. (18), $r_3 \in [0, 1]$ characterizes the probability of switching amid the shrinking- and spiral-shaped path algorithms.

Also, the whales' search on the TH^* through a random solution, TH_r is given below.

$$TH_i(g+1) = TH_r - A \odot D, D' = |TH_r(g) - TH_i(g)| \quad (19)$$

The procedure of upgrading the solution is executed based on a , A , C , and r_3 . The existing solution TH_i is upgraded as follows. If $r_3 \geq 0.5$, it is upgraded by the Eqs. (17) and (18), if $|A| < 1$ or if $|A| \geq 1$. The procedure of upgrading the solution is repeated until the ending condition is met.

4 Results and Discussion

In this section, the text summarization results of the proposed ALTS-MLODL model are examined under two aspects such as single-document summarization and multi-document summarization. The proposed model is simulated using Python 3.6.5 tool on PC i5-8600 k, GeForce 1050Ti 4 GB, 16 GB RAM, 250 GB SSD, and 1 TB HDD. The parameter settings are given as follows: learning rate: 0.01, dropout: 0.5, batch size: 5, epoch count: 50, and activation: ReLU.

Table 1 provides the detailed summarization results of the proposed ALTS-MLODL model under varying file sizes on single-document summarization [17]. Fig. 3 portrays the comparative $sens_y$ inspection results achieved by the proposed ALTS-MLODL model on a single-document summarization process. The figure implies that the proposed ALTS-MLODL model accomplished an improved summarization result under all types of file sizes. For instance, with a file size of 1,000 Kb, the ALTS-MLODL model produced an increased $sens_y$ of 85.74%, whereas the DL-MNN, ANN, KELM and ELM models achieved low $sens_y$ values such as 82.88%, 81.14%, 79.93% and 78.46% respectively. Meanwhile, with a file size of 5,000 Kb, the ALTS-MLODL model gained an improved $sens_y$ of 98.44%,

whereas the DL-MNN, ANN, KELM and ELM models achieved the least $sens_y$ values such as 95.20%, 94.04%, 92.96% and 91.88% respectively.

Fig. 4 depicts the detailed $spec_y$ analysis results of the proposed ALTS-MLODL approach on the single-document summarization process. The figure denotes that the proposed ALTS-MLODL method achieved enhanced summarization results under all types of file sizes. For example, with a file size of 1,000 Kb, the ALTS-MLODL method produced an increased $spec_y$ of 85.39%, whereas the DL-MNN, ANN, KELM and ELM model reached low $spec_y$ values such as 82.49%, 80.11%, 78.84% and 77.66% correspondingly. With a file size of 5,000 Kb, the proposed ALTS-MLODL technique achieved an improved $spec_y$ of 95.01%, whereas the DL-MNN, ANN, KELM and ELM approaches accomplished low $spec_y$ values such as 92.48%, 90.32%, 88% and 86.22% correspondingly.

Table 1: Analytical results of the ALTS-MLODL approach on the single-document summarization process under distinct file sizes

Single document					
File size (Kb)	ALTS-MLODL	DL-MNN model	ANN model	KELM model	ELM model
Sensitivity					
1000	85.74	82.88	81.14	79.93	78.46
2000	90.02	86.57	84.11	83.01	80.71
3000	92.46	89.94	88.93	87.32	84.85
4000	96.75	93.39	91.68	89.94	88.81
5000	98.44	95.20	94.04	92.96	91.88
Specificity					
1000	85.39	82.49	80.11	78.84	77.66
2000	88.23	84.75	82.53	81.28	78.93
3000	91.32	88.19	85.83	84.78	83.30
4000	93.84	91.33	90.29	88.63	87.35
5000	95.01	92.48	90.32	88.00	86.22
Accuracy					
1000	87.23	83.83	81.78	79.36	77.92
2000	89.94	87.16	84.66	83.45	81.29
3000	92.26	89.52	88.00	86.48	84.52
4000	95.25	91.67	89.33	87.72	86.01
5000	97.67	94.23	91.80	89.69	88.38
F-score					
1000	87.21	83.71	82.17	80.82	79.33
2000	89.16	86.48	84.51	83.17	81.86
3000	91.33	88.65	87.58	86.13	84.72
4000	94.96	92.30	91.21	89.00	87.22
5000	96.76	94.22	92.54	91.04	89.66

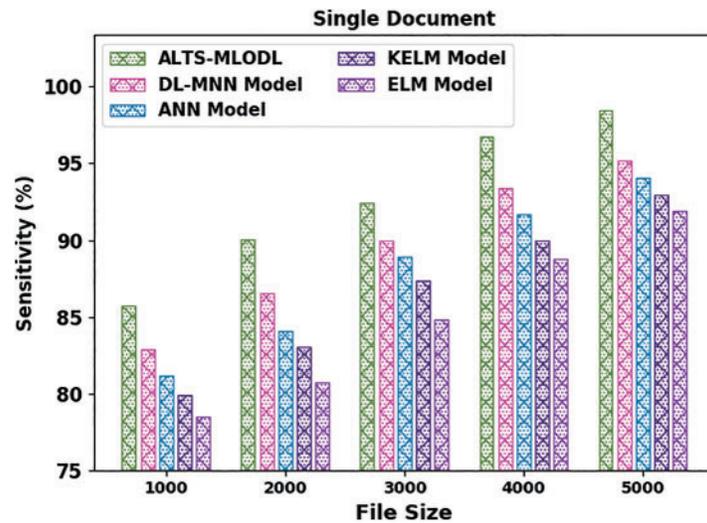


Figure 3: $Sens_y$ analysis results of the ALTS-MLODL approach on the single-document summarization process

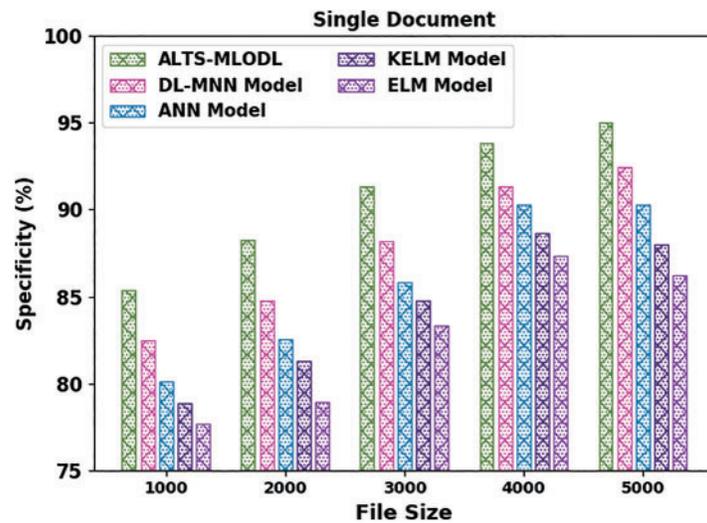


Figure 4: $Spec_y$ analysis results of the ALTS-MLODL approach on the single-document summarization process

Fig. 5 shows the comparative $accu_y$ analysis results accomplished by the proposed ALTS-MLODL approach to the single-document summarization process. The figure infers that the ALTS-MLODL method exhibited improved summarization results under all types of file sizes. For example, with a file size of 1,000 Kb, the proposed ALTS-MLODL technique produced an increased $accu_y$ of 87.23%, whereas the DL-MNN, ANN, KELM and ELM algorithm attained the least $accu_y$ values such as 83.83%, 81.78%, 79.36% and 77.92% correspondingly. In the meantime, with a file size of 5,000 Kb, the proposed ALTS-MLODL approach acquired an improved $accu_y$ of 97.67%, whereas the DL-MNN, ANN, KELM and ELM approach portrayed the least $accu_y$ values such as 94.23%, 91.80%, 89.69% and 88.38% correspondingly.

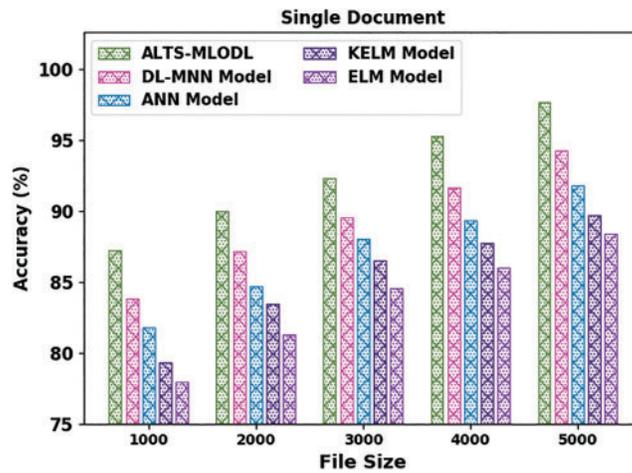


Figure 5: $Accu_y$ analysis results of the ALTS-MLODL approach on the single-document summarization process

Fig. 6 represents the comparative F_{score} review results achieved by the proposed ALTS-MLODL method on the single-document summarization process. The figure denotes that the ALTS-MLODL model established improved summarization results under all types of file sizes. For example, with a file size of 1,000 Kb, the proposed ALTS-MLODL algorithm produced an increased F_{score} of 87.21% whereas the DL-MNN, ANN, KELM and ELM model accomplished the least F_{score} values such as 83.71%, 82.17%, 80.82% and 79.33% correspondingly. Simultaneously, with a file size of 5,000 Kb, the proposed ALTS-MLODL approach reached an improved $sens_y$ of 96.76% whereas the DL-MNN, ANN, KELM and ELM methodology achieved the least F_{score} values at 94.22%, 92.54%, 91.04% and 89.66% correspondingly.

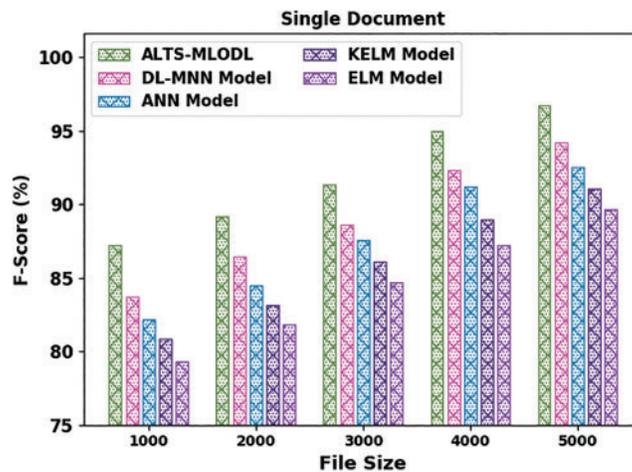


Figure 6: F_{score} analysis results of the ALTS-MLODL approach on the single-document summarization process

Both Training Accuracy (TRA) and Validation Accuracy (VLA) values, acquired by the proposed ALTS-MLODL methodology under a single-document summarization process, are shown in Fig. 7.

The experimental outcomes denote that the ALTS-MLODL approach attained the maximal TRA and VLA values whereas the VLA values were higher than the TRA values.

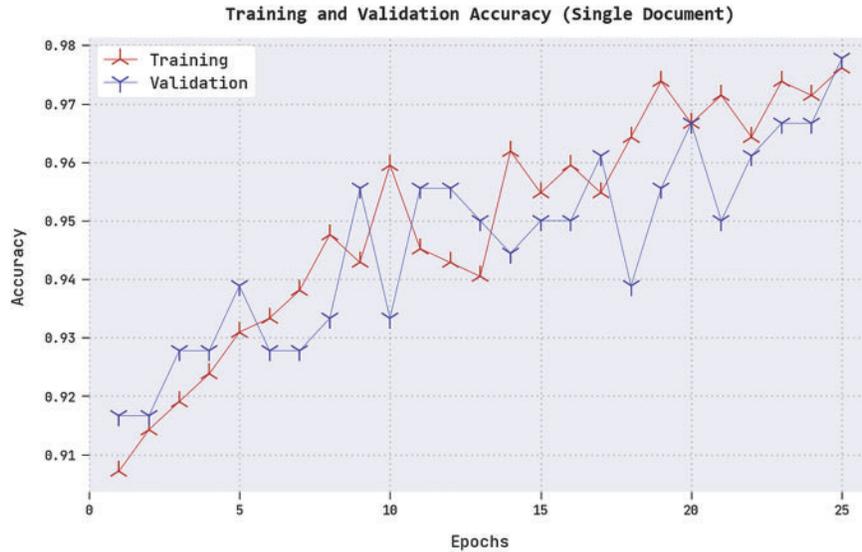


Figure 7: TRA and VLA analyses results of the ALTS-MLODL approach on the single document summarization process

Both Training Loss (TRL) and Validation Loss (VLL) values, gained by the proposed ALTS-MLODL technique under a single-document summarization process, are displayed in Fig. 8. The experimental outcomes infer that the ALTS-MLODL approach displayed the least TRL and VLL values whereas the VLL values were lesser than the TRL values.

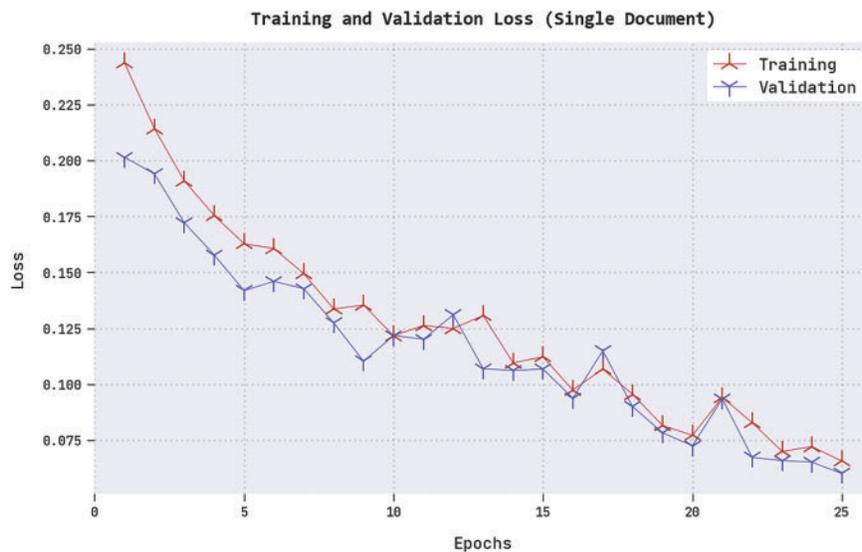


Figure 8: TRL and VLL analyses results of the ALTS-MLODL approach on the single-document summarization process

Table 2 offers the comprehensive summarization outcomes of the ALTS-MLODL algorithm under varying file sizes on the multi-document summarization process. Fig. 9 represents the detailed $sens_y$ examination outcomes achieved by the proposed ALTS-MLODL algorithm on the multi-document summarization process. The figure denotes that the proposed ALTS-MLODL approach established improved summarization results under all types of file sizes. For example, with a file size of 1,000 Kb, the proposed ALTS-MLODL method produced an increased $sens_y$ of 86.44% whereas the DL-MNN, ANN, KELM and the ELM method produced the least $sens_y$ of 83.27%, 81.39%, 79.72% and 77.83% correspondingly. In parallel, with a file size of 5,000 Kb, the proposed ALTS-MLODL approach obtained an improved $sens_y$ of 96.87% whereas the DL-MNN, ANN, KELM and ELM algorithms achieved the least $sens_y$ values such as 93.46%, 92.13%, 89.80% and 88.65% correspondingly.

Table 2: Analytical results of the ALTS-MLODL approach on the multi-document summarization process under distinct file sizes

Multi-document					
File size (Kb)	ALTS-MLODL	DL-MNN model	ANN model	KELM model	ELM model
Sensitivity					
1000	86.44	83.27	81.39	79.72	77.83
2000	88.78	85.86	84.27	82.47	80.03
3000	91.59	88.63	86.26	84.54	82.37
4000	94.49	91.40	89.37	87.99	86.69
5000	96.87	93.46	92.13	89.80	88.65
Specificity					
1000	86.20	83.24	81.96	80.05	77.84
2000	87.07	84.42	82.28	80.13	78.85
3000	88.47	85.86	84.41	82.03	80.96
4000	89.46	86.92	84.72	83.68	81.78
5000	95.30	89.50	87.53	85.70	83.27
Accuracy					
1000	86.36	83.69	82.63	81.36	79.66
2000	90.61	87.45	85.92	84.11	82.97
3000	94.15	90.94	88.47	87.02	84.81
4000	95.80	92.24	90.24	88.53	86.48
5000	97.50	95.71	93.43	91.39	90.22
F-score					
1000	87.30	83.97	81.55	79.47	76.98
2000	89.90	87.05	85.64	84.02	82.46
3000	92.89	90.17	88.60	86.49	84.25
4000	96.26	93.43	91.17	89.40	88.06
5000	97.99	96.98	95.03	93.92	92.62

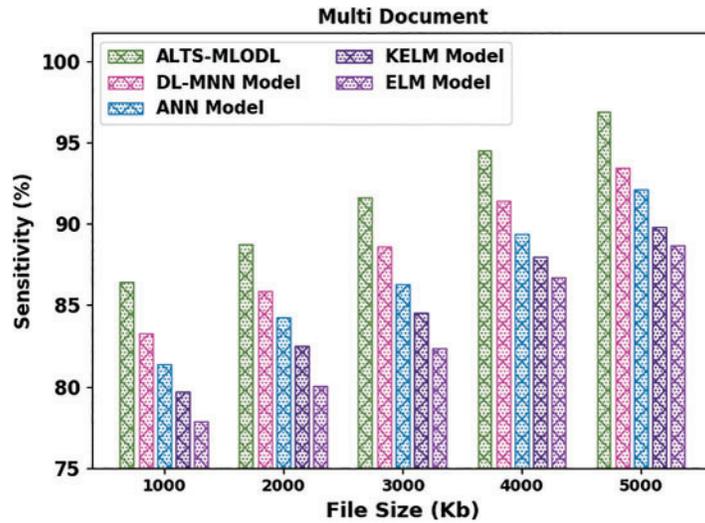


Figure 9: $Sens_y$ analysis results of the ALTS-MLODL approach on the multi-document summarization process

Fig. 10 describes the comparative $spec_y$ analysis results of the proposed ALTS-MLODL methodology on the multi-document summarization process. The figure implies that the proposed ALTS-MLODL approach exhibited improved summarization results under all types of file sizes. For example, with a file size of 1,000 Kb, the ALTS-MLODL technique achieved a maximum $spec_y$ of 86.20% whereas the DL-MNN, ANN, KELM and the ELM algorithm reached the least $spec_y$ values such as 83.24%, 81.96%, 80.05% and 77.84% correspondingly. In the meantime, with a file size of 5,000 Kb, the ALTS-MLODL methodology acquired an improved $spec_y$ of 95.30% whereas the DL-MNN, ANN, KELM and ELM approach achieved the least $spec_y$ values such as 89.50%, 87.53%, 85.70% and 83.27% correspondingly.

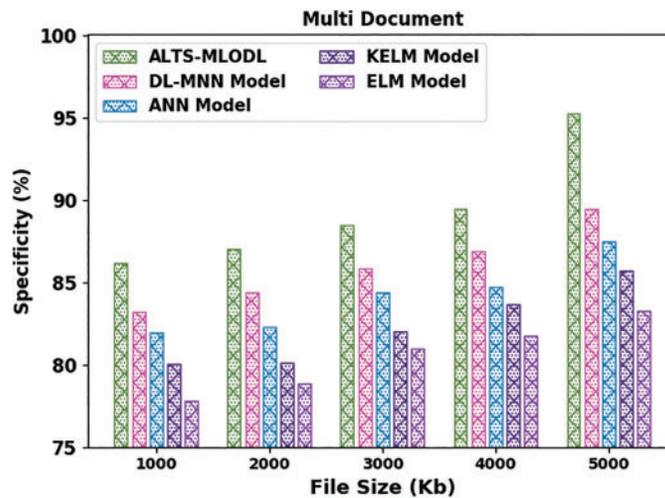


Figure 10: $Spec_y$ analysis results of the ALTS-MLODL approach on the multi-document summarization process

Fig. 11 portrays the comprehensive $accu_y$ analysis outcomes of the ALTS-MLODL method on the multi-document summarization process. The figure indicates that the ALTS-MLODL method established improved summarization results under all types of file sizes. For example, with a file size of 1,000 Kb, the proposed ALTS-MLODL approach produced an increased $accu_y$ of 86.36% whereas the DL-MNN, ANN, KELM and ELM techniques reached the minimum $accu_y$ values such as 83.69%, 82.63%, 81.36% and 79.66% correspondingly. In parallel, with a file size of 5,000 Kb, the proposed ALTS-MLODL method attained an improved $accu_y$ of 95.30% whereas the DL-MNN, ANN, KELM and ELM model achieved low $accu_y$ values such as 89.50%, 87.53%, 85.70%, and 83.27% correspondingly.

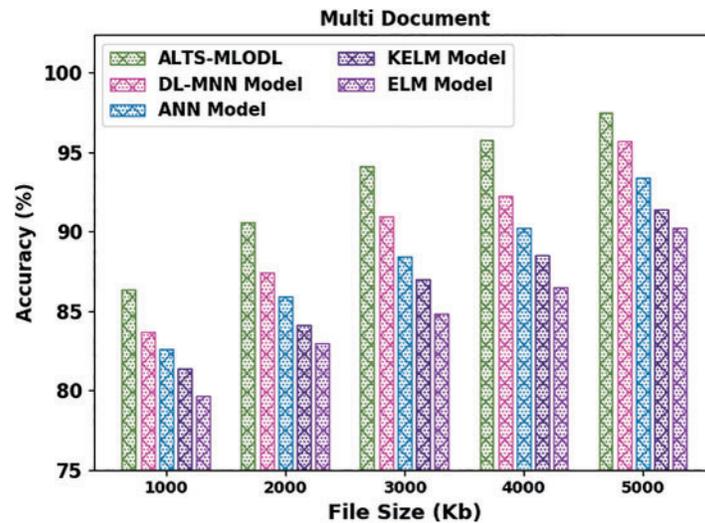


Figure 11: $Accu_y$ analysis results of the ALTS-MLODL approach on the multi-document summarization process

Fig. 12 showcases the detailed F_{score} analysis outcomes achieved by the proposed ALTS-MLODL methodology on the multi-document summarization process. The figure implies that the ALTS-MLODL approach exhibited improved summarization results under all types of file sizes. For example, with a file size of 1,000 Kb, the ALTS-MLODL model resulted in an increased F_{score} of 87.30% whereas the DL-MNN, ANN, KELM and ELM approach achieved the least F_{score} values such as 83.97%, 81.55%, 79.47% and 76.98% correspondingly. Meanwhile, with a file size of 5,000 Kb, the proposed ALTS-MLODL algorithm gained an improved $sens_y$ of 97.99% whereas the DL-MNN, ANN, KELM and ELM models accomplished the least F_{score} values such as 96.98%, 95.03%, 93.92% and 92.62% correspondingly. These results infer that the proposed ALTS-MLODL model achieved enhanced summarization outcomes than the existing summarization approaches.

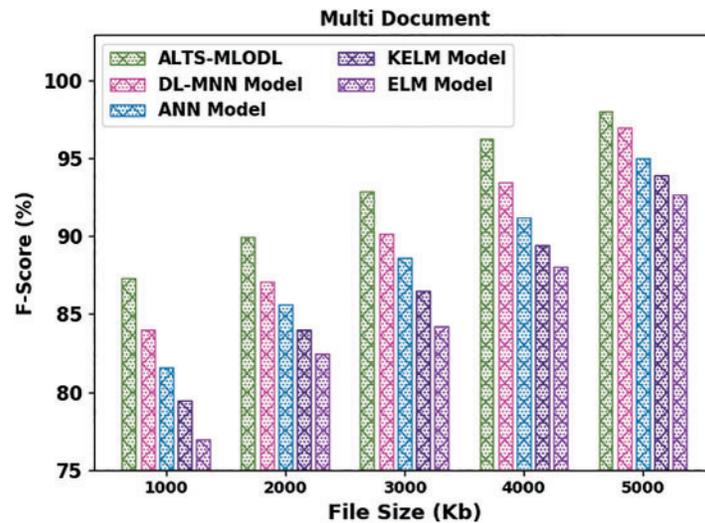


Figure 12: F_{score} analysis results of the ALTS-MLODL approach on the multi-document summarization process

5 Conclusion

In this article, a new ALTS-MLODL technique has been developed for an effectual text summarization outcome. The presented ALTS-MLODL technique aims to summarize the text documents in the English language. To accomplish the objective, the proposed ALTS-MLODL technique pre-processes the input documents and primarily extracts a set of features. Next, the MLO algorithm is used for the effectual selection of the extracted features. For the text summarization process, the CRNN model is exploited with WOA as a hyperparameter optimizer. The exploitation of the MLO-based feature selection and the WOA-based hyperparameter tuning enhanced the summarization results. To exhibit the superior performance of the proposed ALTS-MLODL technique, numerous simulation analyses were conducted. The experimental results signify the superiority of the proposed ALTS-MLODL technique over other approaches. In the future, hybrid DL models can be utilized for ATS and image captioning processes.

Funding Statement: Princess Nourah bint Abdulrahman University Researchers Supporting Project Number (PNURSP2022R281), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia. The authors would like to thank the Deanship of Scientific Research at Umm Al-Qura University for supporting this work by Grant Code: (22UQU4331004DSR09).

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

- [1] E. Vázquez, R. A. Hernández and Y. Ledeneva, "Sentence features relevance for extractive text summarization using genetic algorithms," *Journal of Intelligent & Fuzzy Systems*, vol. 35, no. 1, pp. 353–365, 2018.
- [2] R. M. Alguliyev, R. M. Aliguliyev, N. R. Isazade, A. Abdi and N. Idris, "COSUM: Text summarization based on clustering and optimization," *Expert Systems*, vol. 36, no. 1, pp. e12340, 2019.

- [3] A. Qaroush, I. A. Farha, W. Ghanem, M. Washaha and E. Maali, "An efficient single document arabic text summarization using a combination of statistical and semantic features," *Journal of King Saud University-Computer and Information Sciences*, vol. 33, no. 6, pp. 677–692, 2021.
- [4] K. Yao, L. Zhang, D. Du, T. Luo, L. Tao *et al.*, "Dual encoding for abstractive text summarization," *IEEE Transactions on Cybernetics*, vol. 50, no. 3, pp. 985–996, 2018.
- [5] R. Elbarougy, G. Behery and A. El Khatib, "Extractive arabic text summarization using modified PageRank algorithm," *Egyptian Informatics Journal*, vol. 21, no. 2, pp. 73–81, 2020.
- [6] N. Nazari and M. A. Mahdavi, "A survey on automatic text summarization," *Journal of AI and Data Mining*, vol. 7, no. 1, pp. 121–135, 2019.
- [7] N. Landro, I. Gallo, R. L. Grassa and E. Federici, "Two new datasets for Italian-language abstractive text summarization," *Information*, vol. 13, no. 5, pp. 228, 2022.
- [8] Y. Kumar, K. Kaur and S. Kaur, "Study of automatic text summarization approaches in different languages," *Artificial Intelligence Review*, vol. 54, no. 8, pp. 5897–5929, 2021.
- [9] S. N. Turkey, A. S. A. Al-Jumaili and R. K. Hasoun, "Deep learning based on different methods for text summary: A survey," *Journal of Al-Qadisiyah for Computer Science and Mathematics*, vol. 13, no. 1, pp. 26, 2021.
- [10] X. Wan, F. Luo, X. Sun, S. Huang and J. G. Yao, "Cross-language document summarization via extraction and ranking of multiple summaries," *Knowledge and Information Systems*, vol. 58, no. 2, pp. 481–499, 2019.
- [11] S. Song, H. Huang and T. Ruan, "Abstractive text summarization using LSTM-CNN based deep learning," *Multimedia Tools and Applications*, vol. 78, no. 1, pp. 857–875, 2019.
- [12] J. D'silva and U. Sharma, "Automatic text summarization of konkani texts using pre-trained word embeddings and deep learning," *International Journal of Electrical and Computer Engineering*, vol. 12, no. 2, pp. 1990, 2022.
- [13] D. S. Maylawati, Y. J. Kumar, F. B. Kasmin and M. A. Ramdhani, "An idea based on sequential pattern mining and deep learning for text summarization," *Journal of Physics: Conference Series*, vol. 1402, no. 7, pp. 077013, 2019.
- [14] R. Khan, Y. Qian and S. Naeem, "Extractive based text summarization using k-means and tf-idf," *International Journal of Information Engineering and Electronic Business*, vol. 11, no. 3, pp. 33, 2019.
- [15] N. Lin, J. Li and S. Jiang, "A simple but effective method for Indonesian automatic text summarisation," *Connection Science*, vol. 34, no. 1, pp. 29–43, 2022.
- [16] H. Zhao, J. Cao, M. Xu and J. Lu, "Variational neural decoder for abstractive text summarization," *Computer Science and Information Systems*, vol. 17, no. 2, pp. 537–552, 2020.
- [17] B. Muthu, S. Cb, P. M. Kumar, S. N. Kadry, C. H. Hsu *et al.*, "A framework for extractive text summarization based on deep learning modified neural network classifier," *Transactions on Asian and Low-Resource Language Information Processing*, vol. 20, no. 3, pp. 1–20, 2021.
- [18] F. A. Zeidabadi, S. A. Doumari, M. Dehghani and O. P. Malik, "MLBO: Mixed leader based optimizer for solving optimization problems," *International Journal of Intelligent Engineering and Systems*, vol. 14, no. 4, pp. 472–479, 2021.
- [19] R. Hang, Q. Liu, D. Hong and P. Ghamisi, "Cascaded recurrent neural networks for hyperspectral image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 8, pp. 5384–5394, 2019.
- [20] Q. V. Pham, S. Mirjalili, N. Kumar, M. Alazab and W. J. Hwang, "Whale optimization algorithm with applications to resource allocation in wireless networks," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 4, pp. 4285–4297, 2020.