DOI: 10.6977/IJoSI.202510_9(5).0001
S. Upadhyay & H. K. Soni/Int. J. Systematic Innovation, 9(5), 1-13 (2025)

An ensemble learning framework for text summarization based on an improved multilayer extreme learning machine autoencoder

Sunil Upadhyay*, Hemant Kumar Soni

Department of Computer Science and Engineering, Amity School of Engineering and Technology, Amity University Madhya Pradesh, Gwalior, Madhya Pradesh, India

*Corresponding author E-mail: supadhyay@gwa.amity.edu

(Received 05 October 2024; Final version received 07 July 2025; Accepted 05 August 2025)

Abstract

The massive growth of electronic data has created a demand for efficient tools to manage information and support fast decision-making. Automatic text summarization (ATS) addresses this by condensing large texts into concise, relevant summaries rapidly. ATS methods are categorized as extractive, abstractive, or hybrid. Extractive techniques select key sentences from input documents, whereas abstractive techniques generate new sentences to capture meaning. Hybrid methods combine both strategies. However, despite numerous suggested techniques, machine-generated summaries often fail to match the accuracy and coherence of human-written summaries. This study reviewed existing ATS techniques and highlighted their limitations, particularly high computational costs and low training efficiency. To address these problems, this study proposed an improved multilayer extreme learning machine autoencoder (MLELM–AE) and an ensemble learning framework that integrates four machine learning models: Sentence-bidirectional encoder representations from transformers, autoencoder, variational autoencoder, and the improved MLELM–AE. The proposed framework generates summaries through cosine similarity evaluation, followed by voting-based fusion, re-ranking, and optimal sentence selection. Experimental results showed that the proposed improved MLELM–AE model achieved strong performance, with an execution time of 50,015 ms and a recall-oriented understudy for gisting evaluation 1 score of 0.865145. These findings demonstrate that the proposed ensemble framework delivers more accurate and efficient summaries, offering a promising advancement in ATS.

Keywords: Automatic Text Summarization, Bidirectional Encoder Representations from Transformers, Deep Neural Networks, Multilayer Extreme Learning Machine Autoencoder, Word Embedding, Word2vec

1. Introduction

In today's era, the Internet has huge amounts of data due to the rapid expansion of web-based electronic documents. The proliferation of this vast volume of data makes it complicated to collect pertinent information efficiently. In view of the huge amount of text documents, gathering and processing primary data from various resources is a complex and exhaustive task, often exceeding human capacity. This challenge has motivated researchers to develop techniques for automatic text summarization (ATS), which aim to condense large volumes of text into concise summaries while preserving meaning and context. Over the past several decades, several information retrieval techniques have been explored to address this problem.

Text summarization is a rapidly growing and challenging task in natural language processing (NLP). It aims to produce a condensed version of a document that retains the key ideas of the original text, facilitating the comprehension of these ideas (Mitra et al., 2000). ATS is particularly valuable because manual text summarization is tedious and time-consuming. In the NLP domain, summarization also serves as an intermediary step to reduce text size and complexity. Key application areas of text summarization include text classification, question and answer, legal document summarization, social media text summarization, and headline/title creation.

Text summarization can be categorized by output type into two main approaches (Gambhir & Gupta,

2017): Extractive text summarization and abstractive text summarization. Extractive text summarization is the most widespread approach to text summarization. It extracts important textual units, such as phrases, words, and sentences, based on linguistic and mathematical features to form a summary. On the other hand, abstractive text summarization generates summaries closer to human-written summaries by creating semantic representations and producing new sentences, rather than merely reordering existing ones. Summaries generated by these methods are generally grammatically correct. Therefore, these techniques are not limited to simply picking and reordering sentences from the original text.

Summarization can also be classified by input type (single vs. multi-document) or purpose (generic, domain-specific, or query-based) (Zajic et al., 2008). Generic summarization captures broad themes and addresses a wide community of users; domain-specific summarization incorporates knowledge of specialized fields, such as law or biomedicine, while query-based summarization tailors the output to user needs.

Despite progress, ATS still faces significant challenges. Key issues include encoding high-level semantic structures, handling large input dimensionality, managing out-of-vocabulary words, and ensuring accurate part-of-speech tagging. Conventional machine learning approaches often struggle with these challenges due to their shallow architecture and restricted capability for hierarchical feature learning. Neural network-based methods have improved semantic modeling, but they still face limitations, including computational inefficiency, noisy training data, and the omission of important sentences due to score-based selection.

To overcome these limitations, this study proposed an improved novel ensemble learning-based ATS, called improved multilayer extreme learning machine–autoencoder (MLELM–AE). The multilayer architecture enhances the ability to learn deep and abstract features, improving the identification of salient information. In addition, the proposed algorithm incorporates end-to-end training using backpropagation, allowing iterative refinement of hidden layers and better generalization compared to conventional extreme learning machine (ELM)-based approaches. The AE framework ensures efficient reconstruction, enabling efficient dimensionality reduction while retaining important information for producing high-quality summaries.

The proposed ensemble framework integrates multiple models, including the improved MLELM–AE, AE, variational AE (VAE), and sentence-bidirectional encoder representations from transformers (SBERT). It employs data transformation steps, such as clustering, topic modeling, term frequency–inverse document frequency (TF–IDF) analysis, and frequent term selection to enhance text representation. Entity-focused sentences are captured through topic modeling, while a re-ranking mechanism ensures optimal sentence selection for the final summary. Overall, the proposed ensemble approach significantly advances ATS by combining semantic entity extraction, robust feature learning, and effective sentence re-evaluation.

The remainder of this paper is structured as follows: Section 2 reviews existing works, Section 3 details the proposed methodology, Section 4 presents results and discussion, and Section 5 concludes with future scopes.

2. Related Works

The work by Toprak & Turan (2025) demonstrated an automatic abstractive document summarization framework based on transformers and sentence grouping. The collected dataset was pre-processed and then utilized to train the transformer model. Then, the transformer model proficiently summarized the text. This approach obtained a SimHash text similarity of 93.2%, indicating a high effectiveness and low complexity. However, this model suffered from considerable information loss.

Khan et al. (2025) implemented a hybrid deep learning-based next-generation text summarization for psychological data. Text-to-text transfer transformer (T5) and long short-term memory (LSTM) were employed to perform advanced text summarization. This approach achieved an accuracy, precision, and recall of 74%, 72%, and 72%, respectively, indicating its supremacy. However, the framework had high computational complexity owing to the hybrid scheme.

Alotaibi & Nadeem (2025) introduced an Arabic aspect-based sentiment analysis and abstractive text summarization of traffic services using an unsupervised-centric approach. A fine-tuned AraBART algorithm was employed to perform abstractive text summarization. This algorithm achieved 92.13% precision and 92.07% recall, indicating its high efficacy. However, the model struggled to handle the text from various domains.

Onan & Alhumyani (2024a) propounded an extractive text summarization framework using fuzzy topic modeling and bidirectional encoder representations from transformers (BERT). Here, fuzzy logic was used to improve topic modeling, thereby capturing a nuanced representation of word-topic relationships. This algorithm obtained recall-oriented understudy for gisting evaluation 1 (ROUGE-1) and ROUGE-2 scores of 45.3774 and 24.1808, respectively. It significantly provided high-quality text summaries. However, the framework was ineffective due to the lack of interpretability.

In the work by Onan & Alhumyani (2024b), they implemented a multi-element contextual hypergraph extractive summarizer (MCHES) to perform extractive text summarization. MCHES effectively constructed a contextual hypergraph, showing semantic and discourse hyperedges. The approach achieved an ROUGE-1 score of 44.321 and an ROUGE-2 score of 19.129, indicating its impressive performance in extractive summarization. However, the framework had a maximum risk of bias amplification.

Hassan et al. (2024) demonstrated an approach of extractive text summarization using NLP with an optimal deep learning (ETS-NLPODL) model. The research analysis of various parameters indicated that the ETS-NLPODL approach achieved excellent performance compared to other models regarding diverse performance measures.

Hernández-Castañeda et al. (2023) designed a fitness function based on genetic programming to generate ATS. The experimental outcomes clearly showed that the grouping of lexical and semantic information (LDA+Doc2Vec+TF–IDF) achieved exceptional outcomes in identifying key ideas to form a summary.

Dilawari et al. (2023) proposed a model for both extractive and abstractive summarizations, named as automatic feature-rich model architecture comprises a hierarchical bidirectional LSTM. The results demonstrated that the model outperformed existing techniques, with a ROUGE score of 37.76%, high generality, and high sapiential.

An improved English text summary algorithm based on association semantic rules was proposed in a previous study (Wan, 2018). The method mined relative features among English sentences and phrases, implemented keyword extraction in English abstracts, and applied semantic relevance analysis with association rules distinction, grounded in knowledge theory. Semantic rules were further mined from English teaching texts. The outcome of the replication showed that the technique could accurately extract summaries with improved convergence and output accuracy. This demonstrates strong application value for efficiently reading English texts and gathering important information.

Zenkert et al. (2018) proposed the multidimensional knowledge representation structure. The fallouts of analytics using individual methods for text mining, such as named person recognition, sentiment analysis, and topic detection, were integrated into a knowledge base as dimensions to support knowledge exploration, vision, and computer-aided written tasks. This framework supports cross-dimensional exploration and provides a novel approach for summarization and knowledge discovery.

Similarly, Prameswari et al. (2018) combined sentiment analysis and summary generation, applying their method to hotel reviews in Bali and Labuan Bajo. Their model achieved a rating accuracy of 78% with a Davies–Bouldin index of 0.071, demonstrating potential benefits for the Indonesian tourism industry.

Jain et al. (2017) proposed a neural network-based extractive summarization function, testing on the Document Understanding Conferences (DUC) 2002 dataset. Their approach outperformed four online summarizers in ROUGE evaluations, indicating the importance of robust feature extraction for summary generation. The scale and complexity of training datasets and additional exact methods to convert abstract summaries into extractive summaries will further improve the model.

In clustering-based approaches, Pradip & Patil (2016) developed a hierarchical sentence clustering algorithm to address instability, complexity, and sensitivity issues in traditional methods. Any type of relational clustering algorithm may work with an implemented hierarchical clustering algorithm. The general text mining algorithm can also be used. Experimental results demonstrate that hierarchical clustering was useful and yielded improved results for text documents.

Akter et al. (2017) presented a text summarization method that extracts significant phrases from single or multiple Bengali documents, which were prepared by processes, such as tokenization or interrupt operations. The word score was then determined using the TF–IDF weighting, and the sentence value was calculated with location. For sentence score calculation, the term skeleton and cue were also considered. K-means clustering was used to summarize many or a single document in a final form. Their method reduced redundancy and improved run-time complexity compared to existing extractive approaches.

Jadhav et al. (2019) designed a bidirectional recurrent neural network (RNN)-based encoder-decoder model that identifies key phrases and generates coherent summaries. Initially, key phrases were listed and arranged in a consolidated report. Given the measurable and semantic highlights of sentences, the sense of the sentence was chosen. This shorter representation was then passed through an encoder-decoder template to produce a description of the entire document. The projected model efficiently created a concise and linguistically accurate synthesis by recognizing the content and disclosing it in its terms. The proposed methodology only selected related terms and passed them to a bidirectional RNN to define the central ideas of the article and to represent them.

The ATS problem consists of two main tasks: Single-document and multi-document summarization.

In the case of a single document, input and summarized details are extracted from a specific document, whereas for multiple documents, summaries are generated based on a shared theme. A recent statistical approach was proposed by Madhuri and Kumar (2019) to perform extractive text summarization on single documents. The method of extracting sentences was presented, providing a brief overview of the input text. Phrases were categorized by weight assignment. Highly ranked phrases were then selected to form the final summary, which can also be converted into audio output.

Document review aims to condense the source text into a short and succinct form while preserving accuracy and general significance. Dave & Jaswal (2015) proposed an abstractive summary approach that generates compact and human-readable summaries using WordNet ontology derived from extractive summaries. The generated summaries were grammatically correct and more coherent for human readers.

Elbarougy et al. (2020) introduced an Arabic text summarization method, a graphical system with text expressed on its vertices. An improved PageRank algorithm was applied with initial node scores and multiple iterations to generate optimal summaries while eliminating redundancies. Using the Essex Arabic Summaries Corpus for evaluation, this method outperformed TextRank and LexRank, achieving a final F-measure of 67.98, which surpassed earlier approaches.

Collecting textual information is a challenging activity in biomedical text synthesis. Moradi et al. (2020) proposed a method leveraging BERT-based contextual embeddings to capture the semantic information of biomedical texts. Their deep learning model clustered sentences using BERT and selected the most relevant ones for summary generation. Evaluation with the ROUGE toolkit demonstrated significant improvements in biomedical text synthesis, outperforming other domain-independent approaches.

A multi-target optimization method has contributed to ATS over the years. Sanchez-Gomez et al. (2019) applied a multi-objective artificial bee colony (MOABC) algorithm, incorporating parallelization strategies. Comparative experiments on DUC datasets showed that their asynchronous structure significantly enhanced performance, achieving over 55 times quicker with 64 threads and an efficiency of 86.72%, outperforming traditional synchronous methods.

Qaroush et al. (2021) proposed automated and extractive general Arabic single-document summarizing techniques to construct comprehensive summary details. The proposed extractive methods used statistical and semantic features to evaluate sentence value, diversity, and exposure. Two summarizing techniques were also used to construct a description and then exploited built characteristics, such as score and machine learning supervision. Performance of the proposed technique was tested using the ROUGE metrics, yielding superior results in terms of accuracy, retrieval, and F-score compared to related works.Present graph-based extractive summarization methods represent corpus sentences as nodes, with edges depicting lexical similarity between sentences (Van Lierde & Chow, 2019). However, such approaches cannot adequately capture semantic similarities, since sentences may convey related information using different words. To address this, Van Lierde & Chow (2019) proposed extracting semantical similarities based on topical representations. They introduced a topic model to infer the distribution of hierarchical, context-influenced sentences. Since each concept establishes semantic relationships across sentences by assigning degrees of membership, the authors further proposed a fluid hypergraph model, where nodes represent sentences and fuzzy hyperedges. Sentence collections were then extracted to produce comprehensive summaries while simultaneously optimizing user-defined query relevance, centrality within the hypergraph, and topic coverage. To solve this optimization problem, they developed an algorithm based on submodular function theory. A thorough comparison with other graphic summarizers demonstrated the superiority of their strategy in the coverage of summaries.

Extractive multifocal approaches aim to synthesize key material while reducing redundancy. One promising avenue is multi-objective optimization, which naturally fits the summarization problem (Sanchez-Gomez et al., 2019). This method produces a set of non-dominated solutions or Pareto sequences, though ultimately only one summary is selected. To address this, post-Pareto analyses were performed using various methods, including hypervolume maximization, minimum distance from all points, minimum distance from an ideal point, and a consensus solution. Experiments conducted on DUC datasets and evaluated using ROUGE metrics revealed that the consensus approach outperformed others, improving ROUGE scores by 10.68–27.32%.

In another study, Alami et al. (2019) enhanced ATS efficiency using unregulated deep neural networks combined with a word embedding approach. First, they built a word definition on word integration and demonstrated that the representation of Word2Vec was better than that of traditional bag-of-words (BOW). Second, by combining Word2Vec and unmonitored functional learning approaches, they offered alternative models for incorporating information from various sources. They revealed that uncontrolled neural network models trained on the representation

of Word2Vec were enhanced compared to those trained on BOW models. Third, they described three ensembles: (i) Majority voting between Word2Vec and BOW, (ii) aggregation of BOW with unsupervised neural network outputs, and (iii) a combined ensemble of Word2Vec and unattended neural networks. Results showed that ensemble techniques enhanced ATS performance, with Word2Vec-based ensembles consistently outperforming BOW-based models. Comparative evaluations across two publicly accessible datasets confirmed that Word2Vec ensemble methods yielded the best results, surpassing all studied models in effectiveness.

Abstractive text summarization is a more challenging task than extractive summarization, as it requires generating paraphrased text that conveys the entire meaning of the source. Nonetheless, it typically yields more natural summaries with improved cohesion between sentences. Adelia et al. (2019) demonstrated that RNNs can effectively produce abstractive summaries in both English and Chinese. In their study, a bidirectional gated recurrent unit RNN architecture was used to capture the effect of surrounding words on generated summaries. Applying a similar method to Bahasa Indonesia, they showed that the model could generate summaries closely resembling human-written abstracts, outperforming purely extractive approaches. Their findings suggest that RNN-based abstractive models can achieve strong comprehension of source texts to support high-quality summary generation.

Building on this line of work, Yao et al. (2018) proposed a dual-encoder sequence-to-sequence attentional model for abstractive summarization. Unlike previous research that relied on a single encoder, their model incorporated both a primary encoder, which performed coarse-grained encoding, and a secondary encoder, which provided fine-grained encoding based on raw input and previously generated outputs. By combining both levels, the model reduced redundancy and improved handling of long sequences. The test outcomes of two complicated datasets (DUC 2004 and CNN Daily Mail) revealed that their hybrid model of encoding outperformed existing methods.

Du & Huo (2020) focused on fuzzy logic rules, multi-feature analysis, and genetic algorithms to develop a new automated synthesis paradigm for news texts. Since news articles often contain distinctive elements, such as time, place, and characters, word features were first extracted, and those surpassing a threshold score were identified as keywords. A linear combination of these characteristics revealed the meaning of each sentence, and each feature evaluated the genetic algorithms. Using fuzzy logic, the system generated automated summaries. The simulation results on the DUC 2002 dataset, evaluated with the ROUGE tool, demonstrated that the proposed method outperformed several baseline approaches, including Microsoft Word, System19, System2, System30, single-document summarization–neural network with a genetic algorithm, general context decoder, self-organizing map, and support vector machine ranking.

Alzuhair & Al-Dhelaan (2019) proposed combining multiple graph-based methods to enhance the quality of extractive summary outcomes. Given the widespread use of graph-based techniques in NLP, they developed a hybrid approach that integrates two graph-based techniques (four different weighting methods and two graph methods). To merge the results, both the arithmetic mean and harmonic mean were tested. Experiments conducted on the DUC 2003 and DUC 2004 datasets, evaluated using the ROUGE toolkit, and revealed that the harmonic mean outperformed the arithmetic mean. Furthermore, the hybrid method demonstrated significant improvements over baseline models and several state-of-the-art approaches when combined with weighting schemes.

Building on sequence-to-sequence frameworks, Ding et al. (2020) sought to optimize traditional sequence mapping and semantic representation for abstractive summarization. Their proposed method enhanced the model's semantic comprehension of source texts and improved the coherence of generated summaries. The method was validated on two benchmark datasets, large-scale Chinese short text summarization (LCSTS) and SOGOU datasets, where experimental results showed ROUGE score improvements of 10–13% compared to existing algorithms. These findings demonstrate that optimizing semantic representation can substantially enhance both the accuracy and readability of generated summaries.

Similarly, Liang et al. (2020) introduced an abstractive summarization model tailored for social media texts using a selective sequence-to-sequence (i.e., Seq2Seq) framework. To improve content filtering, a discerning gate was added after the encoder to eliminate irrelevant or redundant information. In addition, they combined inter-entropy with enhancement learning to directly optimize ROUGE scores. Evaluations on the LCSTS dataset demonstrated that their model achieved F1-score gains of 2.6% for ROUGE-1, 2.1% for ROUGE-2, and 2.0% for ROUGE-L compared with the baseline Seq2seq model.

El-Kassas et al. (2020) introduced EdgeSumm, a novel extractive graph-based architecture designed to optimize ATS for single documents. The framework relies on four proposed algorithms, with the first constructing a novel text graph model (NTGM) from the input document. The second and third algorithms identify candidate sentences from the constructed text graph, while the fourth finalizes the summary selection. Unlike many existing methods, EdgeSumm

is domain-independent and unsupervised, requiring no training data. The model was evaluated on the standard DUC 2001 and DUC 2002 datasets using the ROUGE evaluation toolkit. Results showed that EdgeSumm achieved the highest ROUGE scores on DUC 2001, and on DUC 2002, it outperformed several state-of-the-art ATS frameworks by margins of 1.2–4.7% in ROUGE-1 and ROUGE-L. The proposed framework also delivered highly competitive performance on ROUGE-2 and ROUGE-SU4, confirming its robustness and efficiency.

Automatic review summarization has emerged as an effective approach to improving information processing for travelers. However, many review texts contain vague or non-sentimental content, limiting the effectiveness of sentiment-based methods. To address this, Tsai et al. (2020) proposed a systematic framework that first identifies useful reviews through a classifier and then categorizes sentences into six hotel-related features. Subsequently, the polarity of each sentence is evaluated for analytical summaries. Experimental results demonstrated that the proposed method outperformed other methods, producing more accurate and informative summaries of hotel reviews.

Joshi et al. (2019) proposed SummCoder, a novel extractive method for single-document summarization. This framework is based on three sentence-level analysis techniques: Sentence position, content relevance, and sentence novelty. Content relevance is computed using a deep AE network, while novelty is measured through semantic similarity between sentence embeddings in distributed space. Sentence position is modeled using a hand-designed weighting function that assigns higher significance to earlier sentences, with adjustments based on document length. Final summaries are generated by ranking sentences according to a fused score from these three metrics. To support evaluation, the authors introduced the Tor Illegal Documents Summarization (TIDSumm) dataset, specifically built to assist law enforcement agencies in analyzing web documents from the Tor network. Empirical outcomes showed that SummCoder performed on par with or better than, several state-of-the-art approaches across various ROUGE metrics on DUC 2002, blog summarization datasets, and TIDSumm.

Jindal & Kaur (2020) developed an unsupervised approach to summarizing bug reports, aiming to capture both overall content and specific software-related details. Their method begins with automated keyword extraction using TF–IDF, followed by ranking of key sentences. To reduce redundancy, fluid C-means clustering is applied with thresholding, and a rule motor informed by domain knowledge selects the most relevant sentences. Additional hierarchical clustering is employed for re-ranking and improving coherence. The proposed approach was evaluated on the Apache bug report corpus (APBRC) and bug report corpus (BRC) using metrics, such as precision, recall, pyramid precision, and F-score. Experimental results showed substantial improvements over baseline methods, including BRC and logistic regression with crowdsourcing attributes, as well as existing unsupervised methods, such as Hurried and Centroid. The APBRC evaluation reported 78.22% precision, 82.18% recall, 80.10% F-score, and 81.66 pyramid precision, highlighting the method's strong performance in generating cohesive and comprehensive summaries.

3. Methodology

3.1. Improved MLELM–AE

The improved MLELM–AE is a hybrid neural network model that integrates the fast training ability of ELMs with the deep feature learning capability of AEs. Conventional ELMs typically employ only a single hidden layer and compute output weights analytically, which enables extremely fast training but restricts their ability to capture complex patterns. To address these issues, the proposed improved MLELM–AE introduces a multilayer architecture structure as a deep AE. This design enables the network to learn hierarchical and abstract depictions of input data.

This approach is particularly designed for tasks, such as ATS and dimensionality reduction, where capturing deep semantic features is important. The algorithm begins by defining the network architecture, including the input layer size, output layer size (typically matching the input in AEs), and the configuration of hidden layers. Bias vectors and weight matrices are set randomly for every hidden layer. During the forward pass, input data are propagated through each hidden layer using a non-linear activation function, enabling the model to capture complex patterns and relationships within the data. The output layer then attempts to reconstruct the original input, consistent with the fundamental nature of an AE.

In contrast to traditional ELMs that depend exclusively on closed-form solutions to compute output weights, the proposed model adopts an iterative optimization approach. For a pre-defined number of iterations, the model computes the reconstruction error (the difference between the input and the reconstructed output) and updates the weights using a specified learning rate. This hybrid approach preserves the computational efficiency of ELMs in the hidden layers while enabling the model to adaptively fine-tune the output layer weights. Compared to traditional ELM or single-layer AEs, the proposed model demonstrates improved convergence.

The innovation of the improved MLELM–AE stems from its integration of the fast learning capacity of ELMs with the deep feature extraction strength of multilayer AE. This design leverages fixed random weights in the hidden layers while allowing adaptive updates in the output layer, thereby enabling deep feature extraction at a minimal computational cost. By employing reconstruction loss as the training objective, the model is particularly well-suited for unsupervised learning applications. Compared with shallow architectures, it demonstrates superior ability to capture complex data representations, providing an efficient balance among performance, training speed, and architectural simplicity. The flow of data between hidden layers is mathematically formulated in Eq. (1):

$$H_i \quad g(\ _i)\ H_i \tag{1}$$

where β_i is the output weights, T is equivalent to the input data X at the first layer of MLELM, β_{i+1} is the output weight matrix of the i^{th} hidden layer, and $i+1^{th}$ layer weights are the outputs of MLELM. Regularized least squares were used for output layer weight calculation of MLELM.

The proposed improved MLELM–AE algorithm introduces numerous key novelties over conventional models, such as ELMs and AE. The main contributions are outlined in Table 1.

3.2. Algorithm of the Improved MLELM–AE

Input:
- Training data: TRx
- Number of iterations: niterations
- Learning rate: lrate

Output:
- Improved MLELM–AE trained model: A

a. Initialization

1. Describe input dimensions:
   - input_size, hsizes, osize (sizes of input, hidden layers, and output, respectively)
2. Initialize weights and biases for each layer:
   - For each layer k:
     - G[k] = random matrix of size (hsizes[k], input_size if k == 0 else hsizes[k-1])
     - h[k] = random matrix of size (hsizes[k], 1)
3. Initialize output weights and biases:
   - G_out = random matrix of size (osize, hsizes[-1])
   - h_out = random matrix of size (osize, 1)

b. Train the network

For each iteration in range niterations:

1. Forward pass:
   - Initialize activations = [input_data]
   - For each layer k:
     - Compute:
       Y = activation_function(G[k] * Y + h[k])
     - Append Y to activations
   - Compute final output:
     output = G_out * Y + h_out
2. Calculate loss:
   - Compute loss:
     loss = mean((output - activations[0])^2)
3. Backward pass:

i. Compute output error and delta:
- oerror = output - activations[0]
- odelta = oerror

ii. Update output weights and biases:
- G_out -= lrate * (odelta * activations[-1].T)
- h_out -= lrate * mean(odelta, axis=1, keepdims=True)

iii. Compute hidden layer errors:
- Initialize herrors = [odelta]
- For each layer k in reverse:
  - hidden_error = G[k+1].T * herrors[-1]
  - hdelta = hidden_error * activations[k+1] * (1 - activations[k+1])
  - Append hdelta to hdeltas and hidden_error to herrors

iv. Update weights and biases for hidden layers:
- For each layer k:
  - G[k] -= lrate * (hdeltas[k] * activations[k].T)
  - h[k] -= lrate * mean(hdeltas[k], axis=1, keepdims=True)

c. Return the trained model
- Return the trained model A (Improved MLELM–AE)

3.3. Ensemble Learning Framework for Text Summarization

In the proposed ensemble learning framework, the enhancement of sentence representations and the improvement of output summaries' quality are achieved using an ensemble of deep learning models: The improved MLELM–AE, SBERT, AE, and VAE (Fig. 1). From the output of these models, cosine similarity scores are computed, followed by a voting-based fusion strategy, re-ranking, and optimal sentence selection.

In this approach, Word2Vec and SBERT semantic embedding models are first used to convert the input document into dense vector representations, effectively capturing the contextual relationships within the text. These embeddings are then passed through four parallel encoding modules: SBERT, AE, VAE, and the improved MLELM–AE. Each encoder

Table 1. Novelty of the proposed improved MLELM–AE algorithm

Feature	Traditional ELM	AE	Improved MLELM–AE
Hidden layers	Single	Multiple	Multiple
Training	Non-iterative (closed form)	Backpropagation	ELM with backpropagation
Speed	Fast	Moderate	Fast and adaptive
Output update	Only output layer	All layers	Output and hidden layers
Loss function	Classification loss	Reconstruction loss	Reconstruction loss (MSE)
Adaptability	Low	High	High
Learning	Randomized and no tuning	Gradient-based	Hybrid: Random initialization and gradient tuning

Abbreviations: AE: Autoencoder; ELM: Extreme learning machine; MLELM: Multilayer ELM; MSE: Mean squared error.

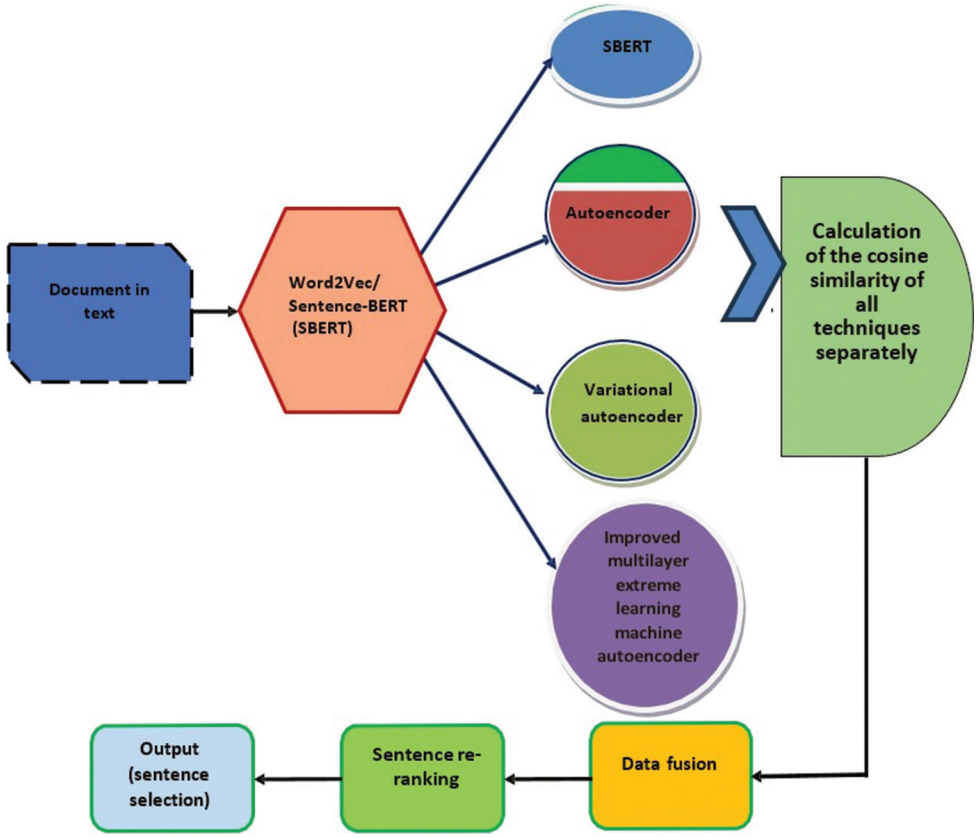

Fig. 1. Ensemble learning framework for text summarization

extracts sentence-level features independently, focusing on different aspects of sentence semantics and information compression. SBERT retains rich contextual information and deep contextual features, while AE and VAE reduce dimensionality and gather latent semantic structures. The improved MLELM–AE leverages the computational efficiency of extreme learning alongside the representational strength of deep learning models to enhance feature abstraction. Once sentence-level embeddings are formed, cosine similarity is computed separately for each model to assess sentence significance. The similarity scores are then integrated using a data fusion method, allowing the integration of diverse model perspectives. Based on the fused scores, sentences are re-ranked to prioritize informative and non-redundant content. Finally, the highest-ranked sentences are selected to form the extractive summary. This ensemble-based framework improves summarization effectiveness, semantic quality, and robustness by integrating the diverse capabilities of various encoding techniques.

4. Results and Discussion

4.1. Software Requirements

The proposed framework was implemented in PYTHON, a widely used general-purpose and high-level programming language that is primarily developed for emphasizing code readability. The

syntax of PYTHON permits developers to define concepts in fewer lines of code. Similarly, PYTHON effectively incorporates the system and works faster. PYTHON is used in numerous applications, including artificial intelligence, scientific computing, and automation. In addition, for many common tasks, the comprehensive standard library of PYTHON provides modules and functions.

4.2. Hardware Requirements

The hardware necessities for the proposed model and framework are as follows:

- Processor: Intel Core i5/i7
- Central processing unit speed: 3.20 GHz
- Operating system: Windows 10
- System type: 64-bit
- RAM: 4 GB

4.3. Dataset Description

The proposed improved MLELM–AE model was evaluated using the DUC 2002 dataset, which is publicly available (https://ieee-dataport.org/documents/sentence-embeddings-document-sets-duc-2002-summarization-task). The DUC 2002 dataset consists of 1,358 text documents. For experimentation, the dataset was divided into training, validation, and testing subsets. Specifically, 70% of the documents (950) were used for training, 10% (135) for validation, and the remaining 20% (271) for testing. The detailed hyperparameters employed in the proposed framework are presented in Table 2.

Table 2. Detailed hyperparameters of the models

Specifications	Proposed improved MELM-AE	AE	VAE	SBERT
Epoch	500	500	500	500
Activation function	ReLU	ReLU	ReLU	ReLU
Weight initialization	Hyperfan-In	Xavier	Xavier	Xavier
Learning rate	0.0001	0.008	0.017	0.124
Batch size	100	80	60	20
Optimizer	Adam	Adam	Adam	Adam
Loss function	MSE	MSE	MSE	MSE
Dropout rate%	0.2	0.5	0.4	0.3

Abbreviations: AE: Autoencoder; MLELM: Multilayer extreme learning machine; MSE: Mean squared error; ReLU: Rectified linear unit; SBERT: Sentence bidirectional encoder representations from transformers; VAE: Variational autoencoder.

4.4. Performance Evaluation of the Proposed Improved MLELM–AE model

The performance of the proposed improved MLELM–AE model was compared with existing techniques, including AE, SBERT, and VAE, to demonstrate its reliability. The evaluation was conducted using standard metrics, such as accuracy, precision, recall, F-measure, sensitivity, and ROUGE-1 score. The proposed improved MLELM–AE achieved superior results, with accuracy, precision, recall, F-measure, sensitivity, and ROUGE-1 scores of 96.32%, 97.16%, 96.01%, 97.24%, 97.01%, and 0.865145, respectively. In contrast, the existing techniques attained comparatively lower average performance across these metrics, as summarized in Table 3. These results confirm that the proposed improved MLELM–AE model significantly outperforms the baseline models in extractive text summarization.

Specifically, the highest accuracy of 96.32% was achieved by the proposed improved MLELM–AE model, significantly outperforming AE (91.41%), SBERT (90.87%), and VAE (91.48%), thereby confirming its robust ability to correctly identify relevant instances. In terms of precision (97.16%) and F-measure (97.24%), the proposed model exhibited exceptional performance, indicating its ability to generate highly accurate summaries or predictions with minimal false positives and a robust balance between precision and recall. Similarly, the high recall score (96.01%) highlights its effectiveness in capturing the majority of relevant outputs, ensuring comprehensive coverage of the target content. In contrast, AE, SBERT, and VAE recorded lower recall values of 91.03%, 91.11%, and 92.49%, respectively, highlighting their limitations in capturing all relevant elements.

The proposed improved MLELM–AE model also attained an outstanding ROUGE-1 score of 0.865145, a significant measure in text summarization that assesses unigram overlap between system-generated and reference summaries. This outperformed AE (0.819125), SBERT (0.805981), and VAE (0.816013), confirming that the summaries produced by the proposed model are more semantically and lexically aligned with human-authored summaries.

Moreover, the execution time of the improved MLELM–AE (50,015 ms) was shorter than that of AE (56,236 ms), SBERT (61,008 ms), and VAE (63,018 ms), demonstrating efficiency without compromising performance (Fig. 2). Finally, the model achieved a low error rate (0.010766), reflecting its accuracy in fitting training data; nonetheless, further assessment on unseen datasets is required to meticulously validate its generalization capability.

Overall, the experimental results indicate that the proposed improved MLELM–AE model not only

Table 3. Comparative assessment of the models

Model	Accuracy %	Precision %	Recall %	F-Measure %	ROUGE-1	Time (ms)	Error
Proposed improved MLELM–AE	96.32	97.16	96.01	97.24	0.905145	50,015	0.010766
AE	91.41	93.21	91.03	93.12	0.819125	56,236	0.031064
SBERT	90.87	92.47	91.11	93.52	0.805981	61,008	0.066596
VAE	91.48	93.52	92.49	94.01	0.816013	63,018	0.092872

Abbreviations: AE: Autoencoder; MLELM: Multilayer extreme learning machine; ROUGE-1: Recall-oriented understudy for gisting evaluation 1; SBERT: Sentence bidirectional encoder representations from transformers; VAE: Variational autoencoder.

Table 4. Comparative analysis with previously described frameworks

References	Techniques	ROUGE-1 score
Proposed ensemble framework in the present study	AE, SBERT, VAE, and improved MLELM–AE	0.865145
Hernández-Castañeda et al. (2022)	GA	0.414000
Hernández-Castañeda et al. (2020)	GA, LDA, and TF–IDF	0.486810

Abbreviations: AE: Autoencoder; GA: Genetic algorithm; LDA: Latent Dirichlet allocation; MLELM: Multilayer extreme learning machine; ROUGE-1: Recall-Oriented Understudy for Gisting Evaluation 1; TF–IDF: Term frequency–inverse document frequency.

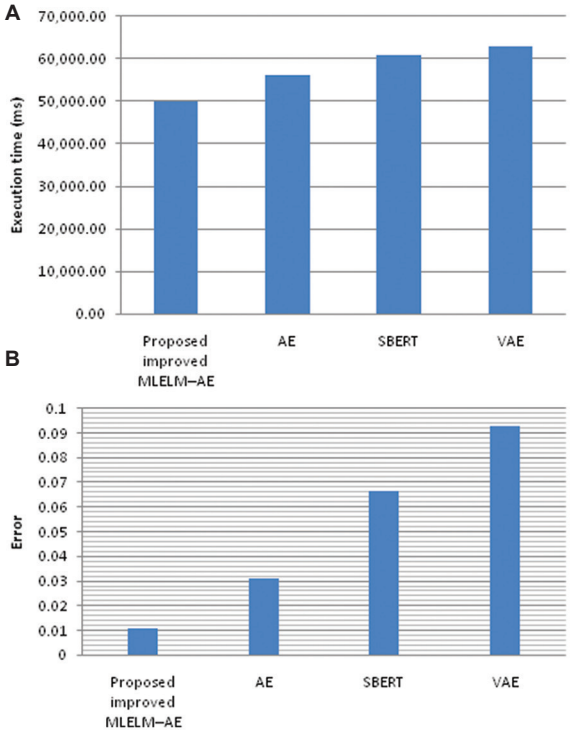

Fig. 2. Execution time (A) and error (B) of the models Abbreviations: AE: Autoencoder; MLELM: Multilayer extreme learning machine; SBERT: Sentence bidirectional encoder representations from transformers; VAE: Variational autoencoder

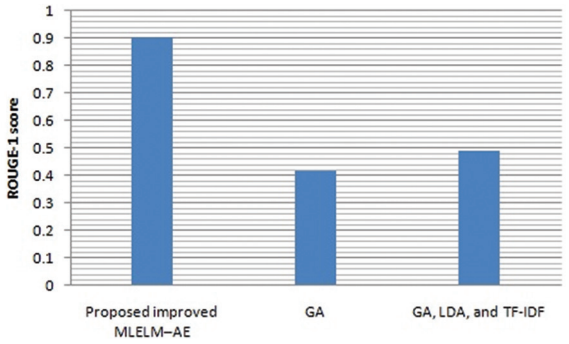

Fig. 3. Comparative analysis of the proposed ensemble framework and previously described models Abbreviations: AE: Autoencoder; GA: Genetic algorithm; LDA: Latent Dirichlet allocation; MLELM: Multilayer extreme learning machine; ROUGE-1: Recall-Oriented Understudy for Gisting Evaluation 1; TF–IDF: Term frequency–inverse document frequency

attains state-of-the-art accuracy and performance metrics but also offers computational proficiency, making it a promising approach for real-world applications in text summarization and related NLP tasks.

4.5. Comparative Analysis of the Proposed Ensemble Framework

A comparative analysis of the proposed ensemble framework and previously described frameworks (Hernández-Castañeda et al., 2020; Hernández-Castañeda et al., 2022) was conducted to further validate the model's reliability. The results are summarized in Table 4 and Fig. 3. The proposed ensemble framework achieved a notably high ROUGE-1 score of 0.865145, primarily due to the incorporation of the improved MLELM–AE model. In contrast, the existing genetic algorithm approach achieved a considerably lower ROUGE-1 score of 0.414 on the same DUC 2002 dataset. Similarly, the

model based on a genetic algorithm, latent Dirichlet allocation, and TF–IDF techniques attained a lower ROUGE-1 score of 0.48681, which can be attributed to their computational complexity. These findings clearly demonstrate that the proposed ensemble framework outperforms traditional approaches in performing ATS.

5. Conclusion

ATS is a widely explored research area in the NLP community, as it enables the generation of concise and informative summaries from large volumes of text. This paper presents an improved ensemble learning-based ATS framework that incorporates the AE, SBERT, VAE, and improved MLELM–AE. The DUC 2002 dataset was employed for training and evaluation. The research methodology involves several steps, including pre-processing, slang identification and filtering, part-of-speech tagging, entity extraction, vectorization, ensemble modeling, similarity evaluation, re-ranking, and optimal sentence selection. Experimental results demonstrate that the proposed improved MLELM–AE achieved high accuracy (96.32%), precision (97.16%), and recall (96.01%). On the other hand, the proposed ensemble framework achieved a high ROUGE-1 score of 0.865145, significantly outperforming existing models. These findings clearly validate the effectiveness of the proposed approaches in delivering improved ATS performance.

Acknowledgments

None.

Funding

None.

Conflict of Interest

The authors declare that they have no competing interests.

Author Contributions

Conceptualization: Sunil Upadhyay
Investigation: Sunil Upadhyay
Writing–original draft: Sunil Upadhyay
Writing–review and editing: All authors

Availability of Data

Data sharing is not applicable to this article as no datasets were generated or analyzed during the present study.

References

Adelia, R., Suyanto, S., & Wisesty, U.N. (2019). Indonesian abstractive text summarization using bidirectional gated recurrent unit. *Procedia Computer Science*, 157, 581–588. https://doi.org/10.1016/j.procs.2019.09.017

Akter, S., Asa, A.S., Uddin, M.P., Hossain, M.D., Roy, S.K., & Afjal, M.I. (2017). An Extractive Text Summarization Technique for Bengali Document(s) using K-Means Clustering Algorithm. In: *Proceedings of the 2017 IEEE International Conference on Imaging, Vision and Pattern Recognition (icIVPR)*. IEEE, p1–6. https://doi.org/10.1109/ICIVPR.2017.7890883

Alami, N., Meknassi, M., & En-nahnahi, N. (2019). Enhancing unsupervised neural networks based text summarization with word embedding and ensemble learning. *Expert Systems with Applications*, 123, 195–211. https://doi.org/10.1016/j.eswa.2019.01.037

Alotaibi, A., & Nadeem, F. (2025). An unsupervised integrated framework for Arabic aspect-based sentiment analysis and abstractive text summarization of traffic services using transformer models. *Smart Cities*, 8(2), 62. https://doi.org/10.3390/smartcities8020062

Alzuhair, A., & Al-Dhelaan, M. (2019). An approach for combining multiple weighting schemes and ranking methods in graph-based multi-document summarization. *IEEE Access*, 7, 120375–120386. https://doi.org/10.1109/access.2019.2936832

Dave, H., & Jaswal, S. (2015). Multiple Text document summarization system using hybrid summarization technique. In: *Proceedings of the 2015 1st International Conference on Next Generation Computing Technologies (NGCT)*, IEEE, p804–808. https://doi.org/10.1109/ngct.2015.7375231

Dilawari, A., Khan, M.U.G., Saleem, S., Zahoor-Ur-Rehman, & Shaikh, F.S. (2023). Neural attention model for abstractive text summarization using linguistic feature space. *IEEE Access*, 11, 23557–23564. https://doi.org/10.1109/access.2023.3249783

Ding, J., Li, Y., Ni, H., & Yang, Z. (2020). Generative text summary based on enhanced semantic attention and gain-benefit gate. *IEEE Access*, 8, 92659–92668. https://doi.org/10.1109/access.2020.2994092

Du, Y., & Huo, H. (2020). News text summarization based on multi-feature and fuzzy logic. *IEEE Access*, 8, 140261–140272. https://doi.org/10.1109/access.2020.3007763

Elbarougy, R., Behery, G., & El Khatib, A. (2020). Extractive Arabic text summarization using modified PageRank algorithm. *Egyptian*

Informatics Journal, 21(2), 73–81. https://doi.org/10.1016/j.eij.2019.11.001

El-Kassas, W.S., Salama, C.R., Rafea, A.A., & Mohamed, H.K. (2020). EdgeSumm: Graph-based framework for automatic text summarization. *Information Processing and Management*, 57(6), 102264. https://doi.org/10.1016/j.ipm.2020.102264

Gambhir, M., & Gupta, V. (2017). Recent automatic text summarization techniques: A survey. *Artificial Intelligence Review*, 47(1), 1–66. https://doi.org/10.1007/s10462-016-9475-9

Hassan, A.Q.A., Al-Onazi, B.B., Maashi, M., Darem, A.A., Abunadi, I., & Mahmud, A. (2024). Enhancing extractive text summarization using natural language processing with an optimal deep learning model. *AIMS Mathematics*, 9(5), 12588–12609. https://doi.org/10.3934/math.2024616

Hernández-Castañeda, Á., García-Hernández, R.A., & Ledeneva, Y. (2023). Toward the automatic generation of an objective function for extractive text summarization. *IEEE Access*, 11, 51455–51464. https://doi.org/10.1109/access.2023.3279101

Hernández-Castañeda, Á., García-Hernández, R.A., Ledeneva, Y., & Millán-Hernández, C.E. (2020). Extractive automatic text summarization based on lexical-semantic keywords. *IEEE Access*, 8, 49896–49907. https://doi.org/10.1109/access.2020.2980226

Hernández-Castañeda, Á., García-Hernández, R.A., Ledeneva, Y., & Millán-Hernández, C.E. (2022). Language-independent extractive automatic text summarization based on automatic keyword extraction. *Computer Speech and Language*, 71, 101256. https://doi.org/10.1016/j.csl.2021.101267

Jadhav, A., Jain, R., Fernandes, S., & Shaikh, S. (2019). Text Summarization using Neural Networks. In *2019 6th IEEE International Conference on Advances in Computing, Communication and Control (ICAC3)*. https://doi.org/10.1109/icac347590.2019.9036739

Jain, A., Bhatia, D., & Thakur, M.K. (2017). Extractive text summarization using word vector embedding. In: *2017 International Conference on Machine Learning and Data Science (MLDS)*, p51–55. https://doi.org/10.1109/mlds.2017.12

Jindal, S.G., & Kaur, A. (2020). Automatic keyword and sentence-based text summarization for software bug reports. *IEEE Access,* 8, 65352–65370. https://doi.org/10.1109/access.2020.2985222

Joshi, A., Fidalgo, E., Alegre, E., & Fernández-Robles, L. (2019). SummCoder: An unsupervised framework for extractive text summarization based on deep auto-encoders. *Expert Systems with Applications*, 129, 200–215. https://doi.org/10.1016/j.eswa.2019.03.045

Khan, B., Usman, M., Khan, I., Khan, J., Hussain, D., & Gu, Y.H. (2025). Next-generation text summarization: A T5-LSTM FusionNet hybrid approach for psychological data. *IEEE Access*, 13, 1–15.

Liang, Z., Du, J., & Li, C. (2020). Abstractive social media text summarization using selective reinforced Seq2Seq attention model. *Neurocomputing*, 410, 432–440. https://doi.org/10.1016/j.neucom.2020.04.137

Madhuri, J.N., & Kumar, R.G. (2019). Extractive Text Summarization using Sentence Ranking. In: *Proceedings of the 2019 International Conference on Data Science and Communication (IconDSC)*, p1–3. https://doi.org/10.1109/icondsc.2019.8817040

Mitra, M., Buckley, C., & Research, S. (2000). *Automatic Text Summarization by Paragraph Extraction*. Available from: https://www.researchgate.net/publication/2457789 [Last accessed on 2025 Sep 09].

Moradi, M., Dorffner, G., & Samwald, M. (2020). Deep contextualized embeddings for quantifying the informative content in biomedical text summarization. *Computer Methods and Programs in Biomedicine*, 184, 105117. https://doi.org/10.1016/j.cmpb.2019.105117

Onan, A., & Alhumyani, H. (2024b). Contextual hypergraph networks for enhanced extractive summarization: Introducing multi-element contextual hypergraph extractive summarizer (MCHES). *Applied Sciences*, 14(11), 4671. https://doi.org/10.3390/app14114671

Onan, A., & Alhumyani, H.A. (2024a). FuzzyTP-BERT: Enhancing extractive text summarization with fuzzy topic modeling and transformer networks. *Journal of King Saud University - Computer and Information Sciences*, 36(6), 102080. https://doi.org/10.1016/j.jksuci.2024.102080

Pradip, K.G., & Patil, D.R. (2016). Summarization of Sentences using Fuzzy and Hierarchical Clustering Approach. In: *2016 Symposium on Colossal Data Analysis and Networking (CDAN)*. IEEE. p1–7. https://doi.org/10.1109/cdan.2016.7570907

Prameswari, P., Zulkarnain, Z, Surjandari, I., & Laoh, E. (2018). *Mining Online Reviews in Indonesia's Priority Tourist Destinations using Sentiment Analysis and Text Summarization Approach*. In: 2017 IEEE 8th International Conference on Awareness Science and Technology (iCAST). IEEE, p36–41.

https://doi.org/10.1109/icawst.2017.8256540

Qaroush, A., Abu Farha, I., Ghanem, W., Washaha, M., & Maali, E. (2021). An efficient single document Arabic text summarization using a combination of statistical and semantic features. *Journal of King Saud University - Computer and Information Sciences*, 33(6), 677–692. https://doi.org/10.1016/j.jksuci.2019.03.010

Sanchez-Gomez, J.M., Vega-Rodríguez, M.A., & Pérez, C.J. (2019). Parallelizing a multi-objective optimization approach for extractive multi-document text summarization. *Journal of Parallel and Distributed Computing*, 134, 166–179. https://doi.org/10.1016/j.jpdc.2019.09.001

Toprak, A., & Turan, M. (2025). Enhanced automatic abstractive document summarization using transformers and sentence grouping. *The Journal of Supercomputing*, 81(4), 1–30. https://doi.org/10.1007/s11227-025-07048-6

Tsai, C.F., Chen, K., Hu, Y.H., & Chen, W.K. (2020). Improving text summarization of online hotel reviews with review helpfulness and sentiment. *Tourism Management*, 80, 104122. https://doi.org/10.1016/j.tourman.2020.104122

Van Lierde, H., & Chow, T.W.S. (2019). Learning with fuzzy hypergraphs: A topical approach to query-oriented text summarization. *Information Sciences*, 496, 212–224. https://doi.org/10.1016/j.ins.2019.05.020

Wan, L. (2018). Extraction Algorithm of English Text Summarization for English Teaching. In: *2018 3rd International Conference on Intelligent Transportation, Big Data and Smart City (ICITBS)*. https://doi.org/10.1109/icitbs.2018.00085

Yao, K., Zhang, L., Du, D., Luo, T., Tao, L., & Wu, Y. (2018). Dual encoding for abstractive text summarization. *IEEE Transactions on Cybernetics*. IEEE, New York. https://doi.org/10.1109/tcyb.2018.2876317

Zajic, D.M., Dorr, B.J., & Lin, J. (2008). Single-document and multi-document summarization techniques for email threads using sentence compression. *Information Processing and Management*, 44(4), 1600–1610. https://doi.org/10.1016/j.ipm.2007.09.007

Zenkert, J., Klahold, A., & Fathi, M. (2018). Towards Extractive Text Summarization using Multidimensional Knowledge Representation. In: *2018 IEEE International Conference on Electro/ Information Technology (EIT)*, p826–831. https://doi.org/10.1109/EIT.2018.8500186

AUTHOR BIOGRAPHIES

Mr. Sunil Upadhyay received his M.TECH. degree from the Department of Computer Science and Engineering at RGEC Meerut, Gautam Buddha Technical University, formerly known as UPTU, Lucknow, U.P., India, in 2012. He is pursuing his Ph.D. degree in Computer Science and Engineering from Amity University, Madhya Pradesh, Gwalior, India. His presentsf research interest includes NLP, data analytics, artificial intelligence, and machine learning.

Prof. (Dr.) Hemant Kumar Soni completed his M. Tech. (IT) at Bundelkhand University, Jhansi, Uttar Pradesh, India, and a Doctoral degree in Computer Science and Engineering from Amity University, Madhya Pradesh, Gwalior, India. He has 26 years of teaching experience for undergraduate and post-graduate courses in Computer Science, and is presently working as a Professor in the Department of Computer Science and Engineering at Amity University Madhya Pradesh, Gwalior, India. His research interests are in natural language processing, data science, machine learning, and data mining. He published many research papers in Web of Science, SCI, and Scopus-indexed journals. His research articles received high levels of citations in Scopus and Google Scholar. He is a Reviewer of many referred journals, including *IEEE Transactions*.