THE SCIENTIFIC TEMPER (2026) Vol. 17 (6): 6478-6489

E-ISSN: 2231-6396, ISSN: 0976-8653

Doi: 10.58414/SCIENTIFICTEMPER.2026.17.6.17

https://scientifictemper.com/

RESEARCH ARTICLE

A Probabilistic Correlative Convolutional Extreme Learning Model for Semantic Classification of Social Media Documents

Sharada C[1*], T N Ravi[2], S Panneer Arokiaraj[3]

Abstract

Text document classification is a fundamental task in Natural Language Processing (NLP) that involves dispersing predefined categories or labels to given text. The rapid expansion of online text data, increased by social media platforms, has highlighted the critical role of text classification in managing large-scale data. Different document classification techniques are carried out for efficient semantic analysis but the time efficiency faced major issues. To address these issues, Probabilistic Bivariate Correlative Convolutional Extreme Learning Machine (PBiC-CELM) developed for efficient document classification with higher accuracy and minimum time. First, numbers of text documents are collected from dataset. These text documents are given as input to Convolutional Extreme Deep Belief Network (CELM). The text preprocessing is carried out in convolutional layer where tokenization, Stop Word Removal and Word stemming are performed. The pre-processed text is transmitted to hidden layer 2. In that layer, Least Angle Regression Analysis is carried out for efficient keyword extraction. Then, the extracted keywords are transmitted to ELM for document classification. In that layer, Bivariate Correlation Coefficient is employed for classifying the document. In this way, an accurate semantic document classification is obtained with higher accuracy and minimal time consumption. Result of PBiC-CELM method achieves efficient performance outcomes, including higher 93% of accuracy, 96% of precision, 95% of recall, 95% of F1-score, and 94% of specificity with minimized classification time by 313 ms and error rate by 1.5% when compared to existing methods and state-of-art-methods.

Keywords: Semantic text document classification, extreme learning machine, Convolutional Neural Networks (CNNs), Least Angle Regression results, Bivariate Correlation Coefficient.

[1]Research Scholar, Department of Computer Science, Thanthai Periyar Government Arts and Science College, Affiliated to Bharathidasan University, Tiruchirappalli, Tamil Nadu, India

[2]Associate Professor, Department of Computer Science, Jamal Mohamed College, Affiliated to Bharathidasan University, Tiruchirappalli, Tamil Nadu, India

[3]Associate Professor, Department of Computer Science, Thanthai Periyar Government Arts and Science College, Affiliated to Bharathidasan University, Tiruchirappalli, Tamil Nadu, India

***Corresponding Author:** Sharada C, Research Scholar, Department of Computer Science, Thanthai Periyar Government Arts and Science College, Affiliated to Bharathidasan University, Tiruchirappalli, Tamil Nadu, India, E-Mail: csharada@gmail.com

How to cite this article: Sharada, C., Ravi, T.N., Arokiaraj, S.P. (2026). A Probabilistic Correlative Convolutional Extreme Learning Model for Semantic Classification of Social Media Documents. The Scientific Temper, **17**(6):6478-6489.

Doi: 10.58414/SCIENTIFICTEMPER.2026.17.6.17

Source of support: Nil

Conflict of interest: None.

Introduction

Processing and analyzing text documents are fundamental in natural language processing (NLP), such as document classification, information retrieval, and summarization. A Three-Way Decision Enhanced Graph Convolutional Networks (3WD-GCN) handles text document classification (Jiang et al. 2025). Here, computational complexity was not reduced. BERT integrated with Multi Layered Convolutional Neural Network (B-MLCNN) [2] classifies document (Atandoh et al. 2023). However, the semantic analysis was not implemented. An automated approach known as (LDA) support identification and analysis of text classification designed (Qiu et al. 2024). However, it was unable to enhance recall. Quantum Selfattention Neural Network (QSANN) carries text classification (Li et al. et al. 2024). But, complexity of algorithm was not reduced. Multi-modal Dynamic Knowledge Graph with Reinforcement Learning for Archival Retrieval (MDKG-RL) model optimizes document management and retrieval (Chen et al. 2025). However, model was not suitable for more complex real-

Received: 10/06/2026 **Accepted:** 13/06/2026 **Published:** 25/06/2026

world scenarios. A Gravitational Network with Enhanced Dependency Semantics (GNEDS) model enhances text's semantic classification (Zuo et al. 2025). However, the precision and reliability of keyword extraction was not reduced. An ontology-based sentiment analysis with BERT method attains accurate text classification ((Taye et al. 2025). However, the error rate was not reduced. A pre-trained Bidirectional Long Short-Term Memory (BiLSTM) performs document-level sentiment classification (Xu et al. 2025). But the model did not succeed in improving the model's computational efficiency. A new self-supervised sentiment analysis method provides semantic similarity and contextual embedding (Alizadeh & Seilsepour 2025). Learning based Text Classification (LbTC) model provides text document classification (Avinash et al. 2025). However, it failed to enhance prediction accuracy. An integration of BERT and graph convolution models presents multimodal text classification (Shao et al. 2025). The designed model was not efficient to enhance classification accuracy. A multi-head graph attention network model portrays accurate text classification (Lv et al. 2024). However, robustness, stability, and effectiveness of model were major challenged. A contrastive multi-graph learning model was designed based on neighbor hierarchical sifting for semi-supervised text classification (Ai et al. 2025). However, the performance of error rate was not effectively reduced. Mask guided BERT framework was introduced to enhance the robustness of text classification through the task related features (Liao et al. 2024). The model was not efficient to reduce the time complexity in the text classification. Adaptive Feature Interactive Enhancement Network (AFIENet) model achieves text classification (Su et al. 2025). However, richer types of feature extraction networks were not explored.

Objectives of the study

- To improve accuracy of semantic text document retrieval, a novel PBiC-CELM has been developed. By integrating CNNs and ELMs in a parallel and bi-directional framework, PBiC-CELM is capable of capturing both deep semantic representations and contextual dependencies within documents, thereby significantly enhancing performance.
- To reduce time required for keyword extraction, Least Angle Regression (LARS) is applied within the pooling layer of the Convolutional Neural Network (CNN).
- To reduce overall classification time, the PBiC-CELM incorporates a text preprocessing stage along with the extraction of highly relevant keywords within the CNN layers.
- To enhance accuracy of semantic document classification, PBiC-CELM employs an Extreme Learning Machine (ELM) to analyze the relationship between keywords and class labels using the bivariate correlation coefficient.
- To compare PBiC-CELM approach against existing techniques, multiple performance metrics utilized to measure the efficiency.

Related works

Advanced Machine Learning algorithms were developed for E text classification across diverse domains and datasets (Ahamad & Mishra 2025). However, efficient sentiment analysis systems were not addressed. A symmetrical multi-task transfer learning framework explains clinical text document classification (Zhang et al. 2025). The model did not reduce classification error. A K-nearest neighbor algorithm and intelligent classification model improve efficiency to characterize and extract text information (Xu et al. 2025). However, feature extraction was not improved. Syntactic-aware text classification method carries feature weight vector (Wang et al. 2025). Though, performance of multilingual text classification was major issues. A comprehensive recommendation system describes news sentiment classification based on multimodel fusion (Xi & Jiang 2025). But it did not including the large-scale deployment. A dual-channel Semantic enhancement-based Convolutional Neural Network model was introduced for text classification (Zhang et al. 2025). The model was not simpler and more effective. Adaptive Search Mechanism Based Hierarchical Learning Networks was introduced for social media text data classification (Al-Anazi et al. 2025). However, it did integrating more diverse data sources. A knowledge-enhanced approach was developed for improving performance of unsupervised text analysis by using latent-factor Bayesian generative model (Costa & Ortale 2025). The model failed to develop more effective AI systems for accurate text analysis. A Multi-Features Maximal Marginal Relevance BERT (MFMMR-BertSum) model handled A novel Deep Learning (DL) -based approach includes pre-processing for better result (Rautaray et al. 2025). However, feature selection was major issues. Genetic optimized fully connected CNNs designs social media text classification (Ghanem et al. 2024). But computational complexity was not minimized. A new Label Knowledge-guided Heterogeneous Graph Contrastive Learning (LKG-HGCL) model was designed for short text sentiment classification (Wu 2025). However, advanced techniques were not employed to influence the linguistic features of labels or their semantic correlations with text content. Large language models were developed efficient text classification (Chausson et al. 2026). However the computational cost of the model was high.

Proposal methodology

In this paper, a novel system utilizes proposed methodology to improve accuracy of document classification.

Figure 1 demonstrates architecture of proposed method for document classification. The proposed method comprises text acquisition, preprocessing, keyword

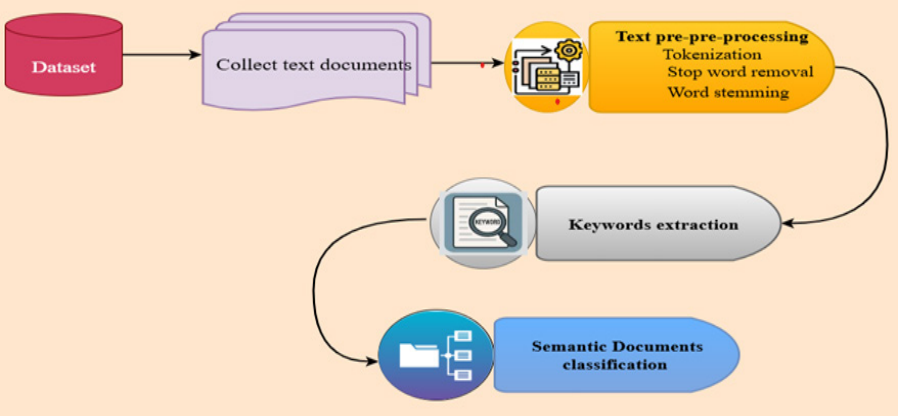

Figure 1: Architecture of proposed PBiC-CELM method

extraction, semantic data classification. Initially, numbers of text documents are gathered in text acquisition from the dataset. After, Query preprocessing plays a vital role for performing the tokenization, Stop Word Removal and Word stemming to improve the accuracy. Then, the keyword extraction extracts the relevant keywords from the pre-processed text to determine most informative features. The main aim of semantic data classification is to categorize the documents with the extracted keywords. This section describes the proposed PBiC-CELM method in detail, focusing on its ability to improve the accuracy of data classification with minimal time complexity.

Text document acquisition

In the Text document acquisition refers to process of collecting, measuring, and digitizing Text document from various sources for purpose of analysis, storage, or real-time processing. The PBiC-CELM method uses text document classification dataset (Baxter 2020). This dataset includes a collection of 1000 newsgroup documents from 10 different newsgroups. The 10 Newsgroups dataset has gained popularity as a benchmark for testing machine learning techniques in text-based tasks, including text classification and clustering.

Convolutional extreme learning machine (CELM) based classification

A CELM is a hybrid neural network model that integrates feature extraction capabilities of CNNs with fast learning and accurate classification ability of ELMs. Compared to other learning mode, CELMs is more suitable for both accuracy and efficiency in text classification.

Figure 2 shows schematic structure of CELM classifier designed for accurate text document classification with minimal time. These documents transferred through a convolutional layer, which performs the preprocessing, followed by a pooling layer that reduces dimensionality while extracting the more relevant features. The resulting feature maps are then passed to the Extreme Learning Machine. Finally, output layer classifies documents into

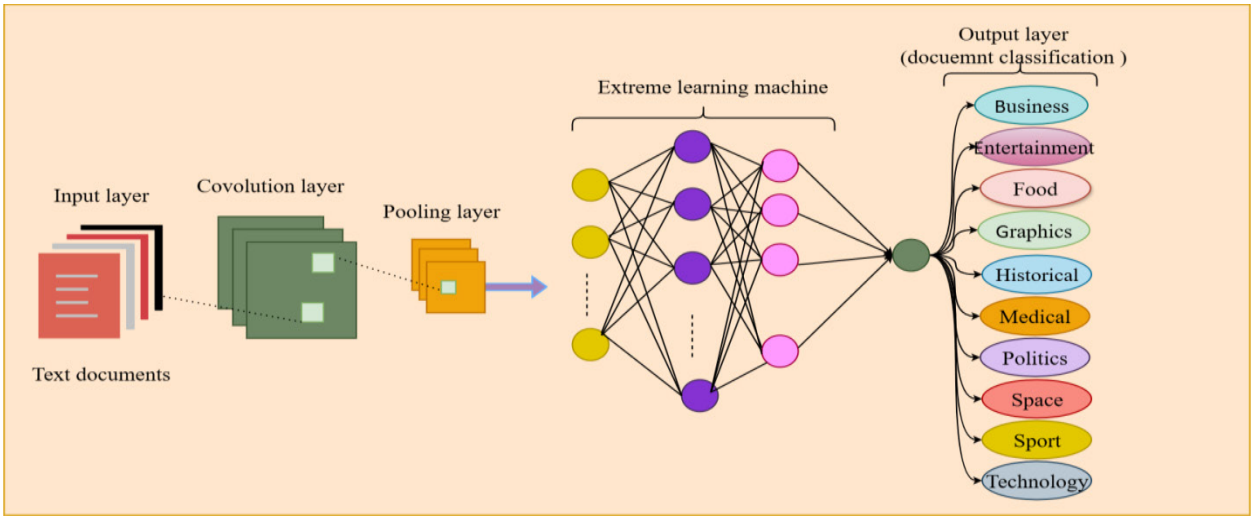

Figure 2: Schematic construction of Convolutional extreme learning machine algorithm

predefined categories such as Business, Politics, Medical, Sport, and so on. This hybrid approach utilizes the powerful feature extraction capabilities of CNNs and high efficiency of ELMs, making it suitable for large-scale and real-time text classification tasks.

The CELM network architecture considers training set $\{TD_i, Y_j\}$ where TD_i indicates a training data samples or text documents '$TD_i = \{TD_1, TD_2, \ldots, TD_n\}$' and Y_j refers to a final classification outcomes. Each neuron in input layer receives training data samples and forwards them to hidden layer where hyper parameters such as weights and biases are randomly assigned..

$$Z = f\left[\sum_{i=1}^{n} (TD_i * W_{in-hi}\right) + b] \quad (1)$$

Where, Z represents a weighted sum output, TD_i represents an input i.e. text documents, W_{in-hi} indicates a weight between neuron 'i' in input layer and neuron j inother consecutive layer, and b represents the bias, f represents activation function,

Text preprocessing

In first hidden layer, text preprocessing involves organizing raw data samples for further analysis. It clean and transform data into a suitable format for providing accurate and significant results. Preprocessing helps to enhance quality and efficiency of dataset and minimizes the likelihood of errors.

Gensim tokenizer

The Gensim tokenizer is used to process input text documents by splitting them into individual tokens or words. Gensim's tokenizer operates by identifying word boundaries based on specific delimiters, such as spaces, punctuation marks, and symbols, which are defined within square brackets. The resulting tokens provide a clean and meaningful representation of text, which then be used for tasks feature extraction, and classification. Gensim's lightweight is more suitable for handling large volumes of textual data in document classification. Let us consider 'n' text documents '$TD_i = \{TD_1, TD_2, \ldots, TD_n\}$ and each text contains 'm' number of words '$w_1, w_2, w_3, \ldots w_m$'. Then tokenization process is expressed below,

$$TD \overset{GT}{\Rightarrow} \left['w_1', \ 'w_2', \ 'w_3', \ldots 'w_m'\right] \quad (2)$$

Where, the query text documents 'TD' is partitioned into a number of words or tokens $w_1, w_2, w_3, \ldots w_m$ using Gensim tokenizer 'GT'.

Stop Word Removal

Stop word removal is a text preprocessing step that involves eliminating frequently occurring words that carry no meaningful information for analysis. These words are known as stop words. Removing the stop words helps to reduce data size, improves processing efficiency, and enhances model performance by focusing on more relevant terms. The probability of stop word is determined through the Welch's t-test method to analyze words against a predefined list of stop words. Stop words are then detected and eliminated based on probability results. Let us consider the text document 'TD' includes a 'm' number of words or tokens $w_1, w_2, w_3, \ldots w_m$ and predefined stop word list 'LS_w'. Therefore, the probability of statistical 'P' analysis is computed as given below,

$$P = \frac{\left| w_{TD} - w_{LS_w} \right|}{\triangle s} \quad (3)$$

$$P = \begin{cases} 1, & Stop\ words \\ 0, & Otherwise \end{cases} \quad (4)$$

Where w_{TD} indicates word count in text documents, w_{LS_w} represents words in the predefined lists, $\triangle s$ denotes a deviation between the words. The probability outcome provides the output values between zero and one.

Snowball Stemmer

The Snowball Stemmer operates on these tokens after stop word removal to identify their stem or root forms.

$$w_{original} = w_{root} + w_{stem} \quad (5)$$

Where, $w_{original}$ denotes an original tokens or words after stop word removal, w_{root} represents the root word, w_{stem} denotes a stem words. The algorithm removes the suffix from the original word $w_{original}$, leaving only the root w_{root}. This root represents base form of word, which helps in grouping different variations of same word together.

$$w_{root} = w_{original} - w_{stem} \quad (6)$$

The algorithm removed the suffix or stem w_{stem} to get the root word 'w_{root}'. In this way, text preprocessing is done in convolutional layer. The preprocessed text is transferred into pooling layer for further processing.

Feature extraction

In text processing using CELM, feature extraction in pooling layer plays a vital role in summarizing and refining the information captured from the input text documents. The pooling layer reduces dimensionality of text documents through the features extraction. It selectively retains the most important features typically by methods like max pooling, which selects the strongest words from the texts. It enhances feature extraction process by highlighting significant features and improving the model's ability to understand and classify text documents effectively. The proposed method utilizes Least Angle Regression Analysis aims to determine the keywords from preprocessed text documents. Consider number of words $w_1, w_2, w_3, \ldots w_k$ as

input to Least Angle Regression algorithm. Initializes score value with zero for each feature.

$$Sim_b = 0 \quad \forall \; w_b = w_1, w_2, w_3, \ldots . w_k \tag{7}$$

After that, compute correlations of all words with documents. For each word, compute its association with target output and words with documents are computed using Braun-Blanquet similarity function as follows,

$$Sim(w_b, w_l) = \frac{|w_b \cap w_l|}{\boldsymbol{max}(w_b, w_l)} \tag{8}$$

Where, $Sim(w_b, w_l)$ denotes asimilarity between the word 'w_b' and word in document label, $max(w_b, w_l)$ representsthe maximum size either the word w_b and w_l. The similarity score ranges from 0 to 1.Based on similarity output, the output score of each feature is increased from 0. The highly correlated features are determined by setting the threshold as follows,

$$H = \begin{cases} Sim(w_b, w_l) > T, & highly\ correlated\ features \\ Otherwise, & Less\ correlated \end{cases} \tag{9}$$

Where, H denotes an output of determining the highly correlated features, T represents the threshold, $Sim(w_b, w_l)$ denotes a similarity between the word 'w_b' and word in document label. After that, computing the residual for each keyword based on difference between output of actual keywords and predicted keywords through the correlation score value.

$$R = \left(H_{actual} - H_{predicted}\right) \tag{10}$$

Where, R denotes a residual for each keyword, H_{actual} denotes an output of output of actual keywords, $H_{predicted}$ represents an output of predicted keywords. Followed by, the angle between residual and keyword 'w_k' is computed as follows,

$$\theta = \cos^{-1}\left[\frac{R^T . w_k}{|R||w_k|}\right] \tag{11}$$

Where, θ denotes an angle, R^T denotes a transpose of residual function, w_k represents the keywords. When the angle 'θ' is small, it indicates that keywords closely matched the residual vector's direction, it indicates that strong keywords are extracted. As a result, the least angle principle directs regression in building accurate text document classification with minimal time consumption.

Classification

The PBiC-CELM method is a text document classification that involves categorizing documents into predefined classes by analyzing features with semantic relationship. By utilizing meaningful features, classification model is accurately assign documents to appropriate class. The quality of feature extraction significantly influences performance of classification. The BCC measures strength and direction of linear relationship between two variables such as selected keywords and particular class. This coefficient is widely used in statistics and data analysis to identify and interpret relationships between pairs of continuous variables.

$$BCC = \frac{\left(f_w - Af_w\right)\left(Y_c - \overline{Y_c}\right)}{\sqrt{\left(f_w - Af_w\right)^2}\sqrt{\left(\left(Y_c - \overline{Y_c}\right)\right)^2}} \tag{12}$$

Where, f_w denotes a frequency of a particular keyword in document, Af_w denotes an average frequency of that keyword across all documents, Y_c denotes a class label for document, $\overline{Y_c}$ mean of class labels.The most commonly used measure is Bivariate Correlation Coefficient, which ranges from -1 to +1. A value close to +1 indicates a strong positive linear relationship. A value close to 0 means no clear relationship. From BCC output, each keyword is matched with specific class labels, allowing documents to be accurately classified based on presence of keywords. This correlation-based method enhances accuracy of text classification. The final classification results are obtained at output layer with softmax activation functions.

$$Y = f_{SM}\left(h_t * W_{in-hi}\right) \tag{13}$$

$$f_{SM} = \frac{\exp\left(Y_k\right)}{\sum_{c=1}^{k}\exp\left(Y_c\right)} \tag{14}$$

Where, Y denotes an output of final classification results, softmax activation function '$f_{f_{SM}}$' at the output layer is used to construct a multiple classification outcomes, Y_k indicates an output for c^{th} class, k denotes a total number of classes used document classification. In this way, accurate document classification results are obtained at output layer. The algorithmic description of the document classification method is given below.

Experimental setup

The proposed PBiC-CELM method along with existing methods 3WD-GCN (Jiang et al. 2025), BERT-MultiLayered Convolutional Neural Network (B-MLCNN) (Atandoh et al. 2023), Latent Dirichlet allocation (LDA) (Qiu et al. 2024) and two state-of-the-art methods, BERT and LSTM are implemented using Python for efficient document classification.

Implementation results

PBiC-CELM method is comprehensively observed to evaluate its effectiveness in analyzing text document classification. First, the numbers of text documents are collected from the dataset. This dataset includes 1000 text documents with 10 various distinct newsgroups document. Once text documents is collected, the dataset undergoes

preprocessing to ensure it is clean, and organized for effective text documents analysis. The sample outcomes of preprocessing stage are illustrated in Figure 3.

Figure 4 describes the stop word removal. Following the tokenization, the PBiC-CELM method performs the stop word removal using the Least Angle Regression Analysis to identify and retain the most relevant words in text documents while discarding less significant ones.

Figure 5 illustrates Snowball Stemming output. After removing stop words, word stemming is applied to reduce each remaining word to its root form by using Snowball Stemmer. Finally, the sample analysis of preprocessed text documents is observed in figure 6.

The outcome of keyword extraction is shown in figure 7. The PBiC-CELM applies a dimensionality reduction technique using the Least Angle Regression Analysis to identify and extract the most efficient keywords features while discarding others after preprocessing stage.

Sample classification outcomes of text classification are shown in figure 8. In the final stage, the PBiC-CELM performs text document classification using bivariate correlation coefficient within ELM through extracted features. This strategy improves efficiency and accuracy.

Experimentation Results

The PBiC-CELM method along with existing methods 3WD-GCN (Jiang et al. 2025), BERT-MultiLayered Convolutional Neural Network (B-MLCNN) (Atandoh et al. 2023), Latent Dirichlet allocation (LDA) (Qiu et al. 2024) and BERT and LSTM, for classifying the text documents.

5.1 Evaluation metrics

Error rate 'ER': it measures proportion of incorrect predictions made by classification model.

$$ER = \sum_{i=1}^{M} \frac{IM_{IAC}}{IM_i} * 100 \qquad (15)$$

Where, the input data samples employed 'IM_i' and input matrix inaccurately classified 'IM_{IAC}'. It is measured in terms of percentage (%).

Keyword extraction time 'ET': It is measured as the amount

Figure 3: text tokenization

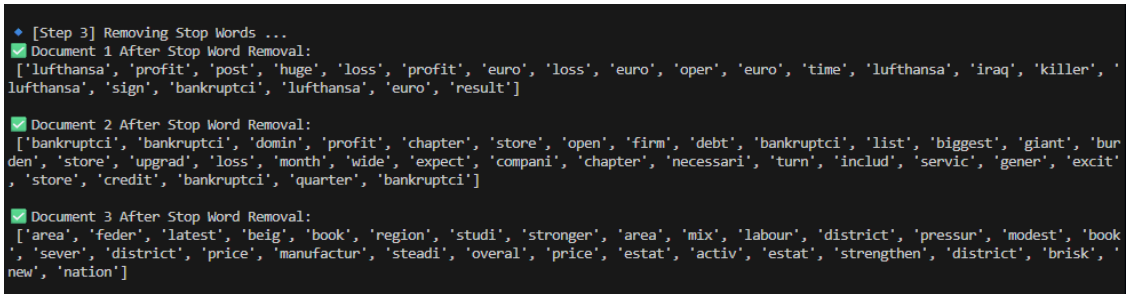

Figure 4: Stop word removal

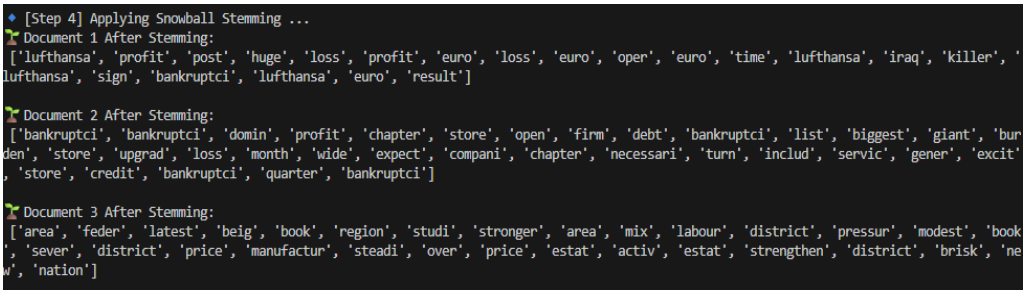

Figure 5: Snowball Stemming output

// **Algorithm 1: P**robabilistic **Bi**variate **C**orrelative **C**onvolutional **E**xtreme **L**earning **M**achine classifier

Input: Dataset' DS ', News groups ' $NG = \{NG_1, NG_2, \ldots, NG_N\}$, text documents ' $TD_i = \{TD_1, TD_2, \ldots, TD_n\}$

Output: text document classification

Begin

Step 1: Collect number of text documents ' $TD_i = \{TD_1, TD_2, \ldots, TD_n\}$ '- **input layer**

Step 2: For each documents TD_i
Step 3: Compute the weighted sum by the neuron using (1)
Step 4: End For

Step 5: For each documents TD_i - **convolutional layer**
Step 6: Partition the text into words using Gensim tokenizer using (2)

Step 7: For each word ' $w_j^{'}$ in text

Step 6: **For each word** ' $w_j^{'}$ in predefined stop word list
Step 7: Measure the probability using Welch's t-test using (3)

Step 8: ***if*** $(P = 1)$ **then**
Step 9: identified as stop word and removed
Step 10:else
Step 11: identified as normal word
Step 12: End if
Step 13: Apply snowball stemmer to remove word
Step 14: End for
Step 15: End for
Step 16: End for
Step 17: For each Pre-processed words **do**
Step 18: Regression analysis and it initializes the score value with zero

Step 19: Measure the similarity $Sim(w_b, w_l)$ using (8)

Step 20: if ($Sim(w_b, w_l) > T$) **then**
Step 21: Identify the correlated features
Step 22: End if

Step 23: Compute angle ' θ ' using (11)

Step 24: if arg min θ then
Step 25: keywords extracted
Step 26: end if
Step 27: End for

Step 28: For each keywords ' w_k ----- **ELM hidden layers**

Step 29: **For each** class labels ' Y_c
Step 30: Measure the bivariate correlation using (12)

🎯 Preprocessing Complete! Summary:

	Title	Class	Stemmed
0	business_1.txt	business	[lufthansa, profit, post, huge, loss, profit, ...
1	business_10.txt	business	[bankruptci, bankruptci, domin, profit, chapte...
2	business_100.txt	business	[area, feder, latest, beig, book, region, stud...

Figure 6: Preprocessed text documents

Figure 7: keyword extraction

🔎 Sample predictions (first 20):

Title	Keywords	TrueLabel	PredictedLabel
sport_79.txt	kenteri, drug test, test, athen, thanou	sport	sport
medical_565.txt	homeopathi, countri, britain, method, treatment	medical	graphics
space_24.txt	instrument, mission, scienc, fund, magellan	space	space
medical_335.txt	citizen, resid, discrimin, luck, worst	medical	sport
sport_98.txt	club, ronaldo, contract, nobodi, unit	sport	sport
sport_76.txt	race, champion, lap, edinburgh, world champion	sport	sport
space_29.txt	island, plant, hack, long, wright	space	space
medical_158.txt	brand, gilbert, pittsburgh, intern medicin, tri	medical	medical
space_1.txt	exist, bruce, physic, curv, space	space	technologie
entertainment_14.txt	friend, sex, star, citi, comeback	entertainment	entertainment
food_71.txt	allergi, food allergi, food, actual, studi	food	entertainment
sport_37.txt	sprinter, johnson, british, britain, pride	sport	sport
politics_272.txt	parti, elect, lib dem, lib, dem	politics	medical
historical_16.txt	war, world war, german, russia, germani	historical	food
historical_13.txt	genocid, bomb, american, vietnam, aerial	historical	politics
business_3.txt	ebber, worldcom, sullivan, mr, fraud	business	medical
food_78.txt	hungri, fung, eat, food, room	food	politics
sport_11.txt	team, radcliff, paula, gb, event	sport	sport
entertainment_39.txt	host, christma, london, station, radio	entertainment	entertainment
medical_572.txt	wrong, doctor, real, joel, okay	medical	medical

Figure 8: text document classification

of time taken by algorithm for classifying based on input data samples.

$$ET = \sum_{j=1}^{M} IM_j * Time\left(t_{PC}\right) \tag{16}$$

Where, input data samples' IM_j ' and time consumed in extracting keywords ' $Time\left(t_{PC}\right)$ '. It is measured in terms of seconds (sec).

Accuracy ' Acc ': This metric evaluates the effectiveness of document classification process by measuring the proportion of correctly classified documents.

$$Acc = \frac{TP + TN}{TP + FP + TN + FN} \tag{17}$$

Where, true positive' TP ', true negative ' TN ', false positive ' FP ' and false negative ' FN '.

Precision ' Pre ': it measures the proportion of documents that model correctly identified as belonging to a particular class out of all documents.

$$Pre = \frac{TP}{TP + FP} \tag{18}$$

Recall ' Rec ': it measures the ability of classification model to identify all relevant documents belonging to a specific class. It calculates the proportion of actual documents in class that model correctly retrieves.

zF1-score: The F1 score combines precision and recall into a single metric by calculating their harmonic mean, offering a balanced measure of a model's accuracy.

$$F1-score=2*\left[\frac{Pre*Rec}{Pre+Rec}\right] \quad (20)$$

Specificity: it is an important metric in text document classification, especially for minimizing false positive rates.

$$Specificity=\frac{TN}{TN+FP} \quad (21)$$

Classification time 'CT': It represents the time required by method to document classification relative to number of input samples.

$$CT=\sum_{i=1}^{n}S_i*TM\left(DC\right) \quad (22)$$

Where, samples 'S_i' and actual time consumed in classifying text documents denoted by '$TM\left(DC\right)$'. It is measured in terms of seconds (sec).

Figure 9 presents Error rate for proposed PBiC-CELM and existing 3WD-GCN, B-MLCNN, LDA, BERT and LSTM. Result of PBiC-CELM delivers lesser error rate than existing approaches. It is achieved due to integration of CELM and ELMs. Keyword extraction is used to extract features and passed to the ELM, the bivariate correlation analyzes the features frequency in class labels thus reduces classification errors. Error rate using PBiC-CELM was reduced by 12%, 19%, 26% 32% and 40% over 3WD-GCN, B-MLCNN, LDA, BERT and LSTM.

Figure 10 presents Keyword extraction time using six models. PBiC-CELM method demonstrated lower extraction time than other two methods. PBiC-CELM achieved reduction in extraction time by 10%, 21%, 33%, 37% and 40% than 3WD-GCN, B-MLCNN, LDA, BERT and LSTM. This improvement is mainly due to the integration of CNN, which analyzes the words in text documents by Least Angle Regression Analysis thereby reducing extraction time.

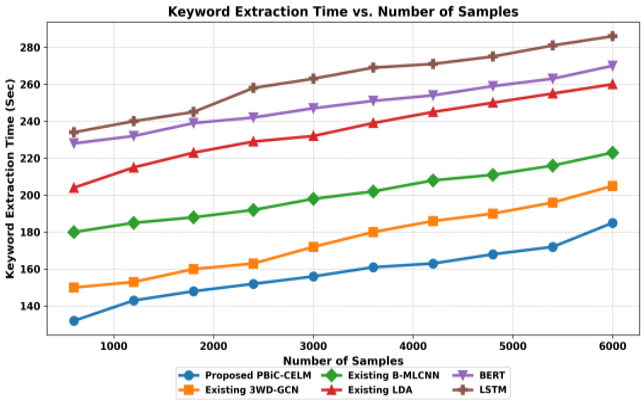

Figure 10: Keyword extraction time

Figure 11 provides accuracy to number of samples. Results of PBiC-CELM outperform existing methods in accuracy. For example, with 600 samples, the PBiC-CELM method reached an accuracy of 0.986, whereas methods 3WD-GCN, B-MLCNN, LDA, BERT and LSTM achieved 0.97, 0.965, 0.955, 0.946 and 0.933 respectively. PBiC-CELM improved accuracy 3%, 6%, 8%, 10% and 13% when compared to other methods. This enhancement is achieved due to use of ELM classifier in PBiC-CELM. Bivariate correlation coefficient examines keywords with class labels, enabling precise classification of text document, which enhances overall accuracy.

Figure 12 shows precision. Among the various models, PBiC-CELM displays superior precision compared to existing approaches. PBiC-CELM achieved precision by 3%, 5%, 7%, 9% and 11% than existing methods. This enhancement is achieved due to use of ELMs, which enables more effective analyses of keywords with class labels. It improves overall accuracy by increasing true positive rate while reducing false positives in text document classification tasks.

Figure 13 depicts recall. PBiC-CELM illustrates advantage in recall than other methods. PBiC-CELM delivers higher recall by 3%, 5%, 6%, 9% and 10% than other models. This recall is achieved due to the CELM and ELM. The convolutional

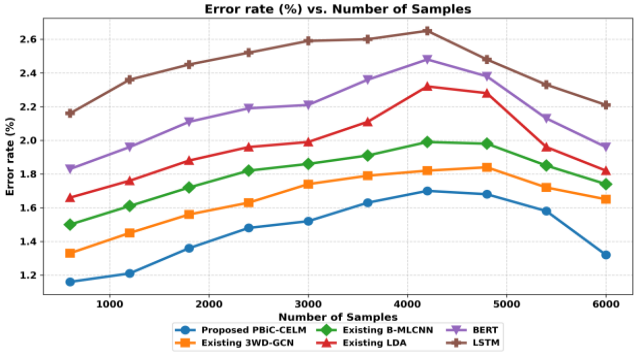

Figure 9: Error rate

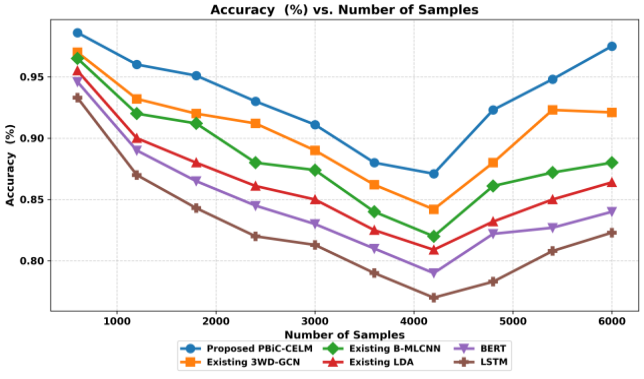

Figure 11: Accuracy

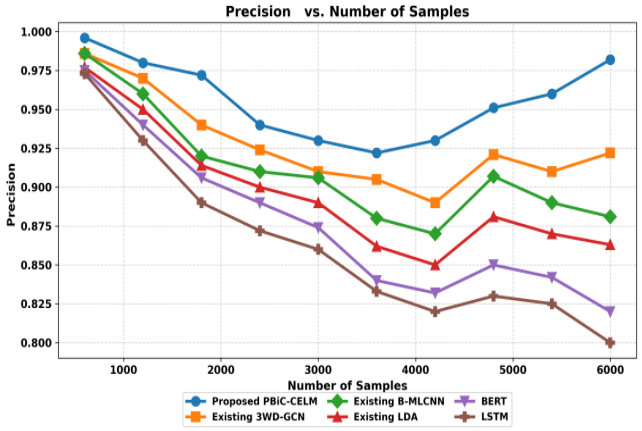

Figure 12: Precision

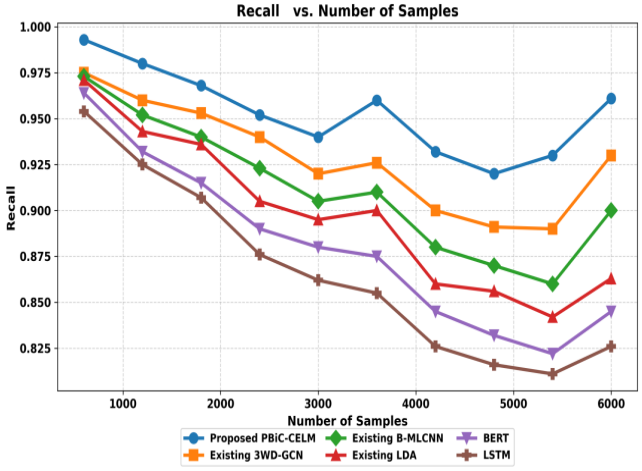

Figure 13: Recall

component captures deep spatial and structural patterns in the input data for identifying subtle or rare features. These rich features are then passed to ELM, which performs text document classification that detects true positives, thereby reducing false negatives and improving recall.

Figure 14 presents F1-score for proposed and existing methods. PBiC-CELM method consistently delivers higher F1-scores across varying sample sizes, attributed to its extreme learning framework that enhances its ability to make accurate text document classification tasks. On average, the PBiC-CELM method demonstrates an F1-score improvement by 3%, 5%, 7%, 9%, 11% when compared to existing 3WD-GCN, B-MLCNN, LDA, BERT and LSTM methods.

Figure 15 illustrates specificity. Results of PBiC-CELM achieved higher specificity compared to existing models. With 600 samples, the specificity obtained as 0.971, 0.901, 0.898, 0.833, 0.816 and 0.797 using PBiC-CELM, 3WD-GCN, B-MLCNN, LDA, BERT and LSTM. PBiC-CELM improved specificity by 7%, 9%, 14%, 16% and 21% than existing methods.

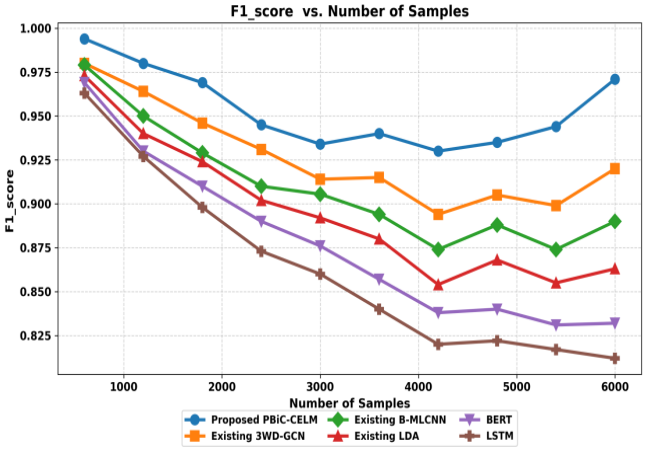

Figure 14: F1 score

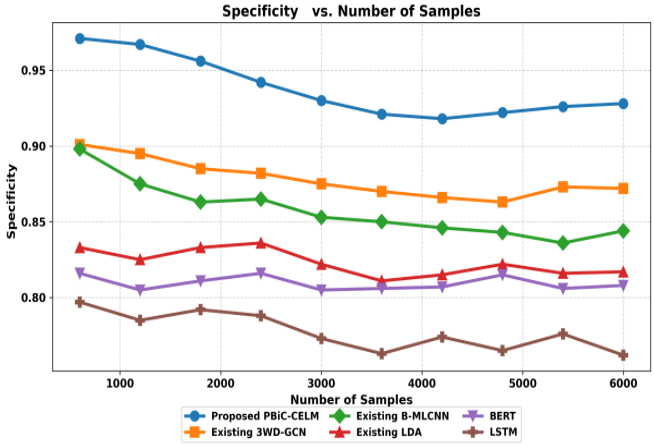

Figure 15: Specificity

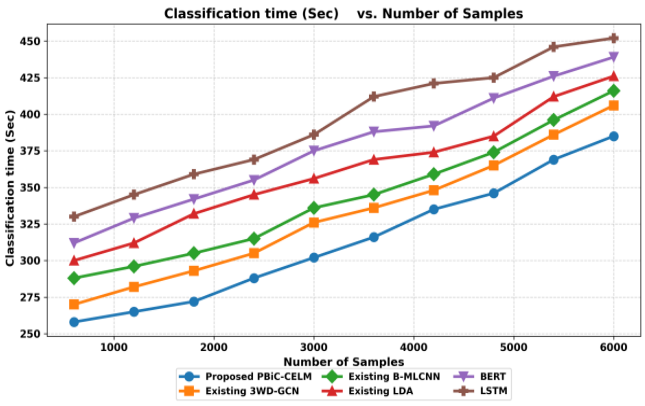

Figure 16: Classification time

Figure 16 presents document classification time. PBiC-CELM of classification time is lesser over other approaches. This is achieved due to efficient text preprocessing and keyword extraction of within the convolutional and pooling layers. These components work together to reorganize the document classification process. PBiC-CELM reduces document classification time by 6%, 9%, 13%, 17% and 21%

compared to existing 3WD-GCN, B-MLCNN, LDA, BERT and LSTM methods.

Conclusion

A PBiC-CELM is developed for text document classification by integration CNN and ELM deep learning model to process user-generated content through text preprocessing and keyword extraction and classification. Keyword analysis and preprocessing is performed within pooling layer of CELM to accurately classify social media documents, while minimizing processing time. The classification is ensured by integrating ELM, using bivariate correlation coefficient to maintain high accuracy. A thorough experimental evaluation was carried out using multiple performance metrics. The proposed PBiC-CELM significantly enhances the accuracy of social media text document classification while reducing computational time and error rate compared to traditional deep learning approaches and state-of-the-art methods.

Limitations and Future Work

- The proposed method is designed for text data communication with higher accuracy. But, a huge number of data not considered during data communication. Future enhancement will focus on vast data for document classification.
- The proposed research work uses only one dataset for performance analysis. Future research work is suggested using various datasets to ensure the applicability and scalability of the provided approach.
- Proposed method failed to consider the memory consumption. Further future work is suggested to analyze and reduce the memory consumption involved in the text document classification.

Data Availability Statement

The dataset used during the current study are available at [28] https://www.kaggle.com/datasets/jensenbaxter/10dataset-text-document-classification.

Acknowledgement

None.

Funding Statement

This study received no funding.

Ethical Approval

No human or animal studies were conducted by any authors for this article.

Consent to Publish

Not applicable

Author's Contribution

All the authors have participated in writing the manuscript and have revised the final version. All authors read and approved the final manuscript.

References

Ahamad, R., & Mishra, K. N. (2025). Exploring sentiment analysis in handwritten and E-text documents using advanced machine learning techniques: a novel approach. *Journal of Big Data*, *12*(1), 1-38.https://doi.org/10.1186/s40537-025-01064-2

Ai, W., Li, J., Wang, Z., Wei, Y., Meng, T., & Li, K. (2025). Contrastive multi-graph learning with neighbor hierarchical sifting for semi-supervised text classification. *Expert Systems with Applications*, *266*, 1-16.https://doi.org/10.1016/j.eswa.2024.125952

Al-Anazi, R. G., Alzaidi, M. S. A., Eltahir, M. M., Alkhiri, H., Alajmani, S. H., Darem, A. A., ... & Alhebaishi, N. (2025). An adaptive search mechanism with convolutional learning networks for online social media text summarization and classification model. *Scientific Reports*, *15*(1), 1-22.https://doi.org/10.1038/s41598-025-95381-4

Alizadeh, M., & Seilsepour, A. (2025). A novel self-supervised sentiment classification approach using semantic labeling based on contextual embeddings. *Multimedia Tools and Applications*, *84*(12), 10195-10220. https://doi.org/10.1007/s11042-024-19086-y

Atandoh, P., Zhang, F., Adu-Gyamfi, D., Atandoh, P. H., & Nuhoho, R. E. (2023). Integrated deep learning paradigm for document-based sentiment analysis. *Journal of King Saud University-Computer and Information Sciences*, *35*(7), 1-15. https://doi.org/10.1016/j.jksuci.2023.101578

Avinash, N. J., Rao, K., Moorthy, R., Raviprakash, B., Raghunadan, K. R., & Venkatadri, M. (2025). A Novel Automatic Text Document Classification Using Learning based Text Classification (LbTC) Approach. *Procedia Computer Science*, *258*, 4279-4290. https://doi.org/10.1016/j.procs.2025.04.677

Baxter, J. (2020). Text document classification dataset: https://www.kaggle.com/datasets/jensenbaxter/10dataset-text-document-classification

Chausson, S., Fourcade, M., Harding, D. J., Ross, B., & Renard, G. (2026). The insight-inference loop: Efficient text classification via natural language inference and threshold-tuning. *Sociological Methods & Research*, *55*(2), 568-615. https://doi.org/10.1177/00491241251326

Chen, Y., Hu, B., & Liu, Y. (2025). Optimizing document management and retrieval with multimodal transformers and knowledge graphs. *PLoS One*, *20*(6), 1-27.https://doi.org/10.1371/journal.pone.0323966

Costa, G., & Ortale, R. (2025). Unifying clustering and representation learning for unsupervised text analysis: A Bayesian knowledge-enhanced approach. *Information Fusion*, *117*, 1-11. https://doi.org/10.1016/j.inffus.2024.102886

Ghanem, F. A., Padma, M. C., Abdulwahab, H. M., & Alkhatib, R. (2024). Novel genetic optimization techniques for accurate social media data summarization and classification using deep learning models. *Technologies*, *12*(10), 1-21.https://doi.org/10.3390/technologies12100199

Jiang, C., Yang, Z., & Yao, J. (2025). Three-Way Decision Enhanced Graph Convolutional Networks for Text Classification. *Neural Processing Letters*, *57*(2), 1-26.https://doi.org/10.1007/s11063-025-11722-4

Li, G., Zhao, X., & Wang, X. (2024). Quantum self-attention neural networks for text classification. *Science China Information Sciences*, *67*(4), 1-13.https://doi.org/10.1007/s11432-023-3879-7

Liao, W., Liu, Z., Dai, H., Wu, Z., Zhang, Y., Huang, X., ... & Li, X. (2024). Mask-guided BERT for few-shot text classification. *Neurocomputing*, *610*, 1-9.https://doi.org/10.1016/j.neucom.2024.128576

Lv, S., Dong, J., Wang, C., Wang, X., & Bao, Z. (2024). Rb-gat: A text classification model based on roberta-bigru with graph attention network. *Sensors*, *24*(11), 1-15.https://doi.org/10.3390/s24113365

Qiu, Q., Tian, M., Tao, L., Xie, Z., & Ma, K. (2024). Semantic information extraction and search of mineral exploration data using text mining and deep learning methods. *Ore Geology Reviews*, *165*, 1-16.https://doi.org/10.1016/j.oregeorev.2023.105863

Rautaray, J., Panigrahi, S., Nayak, A. K., Sahu, P., & Mishra, K. (2025). Deep Learning-Based Feature Extraction Technique for Single Document Summarization Using Hybrid Optimization Technique. *IEEE Access*, *13*, 24515-24529. DOI: 10.1109/ACCESS.2025.3538169

Shao, D., Su, S., Ma, L., Yi, S., & Lai, H. (2025). DA-BAG: A multi-model fusion text classification method combining BERT and GCN using self-domain adversarial training. *Journal of Intelligent Information Systems*, *63*(1), 205-225. https://doi.org/10.1007/s10844-024-00889-2

Su, R., Gao, S., Zhao, K., & Zhang, J. (2025). Adaptive feature interaction enhancement network for text classification. *Scientific Reports*, *15*(1), 1-14.https://doi.org/10.1038/s41598-025-95492-y

Taye, M. M., Abulail, R., Al-Ifan, B., & Alsuhimat, F. (2025). Enhanced Sentiment Classification through Ontology-Based Sentiment Analysis with BERT. *J. Internet Serv. Inf. Secur.*, *15*(1), 236-256. **DOI:** 10.58346/JISIS.2025.I1.015

Wang, M., Kim, J., & Yan, Y. (2025). Syntactic-aware text classification method embedding the weight vectors of feature words. *IEEE Access. 13,* 37572 – 37590. DOI: 10.1109/ACCESS.2025.3545877

Wu, M. (2025). Label knowledge-guided heterogeneous graph contrastive learning for semi-supervised short text sentiment classification. *Journal of Big Data*, *12*(1), 1-38.https://doi.org/10.1186/s40537-025-01276-6

Xi, Q., & Jiang, P. (2025). Design of news sentiment classification and recommendation system based on multi-model fusion and text similarity. *International Journal of Cognitive Computing in Engineering*, *6*, 44-54. https://doi.org/10.1016/j.ijcce.2024.11.003

Xu, G., Chen, Z., & Zhang, Z. (2025). Aspect category sentiment analysis based on pre-trained BiLSTM and syntax-aware graph attention network. *Scientific Reports*, *15*(1), 1-15.https://doi.org/10.1038/s41598-025-86009-8

Xu, Q. (2025). Application of an intelligent English text classification model with improved KNN algorithm in the context of big data in libraries. *Systems and Soft Computing*, *7*, 1-10.https://doi.org/10.1016/j.sasc.2025.200186

Zhang, K., Liu, X., Zhao, N., Liu, S., & Li, C. (2025). Dual channel semantic enhancement-based convolutional neural networks model for text classification. *International Journal of Modern Physics C*, *36*(10), 1-27. DOI: 10.1142/S0129183124420129244 2012-1

Zhang, Q., Chen, S., & Liu, W. (2025). Balanced Knowledge Transfer in MTTL-ClinicalBERT: A Symmetrical Multi-Task Learning Framework for Clinical Text Classification. *Symmetry*, *17*(6), 1-26.https://doi.org/10.3390/sym17060823

Zuo, M., Song, Z., Zhang, Q., Liu, Y., Wu, D., & Cai, Y. (2025). Event type and relationship extraction based on dependent syntactic semantic augmented graph networks. IEEE Access. *13,* 40169 – 40184. DOI: 10.1109/ACCESS.2025.3546963