Bulletin of Electrical Engineering and Informatics
Vol. 15, No. 4, August 2026, pp. 3169~3182
ISSN: 2302-9285, DOI: 10.11591/eei.v15i4.11364

Content categorization using enhanced convolutional neural network for library book availability prediction

Geetha Jayabalan[1,2], Kavitha Venkatesh[3]
[1]Department of Computer Science, Hindusthan College of Arts and Science, Coimbatore, India
[2]Avinashilingam Institute for Home Science and Higher Education for Women, Coimbatore, India
[3]Department of Computer Science with Cognitive Systems, Sri Ramakrishna College of Arts & Science, Coimbatore, India

Article Info

Article history:

Received Sep 9, 2025
Revised May 15, 2026
Accepted Jun 19, 2026

Keywords:

Bidirectional encoder representations from transformers tokenizer
Content categorization
Deep learning
Natural language processing
Optimization techniques
Textual data analysis

ABSTRACT

Deep learning (DL) techniques analyze textual data, but existing research has not explored heuristic approaches for extracting crucial text vectors. Essential preprocessing techniques for cleaning global and local text are also required. This study proposes a DL-based framework to predict book availability based on book-title. The proposed methodology has four phases: data preprocessing, text preprocessing, feature selection, and classification. Missing values, duplicates, and feature extraction are handled during data preprocessing to obtain clean data. Then, lemmatization and bidirectional encoder representations from transformers (BERT) vectorization are applied to derive feature vectors from each word in text preprocessing. Four feature selection algorithms-firefly algorithm (FA), artificial immune system (AIS), genetic algorithm (GA) and ant colony optimization (ACO) were used to select the crucial feature vectors. Comparatively, ACO obtained 90.87% accuracy, 90.87% precision, 100% recall, 91.94% F-score, and was selected as the optimal feature selection algorithm. The extracted features from ACO are trained using convolutional neural network (CNN) to predict book availability. Experimental results show that CNN achieved best performance than decision tree (DT), random forest (RF), k-nearest neighbor (KNN), support vector machine (SVM) and other state-of-art methods, i.e., 99.07% accuracy, 99.83% precision, 99.24% recall, and 99.54% F-score for analyzing textual data.

Corresponding Author:

Geetha Jayabalan
Department of Computer Science, Hindusthan College of Arts and Science
Coimbatore, Tamilnadu, India
Email: jgeethacbe@gmail.com

1. INTRODUCTION

Content categorization is utilized to analyse the content containing large text and classify them in particular domain [1]. At present, the usage of text is increased highly thus it requires content categorization. Content categorization can be accomplished by either manual or automatic. Manual content categorization requires huge labour and time for predicting book availability. In the era of automation, the content categorization plays major role in the area of spam email detection, online tendering and fake news detection. As a result, the presence of machine learning (ML) models made it easy by automatically categorizing the text information for managing the information in internet.

The repository contains the collection of records about a particular domain i.e., library is a repository that contains information regarding books it brought, has, and lend to others for a particular duration. The library comprise of books of various interest that don't fall out of scope of students particularly

focused on academics, storytelling books, and magazines which attracts the most students. Meanwhile, journals and documentaries are less widely utilized by many students. When the student search for particular book, it is required to detect the book availability using book title and author before lending them. Hence, the automatic artificial intelligence-based content categorization is required.

Content categorization is used to classify the text particularly obtained from English language [2]. Existing research applied ML models for content categorization [3]–[5]. However, one of major challenge encountered while performing content categorization is feature reduction of high-dimensional feature space. Feature reduction shows major influences on model performance. It can be accomplished by utilizing feature reduction techniques using metaheuristics approaches. However, selecting the best performing feature selection technique for performing content categorization is problematic and it is required to identify them for optimizing ML models [6].

Convolutional neural network (CNN) consist of convolutional layer that utilizes the convolving filters to obtain features in high dimension. However, determining the correct size of essential filter is difficult. Because, higher value in filter utilize many resources while performing model training, but less filters are prone for minimal accuracy [7], [8]. CNN captures information regarding library content from each features for classification.

The existing research applied ML techniques which is widely utilized for content categorization, but the insight of metaheuristic approach with classification technique not yet exploited. At present, the combination of metaheuristic approach and classification technique have not been proposed for detecting the book availability. Meanwhile, deep learning (DL) techniques are widely utilized by past researchers for better classification in the field of content categorization [9], [10]. However, DL models are particularly prone to overfitting due to redundant and irrelevant features which can be handled using increases input dimensionality. In such case, metaheuristic approach is used for feature selection to handle overfitting.

In this study, an attempt has been made to combine metaheuristic approach with classification technique for detecting the book availability. The major contribution of this proposed methodology is listed below: i) to preprocess the data using essential data preprocessing and text preprocessing techniques to ensure data quality; ii) to select the best performing feature selection technique among several metaheuristic approaches such as artificial immune system (AIS), genetic algorithm (GA), ant colony optimization (ACO), and firefly algorithm (FA) based on the performance of random forest (RF) classifier in terms of F-score, recall, precision, and accuracy; and iii) to compare the performance of combined feature selection technique and DL technique for book availability detection with other ML models and other state-of-the-art methods.

Unlike existing studies on text classification which directly applies CNN on high-dimensional features obtained from textual embeddings, this study introduces a metaheuristic-driven feature selection layer prior to DL classification. Four optimization algorithms (AIS, GA, FA, and ACO) are used to optimize the bidirectional encoder representations from transformers (BERT)-generated feature that eradicates unrelated and redundant features. The combination of metaheuristic and DL techniques improves classification and reduces model complexity, leading to enhanced performance.

The paper is ordered as follows. Section 2 summarize the relevant studies on content categorization. The proposed methodology for book availability detection is explained in section 3. Section 4 discusses the experimental findings and section 5 analyses the obtained findings. Section 6 concludes the paper.

2. RELATED STUDIES

The existing research on several ML and DL techniques for content categorization using textual data is limited which is a major concern. Some of these recent studies are reviewed in this section for content categorization in detail.

Combined the two DL techniques such as BERT and CNN techniques for classifying the long-text in Chinese character which applies BERT for representing the feature vector in news text for contextual understanding [11]. Then, CNN is applied in local feature convolution to obtain key phrases for predicting the news category. Meanwhile, sentiment analysis is incorporated by [12] to analyse the contents in social media by integrating CNN and long short-term memory (LSTM) for understanding characters in English and Roman Urdu. LSTM is used to preserve the long-term dependencies in characters, then these dependencies are processed using CNN to retrieve local features. Several ML techniques such as RF, logistic regression (LR), support vector machine (SVM), Naïve Bayes (NB), and k-nearest neighbor (KNN) are analysed using the local features. SVM performs better compared to others using UCL, RUSA-19, and RUSA dataset for text classification.

On the other hand, the performance of combined CNN and LSTM is enhanced by extracting only key features using term frequency-inverse document frequency (TF-IDF) that eliminates the features obtaining less weights [13]. The key features are used to extract corresponding vectors using Word2Vec

which is further analysed through CNN+LSTM for text categorization using THUCNews and Taobao review datasets. The limitations in CNN and LSTM is counter measured by [14] by proposing an ensemble method that combines DL techniques namely LSTM, CNN, and GRU with meta-classifiers for text classification. The performance is evaluated using several benchmark datasets for text classification.

Predicted risk among university students by analysing e-book reading behaviors in terms of page navigation, marker modifications, and memo editing activities, using neural network, RF, Gaussian NB, SVM, LR, and decision trees (DT) [15]. Meanwhile, compared to ML techniques, neural network namely artificial neural network (ANN) is widely recommended by [16] using back propagation network (BPN) for text categorization which shows higher performance. The combination of CNN and BiLSTM using attention mechanism is highly recommended by [17] for sentiment classification that analyses text present in various documents.

According to Suneera and Prakash [18], LR-based classifier and Bi-channel CNN are considered as best performing algorithm for text classification by analysing news information obtained from 20 newsgroups. Muhammad *et al.* [19] recommended CNN-based DL technique over SVM, LR, RF, and NB by achieving 85% accuracy for classifying short-text containing 6000 instances with multi-class classification.

To enhance the performance of text classification, preprocessing plays important role in data cleaning, and handling missing values. Alqahtani *et al.* [20] applied several classifiers including LR, RF, KNN, LSTM, gated recurrent unit (GRU), and ANN after preprocessing. As a result, LSTM obtained 92% accuracy and outperform others for textual classification. Meanwhile, Bi-LSTM is highly considered for classifying the textual data [21] and the performance is compared with SVM, gradient boosting (GB), NB, conditional random field (CRF), and recurrent weighted average (RWA). Bi-LSTM attained 98.5% accuracy which is higher than other algorithms.

Hybrid approach is proposed that combines traditional feature engineering with DL, such as integrating filter-based selection with deep CNNs to obtain the high performance over state-of-the-art methods for analysing and classifying the text with 7.7% margin in 20 newsgroups and 6.6% margin in BBC news dataset [22].

Some researchers incorporate supervised learning techiques for analysing diverse contexts and language. Eight classifiers including LR, DT, SVM, AdaBoost (AB), RF, multinomial Naïve Bayes (MNB), multilayer perceptrons (MLP), and GB are applied to analyse the textual data for classification by eliminating the stop words. SVM outperforms other in terms of 95% area under curve, 84% ROC, 97% accuracy for text classification [23]. In the area of sentiment analysis, RF classifier are highly recommended by [24] to review the movie based on the comments. RF achieved 85% accuracy, 85% recall, 85% precision, and 85% F-score which is higher compared to NB.

A stacking ensemble method is proposed named RF+GB which combines RF and GB classifiers with TF-IDF for feature extraction based on weights [25]. The performance is compared with several feature extraction techniques consist of NGrams, TF-IDF, glove, bag-of-words (BoW), and Word2vec. The ensemble technique is compared with LR, DT, MNB, SVM, KNN, and RF to recognise the best performing algorithm. RF+GB attained 85% accuracy, 84% F1-score, 87% recall, and 82% precision which is higher compared to other classifier to analyse customer information of amazon users. LSTM is highly recommended for analysing textual content in Turkish language. In addition, SVM, NB, RF, and LR classifiers are compared based on accuracy. As a result, LSTM obtained 85% accuracy which is higher and SVM obtained 75% accuracy which is lowest for analysing Turkish text [26].

Developed a classification algorithm named KFE-CNN for analysing the news text in chinese for classification [27]. It extracts the features containing semantics perform binary vector representation which transforms text features into numerical representation for CNN-based classification. It achieved 97.84% average accuracy and 98% F1-score using THUCNews dataset for news text classification. Combined the fastText to generate the semantic vectors with CNN to extract the global features for classifying the Amharic texts in news document [28]. CNN achieved 93.79% accuracy which outperforms other algorithms including SVM, MLP, DT, extreme gradient boosting (XGBoost), and RF for text classification.

Incorporated the combined multinomial NB for classification with TF-IDF for feature extraction after performing pre-processing i.e., tokenization, eliminating stop word, lemmatization, and dimensionality reduction using news dataset to classify news topics [29]. It achieved minimal performance with 77.39% accuracy which is comparatively lesser compared to other past studies. Alternatively, DL techniques such as transformer-based models are widely used in text classification. A hybrid algorithm namely BERT+ALBERT is integrated to perform contextual embedding and parameter-sharing strategies [30]. The performance of hybrid BERT+ALBERT is higher compared to LSTM, BERT, and ALBERT for text classification.

Alternatively, large language models are highly utilized at present using dependency triplet features which is evaluated using 16 ML models. Among them, SVM achieved higher performance of 93% F-score using 10-fold cross-validation and outcome is evaluated using SHAP analysis [31]. Table 1 (in Appendix) [11]-[31] summarizes the literature review on classification using textual data.

From the literature study, it is observed that most previous studies applied DL techniques for content categorization without utilizing feature selection, resulting in high-dimensional redundant features. Moreover, prior work employs conventional wrapper-based feature selection methods such as KFE with CNN [27], which follow greedy local search strategies and are prone to local optima which limits the exploration of global feature space. Although BERT tokenization integrated with CNN has been explored [11], the absence of metaheuristic-driven feature selection leaves a critical gap in optimizing semantic embeddings and feature relevance.

So, in this study, an attempt has been made to apply four heuristic approaches such as AIS, GA, ACO, and FA for feature selection using probabilistic global search strategies for exploration and exploitation. In comparison, ACO is conceptually superior due to its pheromone-based adaptive learning mechanism, which enables dynamic feature subset optimization and avoids premature convergence for feature selection. The selected features are combined with DL techniques for content classification to detect the book availability.

3. PROPOSED METHOD

In this section, the proposed methodology for book classification using content categorization are detail explained below. The proposed methodology consists of four phases such as data preprocessing, text preprocessing, feature selection, and classification to classify whether the book is available or not. The overview of the proposed methodology to classify the book availability is described in Figure 1. In addition, Figure 2 depicts the working diagram of each phases.

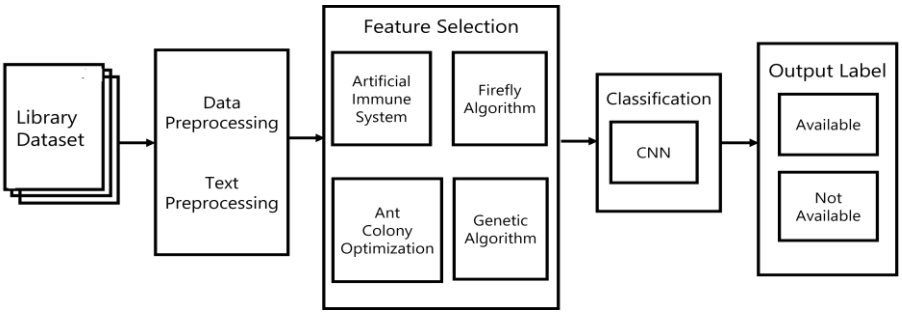

Figure 1. Overview of the proposed method

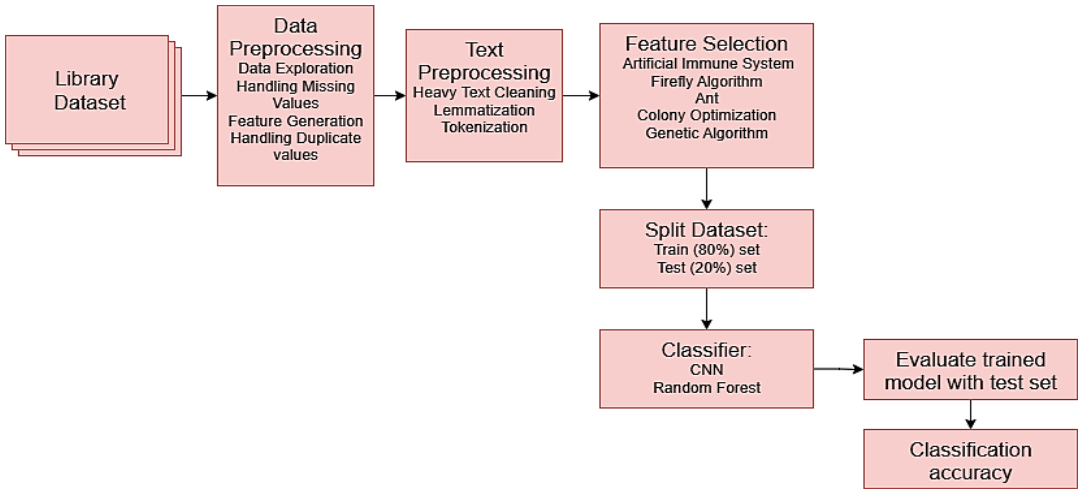

Figure 2. Working diagram of each phase in the proposed method

The major objective of the proposed methodology is to evaluate four feature selection algorithms with RF classifier using various performance metrics and select the best among them for book availability detection.

3.1. Dataset

In this study, 3856 book details from various genres and language are collected from 2022 to 2024 to train and test the proposed methodology, since there is no publicly available library dataset. The book available in library is commonly relevant to academics and it consist of genres including education, research, computers, stories, and literature. The information of each book is saved in a repository that consist of title, author, book classification number, book barcode, book issue date, due date of book return, and actual return date. The primary objective of the proposed methodology is to analyse the borrowing patterns of book and predict the book availability.

3.2. Data preprocessing

The dataset collected from real environment contains irregularities and requires to clean the data to achieve the data quality through several preprocessing techniques. Without preprocessing, analysing raw data affects the model effectiveness for book availability detection. In this section, four data preprocessing techniques are applied to clean the raw data. So, the raw data undergoes data exploration, handling missing values, feature generation, and handling duplicates. While exploring the data, a statistics summarizes two alerts using pandas profiling as shown in Figure 3, they are; i) the presence of duplicate rows witnessed is less than 0.1% and ii) the 'author' feature contains 1.2% of missing values which can be further handled using imputation techniques or remove them. Since, the author name cannot be imputed, it is replaced with "No author".

To predict the availability of book, the new feature named "is available" is generated based on issue date, due date, and return date. The "is available" feature contains two values which is 0/1. If the book is returned, the book availability is considered "1". If the book is not returned, the book availability is considered "0". Next, the duplicate rows are removed if the value of book availability is more than 1 to maintain the consistency.

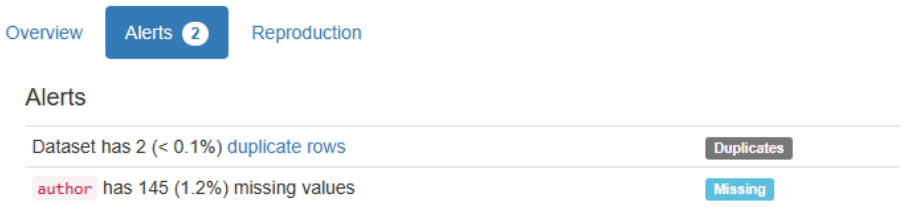

Figure 3. Results of data exploration

3.3. Text preprocessing

After performing necessary data preprocessing, the text preprocessing techniques are used to process unstructured textual information such as book titles and author names through natural language processing (NLP) algorithms. Some of them includes heavy text cleaning, lemmatization, and tokenization are discussed in this section.

a. Heavy text cleaning

In heavy text cleaning, the special characters, punctuation, numbers, HTML tags, stop words, and symbols present in text are removed for handling noise and satisfied data consistency.

b. Lemmatization

To extract the meaningful information from the textual data, lemmatization is performed. As a result, all the words in textual data is reduced to base form that minimizes dimensionality and enhances the semantic interpretation.

c. Tokenization

The lemmatized data is then converted into numerical representation in tokenization technique using BERT tokenizer for embedding. BERT is a pre-trained NLP which is basically derived from transformers for tokenization. BERT tokenizer splits the lemmatized data into 30 sub-fragments containing tokens for each subword using WordPiece embedding. It bidirectionally analyses the adjacent words captured from context of words. Finally, the obtained tokens are mapped into numerical representation using vocabulary in BERT tokenizer.

Table 2 shows the result obtained from text preprocessing. It is understandable, the text preprocessing ensures the uniformity in semantics and transforms the unstructured textual data into structured numerical representation for increase model performance.

Table 2. Results of text preprocessing

Text preprocessing	Attributes	Values
Before text preprocessing	Number of columns	2
	Column data type	Textual data
After text preprocessing	Number of columns	30
	Column data type	Numerical data

3.4. Feature selection

The structured numerical data is used by variants of feature selection algorithm namely AIS, GA, ACO, and FA based on various heuristic algorithms in this section. The main objective is to evaluate the working of RF using features selected from four algorithms and identify the best among them based on F-score, recall, precision, and accuracy. The working mechanism of feature selection in proposed methodology is shown in Figure 4.

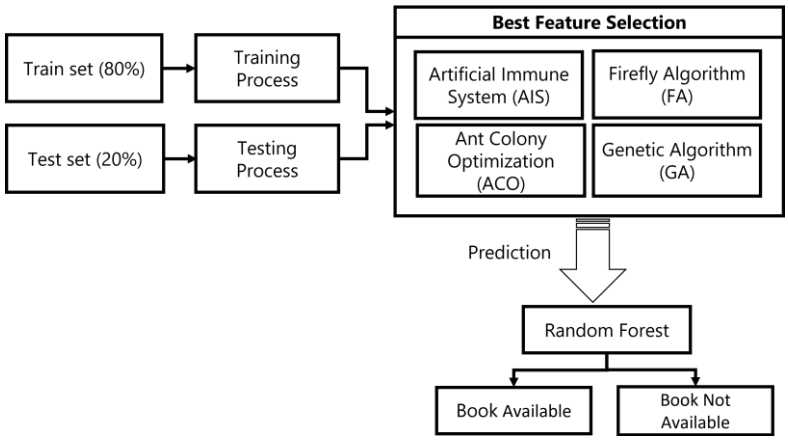

Figure 4. Method for feature selection using RF

The preprocessed data from BERT tokenization from previous section is processed using above mentioned algorithms for selecting best performing features. The best feature selection algorithm among four techniques is selected by evaluating the performance of RF. The aim is to identify the best feature selection technique that in future enhances the performance of classifiers using selected features for content categorization. The detailed explanation of four feature selection algorithms and their working procedure is explained in detail in following subsection.

3.4.1. Artificial immune system

AIS [32] is the human immune system inspired artificial intelligence model that primarily focus on solving complex problems such as feature selection. AIS operates by considering features as antigens for feature selection, and the selection process works as immune system that segregates self and non-self instances. Then, a set of antibodies which is represented as feature subsets are generated and the affinity for the feature subsets are evaluated to distinguish the prediction classes. Subsequently, an iteration is performed to select the best feature subset in immune system using clonal selection and affinity maturation. The affinity value is formulated using (1):

$$\text{Affinity}=\sum_{i=1}^{N} w_i \, Sim(x_i, y_i) \quad (1)$$

Where, N is the number of features, w_i is the weight assigned to feature$i$, and $Sim(x_i, y_i)$ is the similarity between feature i and the class labels. Features with higher affinity values are retained and considered as an optimal subset that further enhances performance in terms of accuracy. AIS provides robust solution and is particularly useful while dealing with high-dimensional datasets.

3.4.2. Firefly algorithm

The FA is one of well-known nature-inspired metaheuristic optimization algorithm that derives from brightening behavior of fireflies [33]. The working of FA for feature selection is as follows: i) each firefly is represented as a potential solution, and the intensity of firefly flashes is represented as the fitness of solution;

ii) while fireflies are attracted to brighter individuals, they tend to move in the search space i.e., better solutions attracts other fireflies; and iii) randomness is then incorporated to explore the solution space and to interrupt local optima. The location of a firefly i towards a brighter firefly j is shown in (2):

$$x_i(t+1) = x_i(t) + \beta e^{-\gamma r_{ij}^2}\left(x_j(t) - x_i(t)\right) + \alpha\, \varepsilon(t) \quad (2)$$

Where x_i and x_j are the locations of fireflies i and j, r_{ij} is considered as the distance measure using Euclidean between them, β is considered as value for controlling the attractiveness, γ is considered as a light absorption coefficient, $\varepsilon(t)$ is considered as a random noise and α is considered as a random parameter. By iteratively updating positions, the FA converges towards an optimal set of features for a given problem.

3.4.3. Ant colony optimization

ACO applies the scavenging actions of ants widely used for optimization [34]. The working of ACO algorithm for feature selection is as follows: i) each ant is considered as a subset of features; ii) then, the ants explore in feature space and produce a path while searching for food sources; iii) the pheromone trails are employed to communicate among ants, where stronger pheromones indicate better solutions; and iv) ants then move towards features with higher pheromone concentrations which is biased in the search toward potential subsets. The pheromone of an ant i towards an ant j is shown in (3):

$$\tau_{ij} = (1-\rho)\tau_{ij} + \Delta\tau_{ij} \quad (3)$$

Where τ_{ij} is the level of pheromone between two ants i and j, ρ is known as the evaporation rate of pheromone, and $\Delta\tau_{ij}$ is known as the pheromone amount deposited by ants. Over iterations, ACO provides an optimal feature subset by exploiting the working of pheromone trails.

3.4.4. Genetic algorithm

GA is an optimization algorithm that applies the behaviour of genetics and natural selection [35]. The working of GA algorithm for feature selection is as follows: i) initially, the potential feature subsets are represented as individuals in a population; ii) then, the population undergoes continuous evolution to refine these subsets through genetic operations such as selection, crossover, and mutation; iii) a population of feature subsets are generated in random manner; iv) subsequently, selection is achieved based on the fitness of subsets which measures the performance through evaluation metrics; v) it determines which individuals is capable to undergo next process; vi) in crossover, the features from two parent subsets are combined to create offspring based on genetic recombination; and vii) next is mutation which introduces small changes to feature subsets. The process of mutation G_mutated of an individual is shown in (4):

$$G_{mutated} = G_{original} + Rand \times MutationRate \quad (4)$$

where, G is the features of an individual, $Rand$ is a random vector, and $MutationRate$ is the value used to control the mutation intensity. GA is particularly effective for feature selection when dealing with large and complex datasets.

Existing studies utilize CNN models to train raw or weighted textual features, whereas the proposed method applies metaheuristic feature selection before CNN training. This differs from conventional pipelines by optimizing the embedding feature space. As a result, it improves learning efficiency, reduces feature redundancy, and enhances the classification performance.

3.5. Classification

After performing feature selection, the subset obtained from the best performing feature selection technique is analysed using CNN classifier to detect the book availability. CNN is highly considered the best performing DL algorithm which is well-known to uncover the hidden patterns and relationships among data items with respect to their adjacent positions [36]. In addition, it acquires knowledge on textual data represented in 1D structure containing word order through convolutional layer. It apprehends confined interactions with adjacent words based on context gaps. Then, the global features are extracted using pooling layers. However, an architecture of CNN model comprise of numerous convolutional and pooling layers. The working model of CNN for content categorization is shown in Figure 5 with the layers.

3.5.1. Embedding layer

The embedding layer is a layer in CNN that derives the feature vectors containing token ID of each word. Each features are transformed into dense vector of same size. It produce features containing token ID

in numerical representation that is suitable for training convolutional layer in 2D image format. It is trained using CNN to detect the text class using the sequence of text vectors.

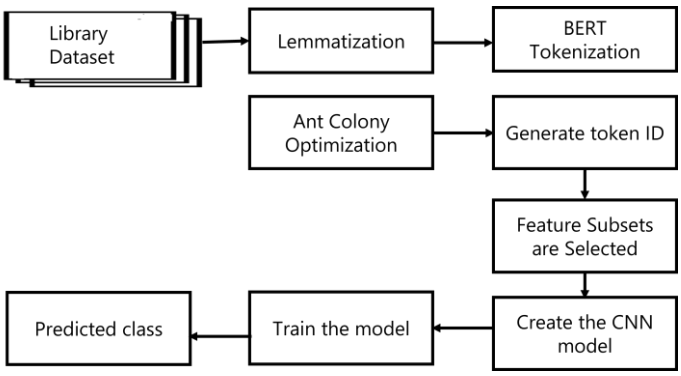

Figure 5. Working of CNN for content categorization

3.5.2. Convolutional layer

Next is the convolutional layer which comprising of several brain-like neurons to compute the action through processing is mentioned as follows [37]: i) initially, a weighted sum is calculated to compute the output based on 'nonlinear' as an activation function while adding them and ii) in convolution layer, there are different-width filters in matrix that is embedded using book title to excerpt diverse features which is represented as vectors with respect to each filter. It also produces a corresponding feature map. The token ID filter in convolution layer considers locations which is unique and varies from each token ID. In case of higher layers, the filter acquires syntactic or semantic relationship among token ID.

3.5.3. Pooling layer

The dimension of mapping features is altered through performing operations such as merge or select using best features among adjacent features. It also handle various global values using layers in higher level. In the study, 1-max-pooling is incorporated in pooling layer of CNN to minimize the parameter size and weights in CNN with the focus on limiting the overfitting issue.

3.5.4. Fully connected layer

After pooling layer, every neurons positioned in all layers are entirely connected with previous layer. The structure of the connection mimics the layers of traditional neural network models. While connecting layer, dropout regularization is performed to evade overfitting which further enhances the general performance. After training, the parameters of the CNN is reserved for later user to predict classes of unlabeled book availability. The hidden layer in traditional feedforward neural network (FNN) is also considered as fully connected layer. It is also known as subcategory containing kernel size 1 which is used as last layer of CNN that contains various trainable layer with weights.

3.5.5. SoftMax function

Finally, the fully connected layer containing of vector representations does undergoes the Softmax function to return the probabilities of output label through performing probability distribution.

4. EXPERIMENT

This section discusses the performance of CNN which is evaluated using features selected from best performing feature selection algorithm among AIS, FA, ACO, and GA for detecting book availability. The results of the CNN with four feature selection algorithms are compared with existing state-of-art techniques. Experiment is done using Python via Jupyter Notebook in Windows 10 containing Core i7 as a processor with 16 GB RAM. The information on the experimental results are listed below.

4.1. Dataset

The proposed methodology is evaluated using library dataset containing information of 3856 book containing book title, book author, book classification number, book barcode, book issue date, due date of book return, and actual return date. The data preprocessing mentioned in proposed methodology is done to

obtain the dataset free from missing values and replace them. Then, text preprocessing is done to obtain only stem words from combined book title and author. Then the stem words are processed using BERT tokenizer to obtain numerical ID for each token from 1 to 30.

4.2. Convolutional neural network training parameters

The hyperparameters in CNN classifier is tuned using library dataset to enhance the CNN's performance for content categorization to detect book availability. It is required to choose the best hyperparameter to increase the performance of CNN classifier. The selected features obtained from feature selection algorithm containing token IDs are trained using CNN classifier. The CNN is analysed using selected features to categorize the numerical tokens of book title and author into two categories i.e., 1 denotes book is available, 0 denotes book is not available. In CNN classifier, there are flatten layer, pooling layer, and convolutional layer with the value of 1, 3, 3 which is present with the network parameter contains 0.5 dropout rate. The epoch for training the CNN classifier is 50. The dataset is divided into 80% training set and 20% testing set. The performance metrics containing F-score, recall, precision, and accuracy is used to evaluate the performance of CNN with four diverse feature selection techniques. Table 3 shows the hyperparameters used in CNN classifier.

Table 3. Parameter specification of CNN classifier

Hyperparameter	Value
Filter size	[32, 64, 128]
Kernel size	[2,1,1]
Activation	Sigmoid
Batch size	32
Dropout	0.5
Optimization	Adam

From the Table 3, it is observed that hyperparameters in CNN such as Filter sizes, kernel size, batch size and dropout is specified with the value of [32, 64, 128], [2, 1, 1], 32, and 0.5 to capture multi-level textual patterns, while ensuring stable training and minimal overfitting. To perform binary classification, Sigmoid is selected for activation and Adam is selected for optimization.

4.3. Performance metrics

In this study, the four main performance metrics including F-score, recall, precision and accuracy are incorporated to evaluate the performance of proposed methodology for content categorization [38] using false negative (FN), false positive (FP), true negative (TN), and true positive (TP). Table 4 displays the description of performance metrics.

Table 4. Performance metrics used in this study

Metrics	Definition	Formula
Accuracy	It is defined as a measure of classifier performance based on overall detection	$\frac{TP+TN}{TN+TP+FN+FP}$
Precision	It calculates a correctly detected available book	$\frac{TP}{TP+FP}$
Recall	It calculates the detection of all available book	$\frac{TP}{TP+FN}$
F-score	It is defined as harmonic mean between precision and recall.	$2 \times \frac{Precision \times Recall}{Precision+Recall}$

5. RESULTS AND DISCUSSION

After preprocessing, four feature selection techniques namely AIS, FA, ACO, and GA is trained using RF classifier to select the best feature selection algorithm using F-score, recall, precision, and accuracy by analysing library book dataset. Figure 6 shows the performance RF with four diverse feature selection algorithms to classify the book availability based on F-score, recall, precision, and accuracy.

From the result, it is understandable RF using GA algorithm 87.83% accuracy, 89.87% precision, 100% recall, and 88.72% F-score which is lower. However, the performance of RF with AIS is comparatively higher than GA by obtaining 88.82% accuracy, 89.53% precision, 100% recall, and 89.89% F-score. The performance of RF with FA achieved higher performance by achieving 89.83% accuracy, 90.83% precision, 100% recall, and 89.92% F-score. Meanwhile, RF with ACO algorithm obtains higher performance by attaining accuracy (90.87%), precision (90.87%), recall (100%), and F-score (91.94%)

compared to AIS, FA, and GA in which they obtain less performance. Thus, it is concluded that performance of RF using ACO optimization technique outperforms other optimization approaches in detecting book availability and is considered as best performing feature selection algorithms.

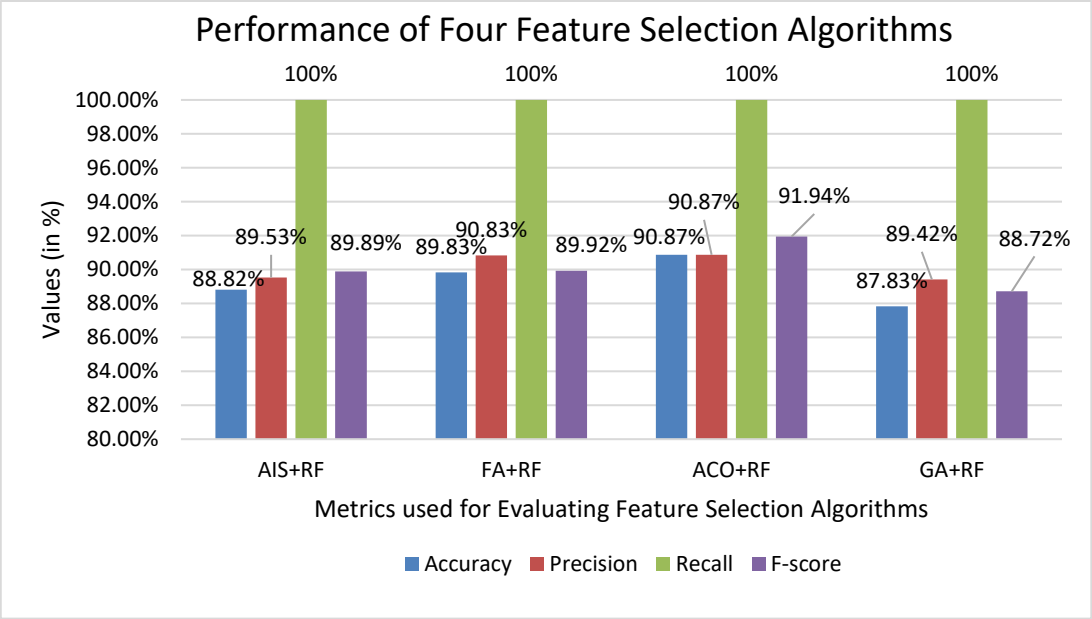

Figure 6. Performance of RF with four feature selection algorithms

Then, the features selected from ACO are trained using the proposed CNN classifier and is evaluated using metrics mentioned in Table 5. However, the performance of CNN with other classifiers using feature obtained from best performing ACO algorithm is discussed below.

Table 5. Performance of CNN and other ML models using ACO-based feature selection algorithm

Model	Accuracy (%)	Precision (%)	Recall (%)	F-score (%)
DT	89.49	89.83	99.66	89.78
SVM	89.66	89.83	99.83	89.83
KNN	89.83	88.83	100	89.91
RF	90.87	90.87	100	91.94
Proposed CNN	99.07	99.83	99.24	99.54

5.1. Convolutional neural network vs other machine learning models

In this section, the performance of CNN is compared with other ML models using features obtained from ACO algorithms based on accuracy, precision, recall, and F-score. Some of the ML models used are DT, SVM, RF, and KNN for library book availability prediction using scikit-library available in "sklearn" package. Table 5 shows the comparison between the performance of CNN with other classifiers using ACO-based feature selection algorithm to classify the book availability using accuracy, precision, recall, and F-score.

From the Table 5, it is observed that DT achieved highest recall of 99.66% but 89.83% precision and 89.49% accuracy which is lesser compared to SVM. Meanwhile, SVM outperforms DT by attaining 89.66% accuracy, 89.83% precision, and 99.83% recall. Followed by KNN with 89.83% accuracy and 88.83% precision and 100% recall and but underperforms RF which attained 90.87% accuracy and 91.94% F-score higher than other ML models. However, CNN obtained 99.07% accuracy, 99.83% precision, 99.24% recall, and 91.94% F-score using features selected from ACO algorithm and outperforms DT, SVM, KNN, and RF for book availability prediction using text-based content categorization. i.e., CNN performs better when combined with ACO-based feature selection for book availability prediction.

The superior performance of combined ACO with CNN shows the effectiveness of metaheuristic optimization by selecting only required features and eliminating the irrelevant features improves feature representation. As a result, it enhances the performance of DL technique for text classification, which has not been explored in existing studies on content categorization for book availability prediction.

The performance of proposed CNN is compared with other state-of-art methods in Table 6. The comparative analysis on Table 6 shows that the proposed CNN with ACO model attained 99.07% accuracy, 99.83% precision, 99.24% recall, and a 99.54% F-score, where the superior performance outperforms all existing methods. However, existing state-of-art method using BERT and CNN [11] obtained 96.2% accuracy which is lower than proposed ACO+CNN method. Followed by, KFE and CNN [27] acquiring 97.84% accuracy, and BERT and CNN+LSTM [12] obtained 90.4% accuracy showcase lower performance. Similarly, TF-IDF and CNN+LSTM [13], CNN-LSTM [17], and CNN [28] achieved 91.81%, 83.29%, and 93.79% accuracy, respectively. From the above analysis, it is known that ACO+CNN achieved utmost performance which highlights the significance of ACO for feature selection influences classification accuracy.

Table 6. Performance of ACO and CNN with the state of art methods

Model	Accuracy (%)	Precision (%)	Recall (%)	F-score (%)
BERT and CNN [11]	96.2	-	-	-
KFE and CNN [27]	97.84	-	-	98
BERT and CNN+LSTM [12]	90.4	89.5	90.3	89.8
TF-IDF and CNN+LSTM [13]	91.81	-	-	-
CNN-LSTM [17]	83.29	-	-	-
CNN [28]	93.79	93.63	93.67	93.76
Proposed ACO+CNN	99.07	99.83	99.24	99.54

6. CONCLUSION

The content is successfully categorized in this paper by analysing text using enhanced CNN using feature selection algorithms for library book availability prediction. At first, the collected library data is preprocessed by handling missing values in data preprocessing. Followed by text preprocessing is performed using heavy data cleaning, lemmatization and BERT tokenizer to obtain token IDs of book title and author. Then, four variants of feature selection algorithms such as AIS, FA, ACO, and GA is applied for feature selection using preprocessed data. Among these four variants, ACO performs better by obtaining higher accuracy, precision, recall, and F-score using RF and considered as best performing feature selection algorithm. Then, the feature selected using ACO is analysed CNN and other ML model for library book availability prediction. The experimental results shows that proposed CNN with ACO obtained 99.07% accuracy, 99.83% precision, 99.24% recall, and a 99.54% F-score which is higher compared to other state-of-art methods by performing content categorization to predict library book availability.

In future, advanced techniques from large language models such as robustly optimized bidirectional encoder representations from transformers approach (RoBERTa), generative pre-trained transformer (GPT) 3 and extreme language network (XLNet) could be utilized for classifying the documents containing long-texts.

FUNDING INFORMATION

Authors state no funding involved.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Geetha Jayabalan	✓	✓	✓					✓	✓	✓	✓			
Kavitha Venkatesh		✓		✓	✓	✓	✓			✓		✓	✓	✓

C : **C**onceptualization
M : **M**ethodology
So : **So**ftware
Va : **Va**lidation
Fo : **Fo**rmal analysis
I : **I**nvestigation
R : **R**esources
D : **D**ata Curation
O : Writing - **O**riginal Draft
E : Writing - Review & **E**diting
Vi : **Vi**sualization
Su : **Su**pervision
P : **P**roject administration
Fu : **Fu**nding acquisition

CONFLICT OF INTEREST STATEMENT

Authors state no conflict of interest.

DATA AVAILABILITY

The data that support the findings of this study are available on request from the corresponding author upon reasonable request.

REFERENCES

[1] Y. Zhou, "A Review of Text Classification Based on Deep Learning," in *Proceedings of the 2020 3rd International Conference on Geoinformatics and Data Analysis*, Apr. 2020, pp. 132–136, doi: 10.1145/3397056.3397082.

[2] X. Luo, "Efficient English text classification using selected Machine Learning Techniques," *Alexandria Engineering Journal*, vol. 60, no. 3, pp. 3401–3409, 2021, doi: 10.1016/j.aej.2021.02.009.

[3] D. Endalie and G. Haile, "Hybrid Feature Selection for Amharic News Document Classification," *Mathematical Problems in Engineering*, pp. 1–8, Mar. 2021, doi: 10.1155/2021/5516262.

[4] A. R. and P. P., "Deep Learning With Conceptual View in Meta Data for Content Categorization," in *Deep Learning Applications and Intelligent Decision Making in Engineering*, IGI Global, 2021, pp. 176–191, doi: 10.4018/978-1-7998-2108-3.ch007.

[5] A. A. Alemu, M. D. Melesse, D. A. Mengesha, and M. A. Widneh, "Attention based hybrid deep learning models for multi class Amharic news categorization with Explainable AI," *Discover Applied Sciences*, vol. 8, no. 2, Dec. 2025, doi: 10.1007/s42452-025-08121-8.

[6] T. Stephan, P. S, C.-C. Lin, and S. Agarwal, "A Comprehensive Study of Grey Wolf Optimizer Variants for Optimizing Feature Selection in High-Dimensional Data," *Applied Artificial Intelligence*, vol. 40, no. 1, Dec. 2025, doi: 10.1080/08839514.2025.2601378.

[7] M. Wang *et al.*, "Adaptive Boosting LLMs for Text Classification," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 37, no. 6, pp. 2770–2781, 2026, doi: 10.1109/tnnls.2025.3639613.

[8] F. M. L. Alrufaye, H. I. Mhaibes, and A. F. Neamah, "Neural Networks Algorithm for Arabic Language Features-Based Text Mining," *IOP Conference Series: Materials Science and Engineering*, vol. 1045, no. 1, pp. 1-8, Feb. 2021, doi: 10.1088/1757-899x/1045/1/012003.

[9] B. Jan *et al.*, "Deep learning in big data Analytics: A comparative study," *Computers &; Electrical Engineering*, vol. 75, pp. 275–287, May 2019, doi: 10.1016/j.compeleceng.2017.12.009.

[10] A. A. O., L. K. T., and O. O. F. W., "Movie Recommendation system with sentiment analysis using deep learning algorithms," *Egyptian Informatics Journal*, vol. 33, pp. 1-7, Mar. 2026, doi: 10.1016/j.eij.2026.100905.

[11] X. Chen, P. Cong, and S. Lv, "A Long-Text Classification Method of Chinese News Based on BERT and CNN," *IEEE Access*, vol. 10, pp. 34046–34057, 2022, doi: 10.1109/access.2022.3162614.

[12] L. Khan, A. Amjad, K. M. Afaq, and H.-T. Chang, "Deep Sentiment Analysis Using CNN-LSTM Architecture of English and Roman Urdu Text Shared in Social Media," *Applied Sciences*, vol. 12, no. 5, pp. 1-18, Mar. 2022, doi: 10.3390/app12052694.

[13] H. Zhou, "Research of Text Classification Based on TF-IDF and CNN-LSTM," *Journal of Physics: Conference Series*, vol. 2171, no. 1, pp. 1-9, Jan. 2022, doi: 10.1088/1742-6596/2171/1/012021.

[14] A. Mohammed and R. Kora, "An effective ensemble deep learning framework for text classification," *Journal of King Saud University - Computer and Information Sciences*, vol. 34, no. 10, pp. 8825–8837, Nov. 2022, doi: 10.1016/j.jksuci.2021.11.001.

[15] C.-H. Chen, S. J. H. Yang, J.-X. Weng, H. Ogata, and C.-Y. Su, "Predicting at-risk university students based on their e-book reading behaviours by using machine learning classifiers," *Australasian Journal of Educational Technology*, pp. 130–144, 2021, doi: 10.14742/ajet.6116.

[16] B. P. Yadav, S. Ghate, A. Harshavardhan, G. Jhansi, K. S. Kumar, and E. Sudarshan, "Text categorization Performance examination Using Machine Learning Algorithms," in *IOP Conference Series: Materials Science and Engineering*, vol. 981, no. 2, pp. 1-6, Dec. 2020, doi: 10.1088/1757-899x/981/2/022044.

[17] E. F. Ayetiran, "Attention-based aspect sentiment classification using enhanced learning through cnn-Bilstm networks," *Knowledge-Based Systems*, vol. 252, 2022, doi: 10.1016/j.knosys.2022.109409.

[18] C. M. Suneera and J. Prakash, "Performance Analysis of Machine Learning and Deep Learning Models for Text Classification," in *2020 IEEE 17th India Council International Conference (INDICON)*, Dec. 2020, pp. 1–6, 10.1109/indicon49873.2020.9342208.

[19] B. A. Muhammad, R. Iqbal, A. James, D. Nkantah, N. N. Hla, and T. M. Aung, "Comparative Performance of Machine Learning Methods for Text Classification," in *2020 International Conference on Computing and Information Technology (ICCIT-1441)*, 2020, pp. 1–5, doi: 10.1109/iccit-144147971.2020.9213788.

[20] A. Alqahtani *et al.*, "An efficient approach for textual data classification using deep learning," *Frontiers in Computational Neuroscience*, vol. 16, 2022, doi: 10.3389/fncom.2022.992296.

[21] N. R. Rajalakshmi, S. Saravanan, and A. Singha, "Surplus Data Prediction and Classification of Textual-Data Using Machine and Deep Learning Comparative Analysis," in *2023 International Conference on Communication, Security and Artificial Intelligence (ICCSAI)*, Nov. 2023, pp. 329–334, doi: 10.1109/iccsai59793.2023.10421287.

[22] M. N. Asim, M. U. G. Khan, M. I. Malik, A. Dengel, and S. Ahmed, "A Robust Hybrid Approach for Textual Document Classification," in *2019 International Conference on Document Analysis and Recognition (ICDAR)*, 2019, pp. 1390–1396, doi: 10.1109/icdar.2019.00224.

[23] B.-M. Hsu, "Comparison of Supervised Classification Models on Textual Data," *Mathematics*, vol. 8, no. 5, pp. 1-16, May 2020, doi: 10.3390/math8050851.

[24] A. R. N. K. N. Nagajothi, "Analysing performance of text classification models for sentiment analysis of movie reviews," *International Journal of Scientific & Technology Research*, vol. 9, no. 2, pp. 1–5, 2020.

[25] M. Parmar and A. Tiwari, "Enhancing Text Classification Performance using Stacking Ensemble Method with TF-IDF Feature Extraction," in *2024 5th International Conference on Mobile Computing and Sustainable Informatics (ICMCSI)*, Jan. 2024, pp. 166–174, doi: 10.1109/icmcsi61536.2024.00031.

[26] Y. I. Alzoubi, A. E. Topcu, and A. E. Erkaya, "Machine Learning-Based Text Classification Comparison: Turkish Language Context," *Applied Sciences*, vol. 13, no. 16, pp. 1-21, Aug. 2023, doi: 10.3390/app13169428.

[27] B. Ge, C. He, H. Xu, J. Wu, and J. Tang, "Chinese News Text Classification Method via Key Feature Enhancement," *Applied Sciences*, vol. 13, no. 9, pp. 1-13, Apr. 2023, doi: 10.3390/app13095399.

[28] D. Endalie and G. Haile, "Automated Amharic News Categorization Using Deep Learning Models," *Computational Intelligence and Neuroscience*, no. 1, Jan. 2021, doi: 10.1155/2021/3774607.

[29] S. Zhang and C. Zhao, "Machine translationese of large language models: Dependency triplets, text classification, and SHAP analysis," *PLOS One*, vol. 21, no. 1, pp. 1-19, Jan. 2026, doi: 10.1371/journal.pone.0339769.
[30] X. Liu, "A Hybrid BERT-ALBERT Model for Text Classification: Improving Accuracy in Document Analysis," *Informatica*, vol. 50, no. 7, Feb. 2026, doi: 10.31449/inf.v50i7.8368.
[31] Y. C. Yew, T. G. Cheng, D. C. W. Sen, B. J. Wei, and Z. A. A. Salam, "Text classification with Naïve Bayes," *Journal of Applied Technology and Innovation*, vol. 5, no. 2, 2021, doi: 10.65136/jati.v5i1.210.
[32] P. K. Myakala, C. Bura, and A. K. Jonnalagadda, "Artificial Immune Systems: A Bio-Inspired Paradigm for Computational Intelligence," *Journal of Artificial Intelligence and Big Data*, vol. 5, no. 1, pp. 1–13, Jan. 2025, doi: 10.31586/jaibd.2025.1233.
[33] W. Zhang, C. Jiao, and Q. Zhou, "Firefly algorithm with multiple learning ability based on gender difference," *Scientific Reports*, vol. 15, no. 1, Aug. 2025, doi: 10.1038/s41598-025-09523-9.
[34] R. Priyadarshi and R. R. Kumar, "Evolution of Swarm Intelligence: A Systematic Review of Particle Swarm and Ant Colony Optimization Approaches in Modern Research," *Archives of Computational Methods in Engineering*, vol. 32, no. 6, pp. 3609–3650, Mar. 2025, doi: 10.1007/s11831-025-10247-2.
[35] M. Sakib, T. Siddiqui, S. Mustajab, R. M. Alotaibi, N. M. Alshareef, and M. Z. Khan, "An ensemble deep learning framework for energy demand forecasting using genetic algorithm-based feature selection," *PLOS ONE*, vol. 20, no. 1, pp. 1-28, Jan. 2025, doi: 10.1371/journal.pone.0310465.
[36] A. Elnagar, R. Al-Debsi, and O. Einea, "Arabic text classification using deep learning models," *Information Processing & Management*, vol. 57, no. 1, Jan. 2020, doi: 10.1016/j.ipm.2019.102121.
[37] A. Fesseha, S. Xiong, E. D. Emiru, M. Diallo, and A. Dahou, "Text Classification Based on Convolutional Neural Networks and Word Embedding for Low-Resource Languages: Tigrinya," *Information*, vol. 12, no. 2, pp. 1-17, Jan. 2021, doi: 10.3390/info12020052.
[38] D. Ö. Şahin and E. Kılıç, "Two new feature selection metrics for text classification," *Automatika*, vol. 60, no. 2, pp. 162–171, Apr. 2019, doi: 10.1080/00051144.2019.1602293.

APPENDIX

Table 1. Literature review on textual data classification

Ref.	Algorithm/model	Domain/focus	Key contribution	Performance
[11]	BERT+CNN	Long-text classification	Combines BERT to perform contextual embeddings with CNN for convolutional layers for classifying the Chinese news text.	High accuracy (98.98%).
[12]	CNN+LSTM	Short-text classification	Combines CNN and LSTM to obtain the dependency features and analyzed using several ML techniques for text classification.	SVM obtained 90.4% accuracy,89.5% precision, 90.3% recall, and 89.8% F1-scores.
[13]	CNN+LSTM	Short-text classification	Applied TF-IDF for feature extraction with high weights and analyzed using CNN+LSTM. Compared the classification performance with CNN, LSTM, LSTM-attention models for text classification.	CNN+LSTM obtained 88.08% accuracy and 88.08% F-score using Taobao review dataset. THUCNews obtained 98.8% accuracy, and 98.79% F-score.
[14]	CNN, LSTM, and GRU	Short-text sentiment classification	Applied ensemble method for text classification that combines DL models such as LSTM, CNN, and GRU.	The ensemble method obtained higher accuracy of 96.31% accuracy which is higher compared to voting and stacking methods.
[15]	LR, GNB, SVM, DT, RF, and neural networks	Short-text classification	Predicted risky student by analyzing the reading behaviors of users in e-book using ML classifiers.	LR had the best average predictions with various evaluation metrics for predicting student academic achievement.
[16]	ANN	Short-text classification	Analyzed the working of diverse ML techniques with neural network algorithms for text classification	ANN obtained higher performance compared to SVM, RF, and NB.
[17]	Attention-based CNN and BiLSTM	Long-text classification	CNN is used to extract features containing high-level semantic and BiLSTM is used to obtain feature representation for classifying topic and sentiments.	Obtained 83.19% accuracy for classification using CNN-BiLSTM algorithm.
[18]	LR, Bi-channel CNN	Long-text classification	Several classifiers are used to analyze the text classification.	Bi-channel CNN performs better compared to other techniques.
[19]	CNN, SVM, LR, RF and NB	Short-text classification	Analyzed the performance of different techniques for classifying 6000 instances.	CNN obtained 85% accuracy and outperforms others for multi-class classification.
[20]	LR, RF, KNN, LSTM, ANN, and GRU	Long-text classification	Performance of diverse classifiers are analyzed for text classification.	LSTM outperforms others with 92% accuracy.
[21]	Bi-LSTM, RWA, CRF, NB, GB, and SVM	Long-text classification	Compared several classifiers for text classification and evaluated using performance metrics.	Bi-LSTM obtained 98.5% accuracy and outperforms others.
[22]	CNN	Short-text classification	Integrated CNN with filter-based selection for text classification using BBC news dataset and newsgroup dataset.	Combined CNN obtained higher performance with 7.7% margin in 20 Newsgroups and 6.6% margin in BBC news dataset.

Table 1. Literature review on textual data classification *(continued)*

Ref.	Algorithm/model	Domain/focus	Key contribution	Performance
[23]	GB, MLP, NB, RF, AB, SVM, DT, and LR	Short-text classification	Compared eight different supervised learning techniques for textual data analysis.	SVM obtained 95% area under curve, 84% ROC, 97% accuracy and outperformed others.
[24]	RF and NB	Short-text classification	Performed sentiment analysis in movie reviews to find the best, worst, moderate films using RF and NB and evaluated using accuracy, precision and recall.	RF obtained 85% accuracy, F-score, precision, and recall, and outperforms NB.
[25]	LR, DT, MNB, SVM, KNN, and (RF+GB)	Short-text classification	Proposed the stacking ensemble method named RF+GB which combines RF and GB classifiers with TF-IDF for feature extraction based on weights to analyze customer information of amazon users.	RF+GB attained 85% accuracy, 82% precision, 87% recall, 84% F1-score which is higher compared to others.
[26]	SVM, NB, LSTM, RF, and LR	Long-text classification	Five classifiers are used to analyze the Turkish text.	LSTM obtained 85% accuracy and outperforms others for classifying non-English text.
[27]	KFE-CNN	Short-text classification	Proposed the integrated KFE with CNN to analyze the news text for classification.	It achieved 97.84% average accuracy and 98% F1-score.
[28]	MLP, SVM, DT, XGBoost, RF, and CNN	Long-text classification	Text classification using Amharic language is handled using fastText to extract sematic vectors and CNN for feature extraction and classification.	CNN achieved higher accuracy of 93.79% and outperforms other techniques.
[29]	Multinomial NB with TF-IDF	Short-text classification	Combines multinomial NB to perform news classification with TF-IDF for feature extraction.	Medium performance with accuracy (77.39%).
[30]	BERT+ALBERT	Long-text classification	BERT is used to perform contextual embedding and ALBERT for parameter-sharing strategies.	96.6% accuracy, 95.7% precision, 95.1% recall, and 95.5% F1-score.
[31]	Tree-based models, boosting algorithms, linear models, probabilistic classifiers, discriminant analysis, SVM, nearest neighbors, and neural networks	Short-text classification	Applied large language models using dependency triplet features and evaluated using various ML techniques.	SVM achieved 93% F1-score which is higher compared to others.

BIOGRAPHIES OF AUTHORS

Geetha Jayabalan received the Master of Philosophy in Computer Science from Bharathiar University, Coimbatore in 2014 and master's degree in computer applications from Bharathidasan University, Tiruchirappalli in 2001. Currently she is working as Information Scientist, Library, Avinashilingam Institute for Home Science and Higher Education for Women, Coimbatore and pursing Ph.D. in Computer Applications from Hindusthan College of Arts and Science, behind Nava India, Coimbatore. Her research interests include application of machine learning, deep learning, and natural language processing algorithms for real time application. She can be contacted at email: jgeethacbe@gmail.com.

Kavitha Venkatesh is currently working as an Associate Professor, Department of Computer Science with Cognitive Systems, Sri Ramakrishna College of Arts and Science, Coimbatore, Tamilnadu, India. She obtained her Ph.D. degree in 2014 in the broad area of data mining. She has over 19 years of teaching and research experience. She has been supervising research scholars, post graduate end under graduate students in the recent areas of cyber security. She has published more than 75 research papers in international as well as national journals, international conferences and book chapters. She has delivered technical talk in the areas of big data analysis and cyber security. Cyber security, IoT security, artificial intelligence, machine learning, and deep learning are some of her research interests. She can be contacted at email: kavithahicas@gmail.com.