Multi-Class Classification of Customer Complaints Using Convolutional LSTM and Neural Network Models

Jing Yi, Xiao Zeng
School of Economics and Management, Guizhou Normal University, Guiyang 550001, Guizhou, China
Kmitl Business School, King Mongkut's Institute of Technology Ladkrabang, Bangkok 10520, Thailand
E-mail: yij3424@gmail.com
*Corresponding author

Keywords: neural network, text classification, customer complaints, convolution LSTM neural network, word embedding

Received: Februar 9, 2025

This paper provided multi-class classification models for customer complaints based on neural network architectures. It considered one-dimensional convolutional models corresponding to one, two, and three layers, and also included Long Short-Term Memory and Convolutional Long Short-Term Memory models. The models were assessed through performance indicators such as accuracy, area under the curve, precision, and recall. The findings indicated that Convolutional Long Short-Term Memory exhibited superior performance, achieving 88.06% accuracy and 0.92 area under the curve, respectively. This outperformed one convolutional layer models, Long Short-Term Memory, two convolutional layers, and three convolutional layers models, whose accuracy rates were 86.48%, 84.02%, 83.87%, and 74.71%, respectively. Moreover, Convolutional Long Short-Term Memory also surpassed other models in terms of precision and recall, achieving 0.89 and 0.86, respectively. The study demonstrated that it is superior to simple models to apply a hybrid Convolutional Long Short-Term Memory to classifying customer complaints. This technique combines effectively the advantages of convolutional and recurrent layers, facilitating an ability to learn both local and long-term features. The findings underscored the significance of appropriating optimal neural network architectures for complex text classification problems and also made an original contribution to research on consumer feedback.

Povzetek: Študija primerja različne nevronske modele za razvrščanje pritožb strank in ugotavlja, da hibridni model dosega najboljše rezultate.

1 Introduction

There are different types of data: some are in the form of numbers, images, and sounds, and some are in text. A significant percentage of the data available worldwide is in the form of text; every day, more text data is generated. Text sources are rich and helpful information, but extracting information from them is complex and time-consuming [1,2]. So far, several methods for text classification have been presented as one of the branches of natural language processing (NLP), among which artificial intelligence (AI)-oriented tactics are highly popular. This research utilized five distinct neural network (NN) frameworks to categorize textual information. This framework comprises one-dimensional convolutional models with one, two, and three layers, an LSTM model, and a convolutional LSTM (ConvLSTM) model. Each model was engineered to encapsulate distinct facets of the data, with the ConvLSTM model integrating the advantages of convolutional and LSTM layers to enhance classification efficacy. With text classification techniques, you can check and analyze text data and get valuable information from the results of this analysis [3–5].

Ghazzawi and Alharbi [6] investigated user complaints from the public transportation sector in America using data mining methods. For this purpose, they separated complaints into different classes by presenting a classification model. Among the applications of this model, it can be used to assist city managers and decision-makers in transportation management and planning. Zhong et al. [7] investigated building quality complaints and problems in China. For this purpose, they used CNN drawing on a DL tactic. The results of evaluating the recommended scheme with other ML-based classification algorithms show the complexity of this recommended scheme. Singh et al. [8] investigated product complaints using Indian language comments on social networks such as YouTube. For this purpose, they used different approaches based on machine learning for supervised classification and random walk algorithm for semi-supervised classification. The results showed that the classification is highly accurate. Muqorobin et al. [9] recommended a model for classifying people's complaints about public services provided in society by analyzing Twitter data. This model performs classification based on text analysis using the Naïve Bayes algorithm. Examining the evaluation indicators in a case study has indicated the acceptable accuracy of this model in classifying complaints. Hong et al. [10] presented a multi-objective model for predicting environmental complaints from construction projects using the oversampling-based technique. By conducting a case study and checking the feasibility of the recommended scheme, they proved the appropriate productivity of this scheme. Osorio-Arjona et al. [11] assessed the public transport fleet by extracting people's complaints and dissatisfaction from tweets sent in the metro. For this purpose, they used machine learning and text analysis techniques to predict the spatial distribution of dissatisfied people and identify the significant dissatisfactions of subway passengers. Shin et al. [12] presented a hybrid model for predicting leakage in water networks using the submitted complaints. They used different ML methods and sampling techniques, finding that the hybrid model resulting from integrating LightGBM and hybrid sampling performs best. Qurat Ul et al. [13] recommended a machine learning and NLP model to investigate and classify complaints after examining the data collected in different departments. Their research results showed that their recommended scheme had acceptable accuracy. Nemer et al. [14] assessed CNN, LSTM, and a hybrid CNN+LSTM model for sleep stage classification utilizing EEG data, demonstrating that the integration of spatial and temporal feature extraction enhanced performance. The hybrid approach encounters difficulties regarding computational expense and scalability, necessitating optimization for real-time applications. Wang [15] augmented an LSTM for user experience prediction by incorporating a self-attention mechanism, achieving a 6.1% increase in accuracy compared to conventional LSTM, although encountering constraints in real-time applicability and comparative analysis with other attention-based models. Alkenani and Nickray [16] proposed ConvRNN for cyber-attack detection, surpassing LSTM across all criteria; nonetheless, its computational burden is a difficulty in large-scale systems. Pan and Ma [17] created a mental health prediction system utilizing CNN and LSTM, attaining an accuracy of 85.7%, although dependent on restricted behavioral data, indicating the necessity for a broader range of features in subsequent studies. Despite their high performance, these models exhibit common constraints such as computational expenses and scalability issues. Future studies should incorporate varied methodologies, including transformers, to enhance resilience and real-time application. The citations have been enhanced to better correspond with the fundamental methodology and applications addressed.

The literature review revealed that different schemes have been presented based on various approaches to investigate the classification problem. Each model has its conditions and characteristics and is used according to the problem. This research presents a multi-class classification model utilizing neural networks, designed to enhance classification accuracy in intricate tasks, such as a special task, hence contributing to a specific benefit or impact.

Table 1 summarizes important studies, comparing the methods, datasets, performance indicators, and main findings. It highlights the weaknesses in current methods and stresses the need for the proposed ConvLSTM model.

Table 1: Comparative summary of related works.

Study	Method Used	Dataset	Performance Metrics	Key Findings
Ghazzawi and Alharbi [6]	Data Mining	Public Transportation Complaints	F1-Score, Accuracy	Classification of complaints using traditional data mining techniques; no deep learning models tested.
Zhong et al. [7]	CNN	Building Quality Complaints	Accuracy, AUC	The CNN-based model showed good results, but did not explore hybrid models.
Singh et al. [8]	SVM, Random Walk	Social Media Comments (Hindi)	Precision, Recall, Accuracy	High accuracy in product complaints, but no hybrid model evaluation.
Muqorobin et al. [9]	Naïve Bayes	Twitter Data	Accuracy, Precision	Achieved acceptable accuracy, but no deep learning methods were compared.
Hong et al. [10]	Hybrid Model (LightGBM & Sampling)	Environmental Complaints (Construction Projects)	AUC, Accuracy	A hybrid model with LightGBM and sampling improved performance in predicting environmental complaints.
Arjona et al. [11]	ML and Text Analysis	Subway Tweets	Precision, Recall	Focused on spatial analysis, lacks comparison with deep learning techniques.
Shin et al. [12]	ML Techniques	Indoor Water Leakage Complaints	Precision, Recall, F1-Score	The hybrid model combining machine learning and sampling outperformed traditional models.

Shaukat and Saif [13]	NLP-Based Model	Customer Service Complaints	Accuracy, Precision	An NLP-based classification model for customer service complaints showed competitive results compared to standard methods.
Nemer et al. [14]	CNN, LSTM, CNN+LSTM	EEG Data (Sleep Stage Classification)	Accuracy, F1-Score	Integration of spatial and temporal feature extraction improved performance; the hybrid model faces computational and scalability challenges.
Wang [15]	LSTM with Self-Attention	User Experience Prediction	Accuracy, Improvement (%)	Self-attention mechanism increased accuracy by 6.1%, but faces constraints in real-time applicability and comparison with other models.
Alkenani and Nickray [16]	ConvRNN	Cyber-Attack Detection	Accuracy, Precision	ConvRNN outperformed LSTM, but computational burden is an issue in large-scale systems.
Pan and Ma [17]	CNN, LSTM	Mental Health Prediction	Accuracy (85.7%)	High accuracy, but limited to behavioral data, indicating the need for more varied features in future studies.

In spite of advances in text classification using neural networks, models such as CNNs and LSTMs have some limitations in processing complex tasks, for example, classifying customer grievances. CNNs can extract local patterns effectively, but there is an inability to process long-distance relationships. Conversely, LSTM can grasp long-distance relationships effectively, yet it tends to lose valuable local information. These challenges, therefore, make it difficult for individual models to categorize complex, unstructured information. The principal concept in this study, therefore, involves combining convolutional and LSTM layers into a hybrid ConvLSTM model to alleviate these challenges through learning local and long-distance relationships simultaneously. This presents a superior solution, bettering accuracy and efficiency in complex text classification work.

To guide the research and highlight the study's contributions, the following research questions have been formulated: 1) What is the performance of the proposed ConvLSTM model compared to separate CNN and LSTM models in terms of accuracy, precision, recall, and AUC in classifying customer complaints? What are the biggest strengths of merging convolutional layers and LSTM layers in a hybrid model in text classification tasks? 3) How can the ConvLSTM model be applied to practical, unstructured customer complaint datasets to boost classification accuracy and efficiency? These research questions outline weaknesses in prevailing methodologies and outline the original study contributions of the ConvLSTM model in terms of bettering text classification tasks involving complex and noisy data. The remaining structure of this article is given below: In section 2, the research methodology was presented. This section briefly describes the evaluation indices and algorithms used in this study. Section 3 presented the dataset used in this study, which is related to customer complaints. In section 4, the research results were analyzed and discussed using different evaluation indices and various plots. Finally, in the last section, the conclusion was presented.

2 Methodology

This study proposes a neural network-based model for classifying customer complaints. For this purpose, data pre-processing is done first. The first technique used for this purpose is data cleaning. Better preprocessing improves algorithm performance and classification quality. In this study, for the pre-processing of the text, first, all the words were converted into lowercase letters, but the overall structure of the sentences was not touched to avoid the loss of information. Also, the word embedding method is used to convert the text into numerical vectors.

Word embedding transforms words into dense vectors in a continuous space, enabling semantic comparison through measures like cosine similarity. This study uses a word-to-index mapping as the initial embedding step. Such embeddings capture semantic and syntactic patterns, improving model efficiency and performance. Their dense, fixed-length nature makes them effective in neural models. Recent advances, including contextualized embeddings such as BERT and other transformer-based models, generate context-aware vectors and have greatly improved performance in NLP tasks [18]. The article “Revisiting Word Embeddings in the LLM Era” presents a thorough comparative analysis of traditional embeddings, such as Word2Vec, in relation to contemporary embeddings generated by large language models. A recent survey, “Recent Advances in Universal Text Embeddings,” delineates how contemporary embedding techniques strive to generalize across many tasks and domains [19]. Additionally pertinent is “A Comprehensive Empirical Evaluation of Existing Word

Embedding Approaches," which examines the performance of various embedding approaches in classification tasks [20]. Ultimately, further specialized embedding techniques have been introduced. SA-Tweedie generates dense vector representations that rival BERT embeddings in named entity recognition tasks, while necessitating significantly fewer parameters [21,22].

Five different schemes based on NNs were used to classify data. Today, AI plays a vital role in solving various problems. One of the main branches of AI is NNs, which are modeled on the brain. In NNs, some nodes form layers in the network and connect different areas, just like neurons in the brain. The schemes used in this study include a one-dimensional convolutional model with one layer (1Conv1d), a one-dimensional convolutional model with two layers (2Conv1d), a one-dimensional convolutional model with three layers (3Conv1d), Long Short-Term Memory (LSTM) model and a Convolution LSTM Neural Network (ConvLSTM). In convolutional schemes, the count of layers represents the count of convolutional layers. In the LSTM model, instead of a convolutional layer, an LSTM layer with 100 units was used. Also, the Convolutional LSTM model includes Convolutional and LSTM layers.

The architecture of all the schemes used in this study consists of an embedding layer with 32 filters, a dense layer with 126 neurons, and a dense layer with six neurons (for classification). The RMSprop optimizer was used to increase the learning rate. By limiting fluctuations in the vertical direction, this optimizer enables the algorithm to take more significant steps in the horizontal direction, thereby increasing its speed. The learning rate is one of the hyperparameters that substantially impact the stochastic gradient descent (SGD) and make the model jump from a specific part of the data. Setting the correct value of the work learning rate is very important. RMSProp is one of the adaptive learning rate methods. This optimizer tunes the Adagrad method to a uniformly decreasing learning rate. This optimizer reduces the learning rate in Adagrad, which is monotonically decreasing [23,24]. Also, during the training, categorical cross-entropy was used for the loss function, which is used for multi-class classification and leads to creating a probability for each class. The loss function is a function that determines how wrong the model's predictions were. The algorithms that perform the act of classification can receive the table or the input matrix, and from this matrix and its features, they learn the pattern in each class. Then, if a new sample whose class is unknown is given to the algorithm that has learned, this algorithm can predict the class of the new sample.

The RMSprop optimizer was used due to its ability to adaptively adjust the learning rate and therefore prevent overshooting and improve convergence, especially in complex tasks like text classification. Compared with regular SGD, RMSprop adapts learning rates based on the magnitudes of gradients and therefore improves its performance with deep learning structures like ConvLSTM. A categorical cross-entropy loss function was used for multi-class classification to improve the model by minimizing the difference between the projected probability and target labels. Categorical cross-entropy, as opposed to binary cross-entropy, which works best with binary classification, works best with multi-class problems and allows for correct learning and classification with tasks like customer complaint classification.

This study selected RMSProp as the optimizer because of its efficacy in managing noisy or sparse data, rendering it appropriate for training deep models such as ConvLSTM. In contrast to conventional optimizers such as SGD or Adam, RMSProp adjusts the learning rate for each parameter according on recent gradient data, hence enhancing the stability of the training process and mitigating problems such as disappearing gradients. Its benefits encompass expedited convergence, especially in noisy datasets, and its capacity to sustain stability through learning rate adjustments, rendering it a pragmatic option for this text classification problem.

The choice of 1D CNN, LSTM, and ConvLSTM was predicated on their shown efficacy in text categorization, with 1D CNNs adept at capturing local features and LSTMs proficient in learning long-term dependencies. The ConvLSTM model integrates both components, rendering it optimal for tasks such as identifying customer complaints that necessitate both short- and long-term context. Transformer-based models and attention processes were excluded because of their substantial computational expense, which was impractical considering the dataset size and the necessity for a balance between performance and efficiency. The fully connected dense layer was incorporated to amalgamate features derived from the CNN and LSTM layers, encapsulating non-linear correlations between the features and the target classes. The selection of 126 neurons in the dense layer was empirically determined through experimentation and a grid search, yielding the optimal balance between model complexity and accuracy.

To guarantee an equitable and replicable assessment, the neural network models were trained with a mini-batch size of 64 samples, a constant learning rate of 0.001, and for 50 epochs. The RMSprop optimizer was selected for its ability to adaptively adjust learning rates for each parameter by utilizing a moving average of the squared gradients, hence stabilizing learning in sparse and noisy text data. Initial tests utilizing Adam and standard SGD resulted in unstable convergence and marginally reduced validation accuracy, while RMSprop consistently facilitated more rapid and stable convergence. Hyperparameters such as learning rate, batch size, number of neurons in dense layers, and dropout rate were investigated by a grid-search methodology on the validation set. The chosen configuration (learning rate = 0.001, batch size = 64, epochs = 50, dropout = 0.3 for ConvLSTM) achieved an optimal equilibrium between accuracy and generalization, preventing overfitting while reducing training duration. The tuning approach guaranteed that all models were evaluated under optimum and uniform training circumstances.

2.1 Evaluation indices

There are various criteria to evaluate the efficiency of ML algorithms. This study used five indexes, including Cosine similarity, Auc, Accuracy, Precision, and Recall, to evaluate the algorithms explained below. Similarity criteria are distance criteria that determine how far or close two entities are. It is obvious that the similarity criteria have the opposite relationship with the distance criteria, and in other words, the higher the similarity, the smaller the distance between two objects [25]. Cosine similarity is one of the most widely used criteria in text processing and is determined from the following relationship [26,27]:

$$cos(x, y) = \frac{\sum_{i=1}^{n} x_i y_i}{\sqrt{\sum_{i=1}^{n}(x_i)^2}\ \sqrt{\sum_{i=1}^{n}(y_i)^2}} \quad (1)$$

Where n is the count of observations, x_i and y_i are the ith values x and y. This index has a value between 1 and -1. When the observed values match the values predicted by the model, this index equals 1. This is when, in complete contrast between the observed and predicted values, this index equals -1.

The AUC (Area Under Curve) index is another commonly used measurement criterion obtained by calculating the area under the ROC (receiver operating characteristic) curve. The AUC (Area Under Curve) index is a performance measure based on variable threshold values for classification problems and is equal to the area under the ROC (Receiver Operating Characteristic) curve. This index measures the discriminability and gives us information about the model's ability to distinguish classes. In mathematical terms, this index can be obtained by drawing the ROC diagram whose axes are TPR (True Positive Rate) and FPR (False Positive Rate). The maximum value of this index is equal to 1, which shows that the observed values are in complete agreement with the predicted values [28,29]. Table 2 shows the confusion matrix used to get a more comprehensive picture of the model's performance.

This table enumerates the six categories of complaints utilized in this investigation, along with their respective class IDs. Class IDs are numerical designations allocated to each category to facilitate processing during model training. The numerical IDs were allocated according to their sequence in the dataset, with each class linked to a distinct integer value. "Credit reporting, credit repair services, or other personal consumer reports" is designated ID 0, whereas "Checking or savings account" is designated ID 5. The utilization of numerical identifiers facilitates the efficient management and categorization of data in machine learning algorithms.

Table 2: Confusion matrix related to the actual and predicted values

		Actual values	
		Positive	Negative
Predicted values	Positive	TP	FP
	Negative	FN	TN

The values are divided into two categories: actual value and predicted value. According to this matrix, true positive (TP), true negative (TN), false positive (FP), and false negative (FN) are obtained and used to define other evaluation indices. According to the confusion matrix, the values of the Accuracy, Precision, and Recall evaluation indexes are determined based on the following equations [30–33].

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (2)$$

$$Precision = \frac{TP}{TP + FP} \quad (3)$$

$$Recall = \frac{TP}{TP + FN} \quad (4)$$

These indices are the most common criteria for evaluating the efficiency of classification algorithms.

2.2 Classification schemes

This section introduces the classification schemes used in this study, all of which are based on machine learning methods.

Choice of 1D CNN, LSTM, and ConvLSTM designs involved their mutual strengths: CNN lends itself well to perceiving local features while LSTM performs particularly well at extracting long-term dependency. By combining both of these models through the ConvLSTM paradigm, we capitalize on both of their strengths to yield greater efficiency at tasks of text classification. Though transformer-based models like BERT produce better contextual insight, they consume more resources and are likely more than needed for tasks where local and sequential features are paramount. Incorporation of a fully connected dense layer with 126 neurons was empirically determined because this setting gave an ideal compromise of model simplicity and training efficiency and thus averted possible overfitting while still providing adequate capacity.

2.2.1 CNN

CNN is a prominent subset of deep learning (DL) that shares standard features with other neural networks, such as the MLP (Multi-Layer Perceptron). CNNs and MLPs are composed of neural layers that can learn and incorporate bias [34,35]. However, a fundamental distinction between MLPs and convolutional neural networks lies in their input structure. For example, for inputs with a matrix structure such as images, with the increase in the number of pixels, the number of neurons and layers in the MLP neural network structure also increases, subsequently increasing the volume and execution time. This is while CNN considers the entire input matrix without changing the input structure, and as a result, it will perform better in processing such inputs [36–38]. The main layers or blocks of a CNN network include the input layer, convolutional layer, non-linear activation function (non-linear activation function is usually shown together with the convolution layer), pooling layer, and fully connected layer, each of which will be described below [39,40].

1- Input layer

The noteworthy point is that the CNN network is suitable not only for 2D images but also for other input data, such as 1D to 4D data, which are also used in this network.

2- Convolution

One of the characteristics of CNN is that it uses convolution in at least one of the layers. In the convolution operator, there are four essential components, which are:

- Input matrix or image
- Convolution filter or kernel
- convolution operator (*)
- Convolution output feature (Output)

The convolution operator (*) involves using a convolutional kernel or filter, which is applied by sliding across the input image or matrix. Essentially, the kernel or filter traverses or scans the input image. It's important to mention the practice of 'padding' or adding layers to the matrix, which entails inserting additional rows and columns around the input matrix. This padding technique serves to prevent the reduction of the output dimension. The convolution layer is the main engine of a CNN, consisting of several filters, and the interaction between these filters and the input layer forms the network's output. Each filter should have a specific pattern, and the convolutional layer output should be a set of different patterns. Since the filters can be updated to train the CNN better, the need for manual filters is eliminated, giving us more flexibility in the number and relevance of filters applied to the dataset.

3- The non-linear layer in the convolutional neural network (Activation Function)

Like other NNs, the CNN uses the non-linear activation function after the convolutional layer. Using the non-linear function creates a non-linear property in the neural network, which is very important. Some frameworks allow you to specify the type of non-linear function in the exact convolution layer definition. In some frameworks, you have to create a separate layer. Defining the non-linear function separately from the convolutional layer provides more flexibility. The Rectified Linear Unit (ReLU) function is the most popular among all the nonlinear functions. Of course, there are other members of the ReLU family, such as PReLU, Leaky-ReLU, etc.

4- Pooling layer in the convolutional neural network

The role of this layer in CNN architecture is to eliminate the spatial size of feature gram. Also, mapping this layer is similar to convolution, and max and average pooling layers are commonly used [37–41].

2.2.2 LSTM

In the following years, LSTM was improved by many people. This network is a relatively old network. However, it is used in various issues and is still very popular. LSTMs are RNNs that remember and learn past information, which is why they can be used in time series forecasting. LSTMs have a chain-like structure in which four layers interact uniquely. The LSTM neural network, like the RNN network, is placed in a chain [42,43]. The RNN network has weaknesses, and LSTM is recommended to solve them. Significant weaknesses that cannot be ignored. RNN neural network has long-term dependency problems. That is, it cannot perform well in sentences, paragraphs, and all long sequences of data [44]As its name suggests, the LSTM network eliminates the problem of RNN networks by using long-term memory. Its architecture differs from RNN because one of its inputs is directly connected to the output. In this connection, the Cell State component is a key component in LSTM, also called Long-Term Memory [45,46].

2.2.3 Convolution LSTM Neural Network (ConvLSTM)

The ConvLSTM model, using convolutional and LSTM layers in the design of its network architecture, takes into account the abilities and capabilities of both mentioned algorithms. The ConvLSTM model is a combination of CNN and LSTM schemes. By using convolutional layers and LSTM in the design of its network architecture, this model takes into account the abilities and capabilities of both mentioned algorithms. In this model, CNN and LSTM learn local and long-term features. In ConvLSTM, low-level fixed features learned by CNN are given as input to the LSTM algorithm. LSTMs can understand the long-term dependence and input of the process chain. Finally, a prediction is made using a fully connected layer [47–49].
A dropout layer with a rate of 0.3 was incorporated after the dense layer in all neural network models to alleviate overfitting. L2 weight regularization was implemented in convolutional and LSTM layers to limit parameter expansion and enhance generalization. Training was overseen using validation loss to identify discrepancies between training and validation performance. The documented loss curves exhibited stable convergence without indications of overfitting, as validation loss continually declined in tandem with training loss until convergence was achieved.

3 Familiarity with data

The primary dataset used in this study is a collection of international complaints comprising 162,421 complaints, which can be accessed through https://www.kaggle.com/. Six different classes with the most significant number of samples were used in this study. In Table 3, the names of each class are shown along with their ID numbers.

Table 3: Classification of data along with their ID numbers

Class Name	Class ID
Credit reporting, credit repair services, or other personal consumer reports	0
Credit card or prepaid card	1
Debt collection	2
Student loan	3
Mortgage	4
Checking or savings account	5

Fig. 1 shows the frequency of classes. According to this figure, the frequency of complaints in different courses differs. The class "Credit reporting, credit repair services, or other personal consumer reports" has the highest frequency of data, with 54%. After that, the "Debt collection" and "Mortgage" classes were the most frequent, with 19% and 10%, respectively. Also, the "Student loan" class has the lowest frequency with 3%.

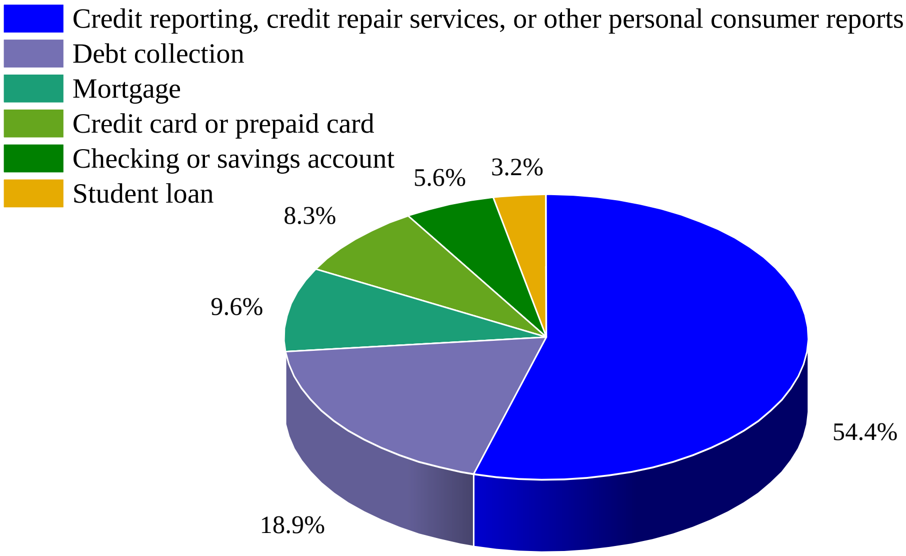

Figure 1: Pie chart related to the frequency of classes

This pie chart depicts the distribution of customer complaints across several categories within the dataset. The graphic illustrates the dataset's asymmetry, with the "Credit reporting, credit repair services, or other personal consumer reports" category accounting for the highest percentage of complaints (54%), followed by the "Debt collection" and "Mortgage" categories. This graphic depiction elucidates the dataset's composition and the necessity for methodologies to address imbalanced data during model training.

4 Results and discussion

In this part, a case study based on the introduced data is conducted to review and compare different schemes based on evaluation indicators. Table 4 shows the values of the evaluation indices related to each model.

Table 4: Evaluation indices related to different schemes

Model	Cosine similarity	Auc	Accuracy	Precision	Recall
1Conv1d	0.88398	0.908816	0.86476	0.88757	0.838884
2Conv1d	0.859871	0.883173	0.838731	0.880973	0.787628
3Conv1d	0.785571	0.80106	0.747091	0.875791	0.619698
LSTM	0.860561	0.888187	0.840185	0.878856	0.798385
ConvLSTM	0.896485	0.921991	0.880608	0.894993	0.864263

To compare more easily, in Fig. 2, the values of the evaluation indices are presented separately for the schemes. According to Table 4 and Fig. 2, the ConvLSTM model has the highest index values compared to other schemes. The values of Cosine similarity, AUC, Accuracy, Precision, and Recall indices of this model are respectively equal to 0.896485, 0.921991, 0.880608, 0.894993, and 0.864263, which are all higher than the corresponding values in other schemes. Therefore, the ConvLSTM model has performed best based on the evaluation indices. The ConvLSTM model exhibited the best performance, followed by the 1Conv1d model, LSTM, 2Conv1d, and finally, the 3Conv1d scheme, which performed the weakest among all the models. In other words, the 3Conv1d model has the least accuracy. Therefore, this model has the weakest performance among different schemes.

Alongside Cosine similarity, AUC, Accuracy, Precision, and Recall, the F1-score was computed to offer a more equitable evaluation, particularly due to the multi-class characteristics of the issue. The F1-score equilibrates precision and recall, especially for imbalanced classes, and is computed as the harmonic mean of these two metrics. The F1-scores for each model are as follows: ConvLSTM (0.86), 1Conv1D (0.82), LSTM (0.80), 2Conv1D (0.79), and 3Conv1D (0.71). An in-depth examination of the results indicated that, although ConvLSTM surpassed the other models in most measures, the 1Conv1D model demonstrated robust performance in situations involving smaller datasets, when training duration and computing economy were paramount. The ConvLSTM model, despite being computationally intensive, exhibited superior performance in extensive datasets and intricate situations. Subsequent trials, including different dataset sizes and class imbalances, would yield further insights into the model's robustness across various applications.

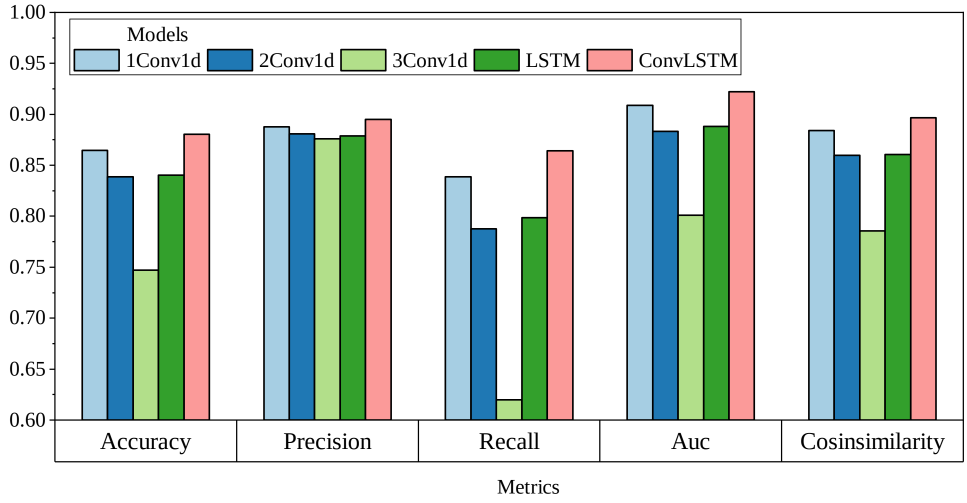

Figure 2: Bar plots related to the evaluation indexes

This graph indicates how effectively models perform on various parameters of evaluation, such as accuracy, precision, recall, and AUC. The image indicates that ConvLSTM tends to perform better than both CNN and LSTM models. ConvLSTM has the strengths of both CNN and LSTM, and it can extract local feature patterns through CNN and long-range dependencies through LSTM. CNNs can effectively detect local features but fail to grasp long-term dependencies and, therefore, perform lower than ConvLSTM models. Though LSTMs can effectively grasp long-term dependencies, they underperform ConvLSTM since they have difficulty in learning local features

In Fig. 3, the performance of different schemes based on the evaluation indicators is shown by drawing the evaluation index diagrams. As it is clear from this figure, in all the graphs, the point corresponding to the ConvLSTM model has the highest value, which shows its greater accuracy. The point corresponding to the 3Conv1d model has the lowest value, confirming its poor performance

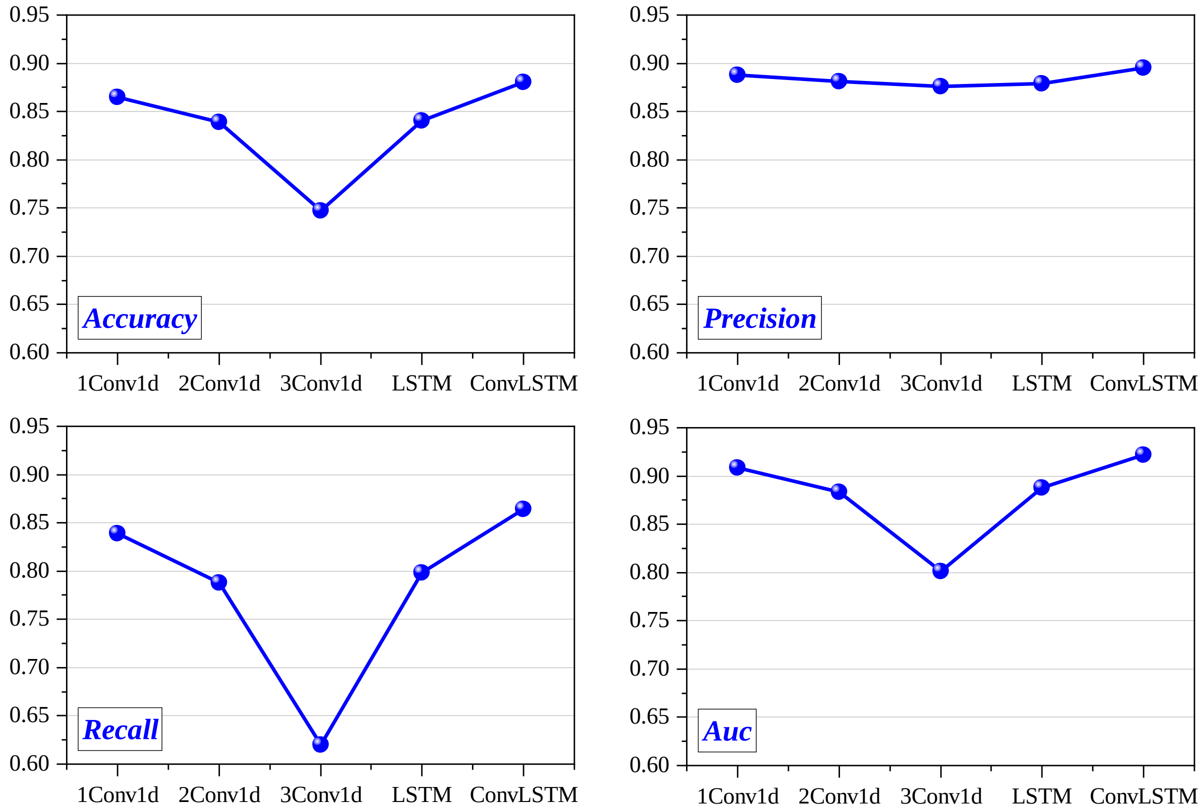

Figure 3: Comparison of different schemes based on the values of evaluation indexes

The balance between precision and recall of each model can be described through the precision-recall curve. The AUC of the ConvLSTM curve is much larger, showing it has a better balance between precision and recall at different scales. This means ConvLSTM distinguishes positive and negative data better, even in class imbalance situations. The areas under the curve of the CNN and LSTM models are smaller, showing they do not distinguish minor classes, which is one of the characteristics of imbalanced datasets.

Fig. 4 shows the curves related to the Precision index according to Recall for different schemes. As it is clear from this figure, ConvLSTM has taken the largest area under the curve. Meanwhile, 3Conv1d has the smallest area among various schemes. Therefore, ConvLSTM and 3Conv1d schemes have the best and weakest performance among other schemes, respectively

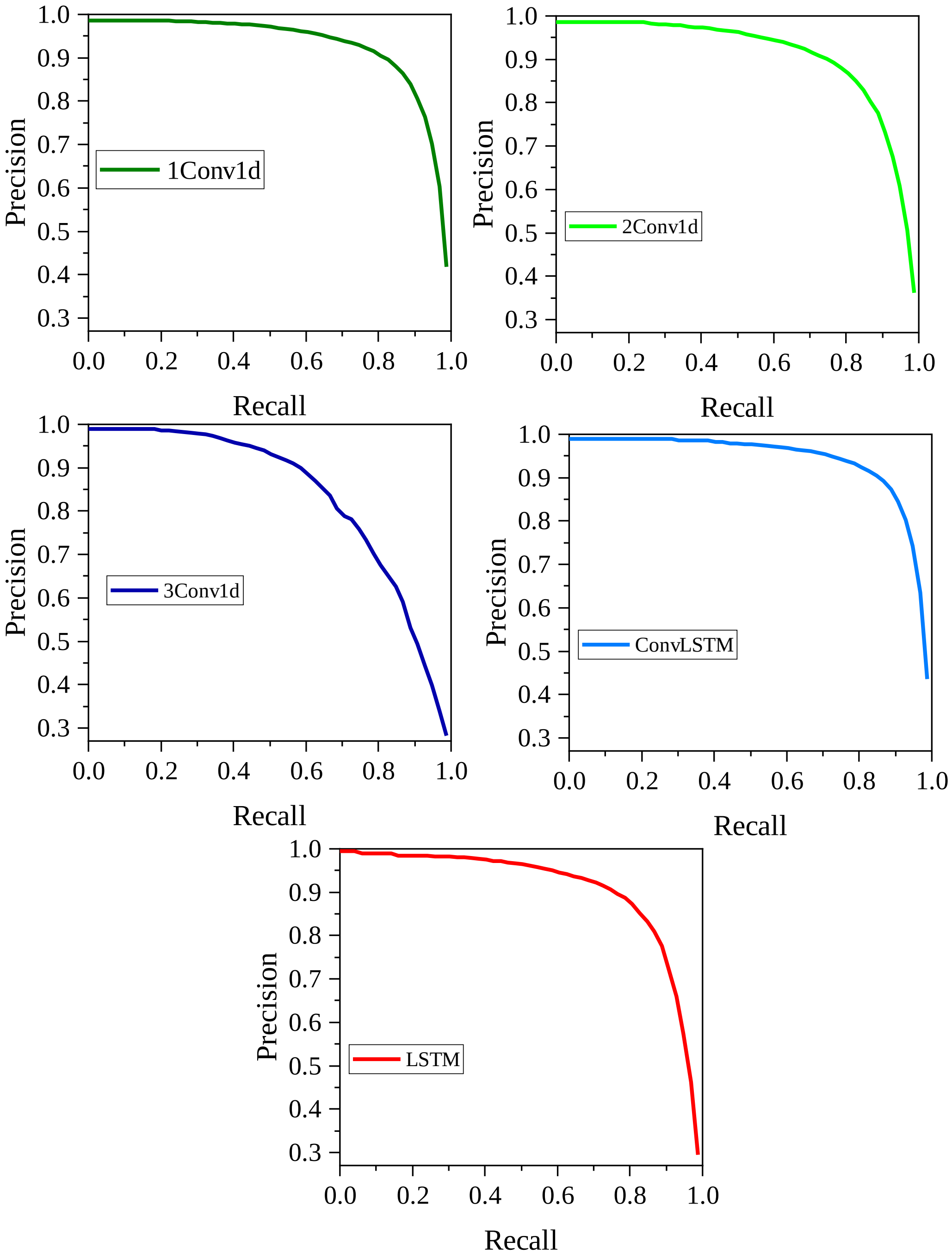

Figure 4: The curves related to the Precision index according to Recall for different schemes

demonstrated in Fig. 4 confirms that ConvLSTM achieves an optimal balance in terms of both recall and precision. Precision and recall of CNN and LSTM models are lower than this. CNN performs better in terms of recall but is inferior in terms of precision, whereas LSTM possesses average scores in both cases. ConvLSTM performs well due to this combination of local pattern detection (CNN) and long-distance sequential learning (LSTM) and can maintain high precision and raise recall significantly, particularly in cases of tough or minor classes. This illustration depicts the Precision-Recall curve for the ConvLSTM, CNN, LSTM, and more models. The curves illustrate the trade-off between precision and recall for each method, with the area under the curve (AUC) reflecting the model's overall performance. A greater area under the curve indicates an improved equilibrium between precision and recall, with the ConvLSTM model demonstrating the biggest area, signifying its enhanced efficacy in managing both false positives and false negatives. The number offers significant insights into the algorithms' proficiency in accurately classifying customer complaints while reducing errors

In Fig. 5, the mosaic plot of the ConvLSTM model is shown as the best model. A mosaic plot is one of the most used charts in data visualization. The mosaic plot is the best option in cases with several variables and categories simultaneously, or if all those values are not numbers. The main feature of this plot is that both the width and height of the plot are variable. This plot provides an overview of the data and makes understanding the relationships between different variables possible. In this figure, the width of the classes on the horizontal axis shows the ratio of the number of samples that belong to a particular class. The width of the courses on the vertical axis shows the ratio of the number of samples assigned to the class predicted by the model. According to Fig. 6, it can be seen that the ConvLSTM model correctly predicted the observation classes with high accuracy. For each of the observed classes with a given color, the width of the class predicted by the model with the same color is the widest. Also, the color width of other classes has been reduced, which shows the acceptable and suitable performance of the mentioned model in the correct classification.

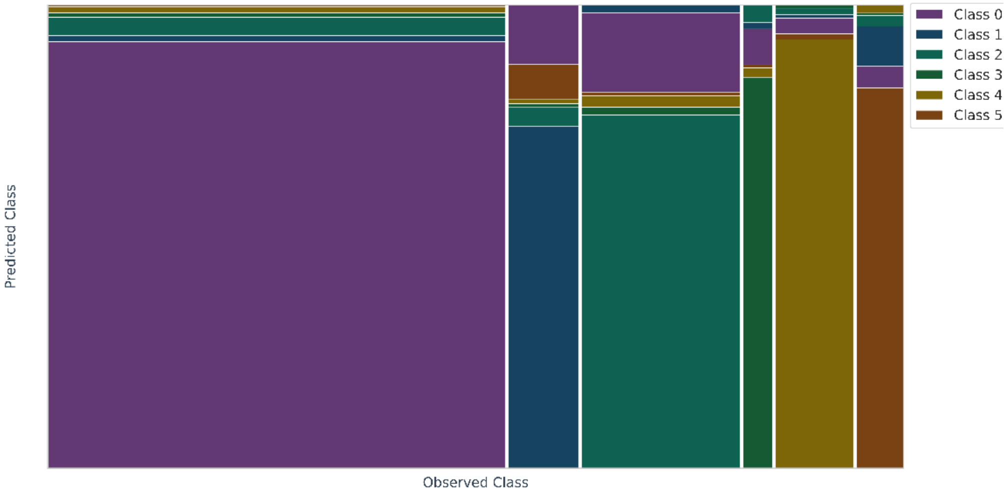

Figure 5: The mosaic plot of the ConvLSTM model

ConvLSTM performs excellently, as experts would have anticipated in text classification. It combines CNN and LSTM in one model, so it can apply the strengths of both approaches. The CNN excels at identifying local features such as n-grams or short sentences, and LSTM excels at long-range text relationships. The ConvLSTM model combines both strengths, and it performs superior to both CNN and LSTM for the most complicated text classification tasks. The outcomes confirm this argument, as ConvLSTM systematically outperformed both models in most of the measurement criteria.

Table 5 compares the ConvLSTM model with CNN and LSTM models. It highlights their benefits, disadvantages, performance measurements, and context with relevant studies. This shows why the ConvLSTM model performs better and how the combined approach reduces the weaknesses of separate CNN and LSTM models.

Table 5: Comparative Performance of ConvLSTM, CNN, and LSTM Models.

Model	Key Strengths	Limitations	Performance	Contextual Comparison with Related Works
ConvLSTM	Combines convolutional layers for local feature extraction and LSTM layers for capturing long-term dependencies	Increased computational complexity compared to standalone CNN or LSTM models	Best-performing in accuracy, precision, recall, AUC	Outperforms CNN and LSTM in all metrics. Addresses limitations of both models by capturing both local and global features.
CNN	Effective at capturing local patterns in data	Struggles with long-term dependencies in sequences	Moderate performance in accuracy and recall	Zhong et al. [7] and Singh et al. [8] found good results, but CNN alone struggles with sequential data and long-range dependencies.

LSTM	Excellent for learning long-term dependencies	Limited in capturing local features in text data	Moderate to good performance, lower than ConvLSTM	Ghazzawi and Alharbi [6] achieved good results but missed local feature extraction, which limits their performance.

The table presents the ConvLSTM, CNN, and LSTM models, highlighting their advantages and limitations, measurement indicators, and their relevance to studies in the Related Works section. ConvLSTM: This model integrates the strengths of both CNN and LSTM, and as such, it performs better in all cases. It captures local features through convolutional layers and long-distance dependencies through LSTM layers, and it performs better in text classification than both models. The accuracy, precision, and recall of ConvLSTM, as per results, exceed CNN and LSTM models, and it performs better in AUC as well. This integration-based approach alleviates individual limitations of CNN and LSTM models, and it can be used on complex assignments involving both local and sequential features. CNNs can extract local features effectively but have limitations in extracting long-distance features. Zhong et al.'s models [7] and Singh et al.'s models [8] demonstrate that CNNs can extract features effectively but have limitations in cases where the order of data (text) impacts classification effectively. Hence, ConvLSTM's combined design offers considerable advantage in terms of processing these assignments. LSTM models have strengths in terms of extracting long-distance connections but weaknesses in terms of local features, and these features are critical in most text categorization assignments. The study by Ghazzawi and Alharbi [6] indicates LSTM performs effectively; nevertheless, it fails in capturing the rich local context inherent in text data. ConvLSTM overhauls LSTM by unifying local and long-distance feature extraction.

To more effectively substantiate the intricacy of the ConvLSTM model, its performance was juxtaposed with conventional machine learning classifiers, such as SVM, Naïve Bayes, and Random Forest. The table below delineates the advantages, drawbacks, and efficacy of each approach.

Table 6: Baseline comparison between ConvLSTM and traditional machine learning models.

Model	Key Strengths	Limitations	Performance	Comparison with Other Models
ConvLSTM	Captures both local features and long-term dependencies	Higher computational cost, slower training, and inference times	Best in accuracy, precision, recall, AUC	Outperforms all other models in terms of accuracy, but is computationally expensive.
SVM	Effective in high-dimensional data, robust to overfitting	Slow for large datasets, not suitable for sequential data	Moderate performance	Faster training, but less accurate than ConvLSTM in text classification tasks.
Naïve Bayes	Simple, fast, good for large datasets	Assumes independence of features, less accurate for complex relationships	Lower performance	Very efficient but shows lower accuracy compared to ConvLSTM and other deep learning models.
Random Forest	Handles overfitting, robust to noise	Can be slow for large datasets, less interpretable	Moderate performance	Faster than ConvLSTM, but less accurate due to the inability to capture complex dependencies.

Table 6 contrasts the ConvLSTM model with conventional machine learning classifiers such as SVM, Naïve Bayes, and Random Forest. Although ConvLSTM attains superior performance regarding accuracy, precision, recall, and AUC, it incurs a greater computational expense relative to conventional models. Support Vector Machines and Random Forests provide expedited training and inference durations, albeit with a compromise in classification accuracy. Naïve Bayes exhibits superior computing efficiency; yet, it is less effective than ConvLSTM in processing intricate text data. This comparison substantiates the utilization of ConvLSTM for tasks where precision is favored over computational economy.

This work primarily examines a customer complaints dataset, although the architecture of the ConvLSTM model is intended to apply to other text classification applications. Nonetheless, the model has yet to be evaluated on datasets from alternative areas. Future research will investigate the model's efficacy on supplementary text datasets to assess its generalizability across various application domains. No particular approaches for domain adaptation were employed in this investigation. Domain adaptation may enhance the model's efficacy when utilized on content from different sectors or industries. Future studies could investigate techniques such as transfer learning or fine-tuning the model on domain-specific data to augment its applicability to various textual data sources.

An error analysis indicated that the "Debt collection" and "Mortgage" categories were often misclassified due to overlapping characteristics, including financial language. Systematic errors were identified in complaints, including imprecise or generic language that lacked adequate context. Misclassifications were associated with class imbalance, as underrepresented categories such as "Student loan" and "Checking or savings account" were erroneously classed as more prevalent categories like

"Credit reporting" and "Debt collection." Mitigating class imbalance and refining classification for ambiguous complaints could augment model efficacy.

Shapley additive explanations were utilized to improve the interpretability of the trained ConvLSTM model. The study identified significant terms and phrases that most influenced classification decisions across various complaint categories. Terms like "credit report," "loan," and "mortgage" exhibited the most significant beneficial impact on their respective categories. The results validated that the model's predictions were influenced by semantically pertinent phrases rather than arbitrary patterns, thus enhancing transparency and facilitating practical implementation.

All experiments were conducted on a workstation using an Intel Core i9-11900K CPU (3.5 GHz), 32 GB RAM, and an NVIDIA RTX-3080 GPU with 10 GB VRAM, utilizing TensorFlow 2.13. The ConvLSTM model necessitated approximately 72 minutes per epoch and converged in roughly 50 epochs, while the 1-Conv1D and LSTM models required approximately 38 minutes and 46 minutes per epoch, respectively, under identical conditions. Inference on the held-out test set, comprising approximately 30,000 samples, revealed that the ConvLSTM model achieved a throughput of roughly 2,300 samples per second, while the 1-Conv1D model obtained 3,800 samples per second and the LSTM model reached 3,100 samples per second. Consequently, ConvLSTM experienced an inference delay around 40 percent greater than that of the simpler systems. Despite ConvLSTM's higher computing requirements, its enhanced accuracy, precision, recall, and AUC warrant its implementation in contexts where predictive performance is prioritized over real-time limitations. These findings highlight the compromise between efficiency and performance in text categorization tasks. Subsequent research will explore model compression methodologies, including pruning and quantization, to reduce training and inference costs while maintaining robust predictive performance.

5 Conclusion

In general, the applications of text analysis in daily life are diverse and can be used in different fields, including educational environments, therapy, industry, etc. This research attempted to propose an accurate text classification model using text analysis techniques based on artificial intelligence (AI). For this purpose, a multi-class classification model was presented to classify textual data related to customer complaints. For this purpose, various schemes based on NN, including a one-dimensional convolutional model with one layer, two layers, and three layers, the LSTM model, and the LSTM-Convolutional model, were used. Also, during the processes, the RMSprop optimizer was used to increase the learning rate, categorical cross-entropy was used for loss during the training process, and word2index and word embedding methods were used to convert text into numerical vectors. Finally, the recommended schemes were examined and compared by conducting a case study based on an international dataset of customer complaints and using different evaluation indices. The results showed that the ConvLSTM model performed best based on the evaluation of the index. The values of the evaluation indices related to this model were all higher than the corresponding values in other schemes. Local and long-term features are learned by CNN and LSTM, respectively. Therefore, combining these two schemes allows the ConvLSTM model to consider both algorithms' abilities and capabilities. After that, the 1Conv1d, LSTM, 2Conv1d, and 3Conv1d schemes were placed in the following best-performing ranks. In other words, the 3Conv1d model has the least accuracy. Therefore, this model has the weakest performance among different schemes.

This work introduces the ConvLSTM model as an efficacious solution for multi-class classification tasks, exhibiting enhanced performance across multiple assessment measures. The model's capacity to integrate the advantages of convolutional and recurrent layers enables it to capture spatial and temporal information, rendering it especially fit for intricate data like text. Although the results are encouraging, the model's actual application may be constrained by issues such as computational expense and scalability, particularly when utilized with extensive datasets or real-time systems. Future research should investigate methods to mitigate these constraints, including the optimization of computing efficiency and the incorporation of supplementary NLP techniques to augment model performance. Furthermore, modifying ConvLSTM for real-time classification tasks may create new chances for its utilization in dynamic contexts, hence broadening its applicability across other fields.

Nomenclature

1Conv1d	A one-dimensional convolutional model with one layer.	n	The number of observations.
2Conv1d	A one-dimensional convolutional model with two layers is used.	NLP	Natural language processing
3Conv1d	A one-dimensional convolutional model with three layers is used.	NN	Neural network
AI	Artificial intelligence	PCA	Principal Component Analysis
AUC	Area Under Curve	ReLU	Rectified Linear Unit.
CNN	Convolutional Neural Network	RNN	Recurrent Neural Network
ConvLSTM	Convolution LSTM Neural Network.	ROC	Receiver Operating Characteristic
DL	Deep learning	TN	True Negative

FN	False Negative	TP	True Positive
FP	False Positive	TPR	True Positive Rate
FPR	False Positive Rate	x_i	The ith value of vector x.
LSTM	Long Short-Term Memory.	y_i	The ith value of vector y.
ML	Machine Learning		

Authorship contribution statement

Jing Yi: Writing-Original draft preparation, Conceptualization, Supervision, Project administration.

Xiao Zeng: design of methodology

Conflicts of interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Author statement

The manuscript has been read and approved by all the authors, the requirements for authorship, as stated earlier in this document, have been met, and each author believes that the manuscript represents honest work.

Ethical approval

All authors have been personally and actively involved in substantial work leading to the paper, and will take public responsibility for its content.

References

[1] Gentzkow, M., B. Kelly and M. Taddy (2019). Text as data. *Journal of Economic Literature*, aeaweb, 57(3): 535–574. DOI: 10.1257/jel.20181020

[2] Hannigan, T.R., R.F.J. Haans, K. Vakili, H. Tchalian, V.L. Glaser, M.S. Wang, S. Kaplan and P.D. Jennings (2019). Topic modeling in management research: Rendering new theory from textual data. *Academy of Management Annals*, Academy of Management, 13(2): 586–632. https://doi.org/10.5465/annals.2017.0099

[3] Kowsari, K., K. Jafari Meimandi, M. Heidarysafa, S. Mendu, L. Barnes and D. Brown (2019). Text classification algorithms: A survey. *Information*, MDPI, 10(4): 150. https://doi.org/10.3390/info10040150

[4] Hartmann, J., J. Huppertz, C. Schamp and M. Heitmann (2019). Comparing automated text classification methods. *International Journal of Research in Marketing*, Elsevier, 36(1): 20–38. https://doi.org/10.1016/j.ijresmar.2018.09.009

[5] Chakraborty, B. and T. Bhattacharjee (2020). A review on textual analysis of corporate disclosure according to the evolution of different automated methods. *Journal of Financial Reporting and Accounting*, Emerald, 18(4): 757–777. https://doi.org/10.1108/JFRA-02-2020-0047

[6] Ghazzawi, A. and B. Alharbi (2019). Analysis of customer complaints data using data mining techniques. *Procedia Computer Science*, Elsevier, 163: 62–69. https://doi.org/10.1016/j.procs.2019.12.087

[7] Zhong, B., X. Xing, P. Love, X. Wang and H. Luo (2019). Convolutional neural network: Deep learning-based classification of building quality problems. *Advanced Engineering Informatics*, Elsevier, 40: 46–57. https://doi.org/10.1016/j.aei.2019.02.009

[8] Singh, R.P., R. Haque, M. Hasanuzzaman and A. Way (2020). Identifying complaints from product reviews: A case study on hindi, In *Proceedings of the 17th International Conference on Natural Language Processing (ICON)*, ACL Anthology, pp: 108–116.

[9] Muqorobin, M., S. Rokhmah, I. Muslihah and N.A.R. Rais (2020). Classification of Community Complaints Against Public Services on Twitter. *International Journal of Computer and Information System (IJCIS)*, NELITI, 1(1): 7–10. https://doi.org/10.29040/ijcis.v1i1.6

[10] Hong, J., H. Kang and T. Hong (2020). Oversampling-based prediction of environmental complaints related to construction projects with imbalanced empirical-data learning. *Renewable and Sustainable Energy Reviews*, Elsevier, 134: 110402. https://doi.org/10.1016/j.rser.2020.110402

[11] Osorio-Arjona, J., J. Horak, R. Svoboda and Y. García-Ruíz (2021). Social media semantic perceptions on Madrid Metro system: Using Twitter data to link complaints to space. *Sustainable Cities and Society*, Elsevier, 64: 102530. https://doi.org/10.1016/j.scs.2020.102530

[12] Shin, J., S. Son and Y. Cha (2022). Spatial distribution modeling of customer complaints using machine learning for indoor water leakage management. *Sustainable Cities and Society*, Elsevier, 87: 104255. https://doi.org/10.1016/j.scs.2022.104255

[13] Shaukat, A. and U. Saif (2022). NLP based Model for Classification of Complaints: Autonomous and Intelligent System, In *2022 2nd International Conference on Digital Futures and Transformative Technologies (ICoDT2)*, IEEE, pp: 1–6.

[14] Nemer, Z.N., S.A.W. Saddam, R.J. AL-Sukeinee and E.J. Harfash (2025). A Comparative Performance Analysis of CNN, LSTM, and CNNLSTM Models for Classifying Sleep Stages Using Electroencephalography Signals. *Informatica*, informatica.si, 49(28). https://doi.org/10.31449/inf.v49i28.8082

[15] Wang, L (2025). Analysis Method of User Experience Influencing Factors Based on Improved LSTM Algorithm Technology. *Informatica*, informatica.si, 49(24). https://doi.org/10.31449/inf.v49i24.7406

[16] Alkenani, J. and M. Nickray (2025). Enhancing Network QoS via Attack Classification Using Convolutional Recurrent Neural Networks. *Informatica*, informatica.si, 49(2). https://doi.org/10.31449/inf.v49i2.7637

[17] Pan, L. and H. Ma (2025). A Computational CNN-LSTM-Based Mental Health Consultation System in a College Environment. *Informatica*, informatica.si, 49(10). https://doi.org/10.31449/inf.v49i10.7136

[18] Gardazi, N.M., A. Daud, M.K. Malik, A. Bukhari, T. Alsahfi and B. Alshemaimri (2025). BERT applications in natural language processing: a review. *Artificial Intelligence Review*, Springer, 58(6): 1–49. https://doi.org/10.1007/s10462-025-11162-5

[19] Mahajan, Y., M. Freestone, N. Bansal, S. Aakur and S.K.K. Santu (2024). Revisiting word embeddings in the LLM era. *ArXiv Preprint ArXiv:2402.11094*, Acl Anthology.

[20] Cao, H (2024). Recent advances in universal text embeddings: A Comprehensive Review of Top-Performing Methods on the MTEB Benchmark. *Preprint, ArXiv*. Https://Doi. Org/10.48550/ArXiv, 2406, Cornell University. https://doi.org/10.48550/arXiv.2406.01607

[21] Kim, T. and H. Wang (2025). Global dense vector representations for words or items using shared parameter alternating Tweedie model. *Mathematics*, MDPI, 13(4): 612. https://doi.org/10.3390/math13040612

[22] Zaland, O., M. Abulaish and M. Fazil (2023). A Comprehensive Empirical Evaluation of Existing Word Embedding Approaches. *ArXiv Preprint ArXiv:2303.07196*, Cornell University. https://doi.org/10.48550/arXiv.2303.07196

[23] Zeiler, M.D (2012). ADADELTA: an adaptive learning rate method. *ArXiv Preprint ArXiv:1212.5701*, Cornell University. https://doi.org/10.48550/arXiv.1212.5701

[24] Smith, L.N (2017). Cyclical learning rates for training neural networks, In *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*, IEEE, pp: 464–472. https://doi.org/10.1109/WACV.2017.58

[25] Lahitani, A.R., A.E. Permanasari and N.A. Setiawan (2016). Cosine similarity to determine similarity measure: Study case in online essay assessment, In *2016 4th International Conference on Cyber and IT Service Management*, IEEE, pp: 1–6. https://doi.org/10.1109/CITSM.2016.7577578

[26] Thongtan, T. and T. Phienthrakul (2019). Sentiment classification using document embeddings trained with cosine similarity, In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, ACL Anthology, pp: 407–414. https://doi.org/10.18653/v1/P19-2057

[27] Li, B. and L. Han (2013). Distance weighted cosine similarity measure for text classification, In *Intelligent Data Engineering and Automated Learning–IDEAL 2013: 14th International Conference, IDEAL 2013, Hefei, China, October 20-23, 2013. Proceedings 14*, Springer, pp: 611–618. https://doi.org/10.1007/978-3-642-41278-3_74

[28] Narkhede, S (2018). Understanding auc-roc curve. *Towards Data Science*, 26(1): 220–227.

[29] Walter, S.D (2005). The partial area under the summary ROC curve. *Statistics in Medicine*, Wiley Online Library, 24(13): 2025–2040. https://doi.org/10.1002/sim.2103

[30] Juba, B. and H.S. Le (2019). Precision-recall versus accuracy and the role of large data sets, In *Proceedings of the AAAI Conference on Artificial Intelligence*, AAAI, pp: 4039–4048. https://doi.org/10.1609/aaai.v33i01.33014039

[31] Alvarez, S.A (2002). An exact analytical relation among recall, precision, and classification accuracy in information retrieval. *Boston College, Boston, Technical Report BCCS-02-01*, Academia, 1–22.

[32] Powers, D.M.W (2020). Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation. *ArXiv Preprint ArXiv:2010.16061*, Cornell University.

[33] Susmitha, I. and V. Sivasamy (2022). Detection of forest fire using support vector machine in comparison with random forest to measure accuracy, precision and recall, In *2022 14th International Conference on Mathematics, Actuarial Science, Computer Science and Statistics (MACS)*, IEEE, pp: 1–6. https://doi.org/10.1109/MACS56771.2022.10022546

[34] Habib, G. and S. Qureshi (2022). Optimization and acceleration of convolutional neural networks: A survey. *Journal of King Saud University-Computer and Information Sciences*, Elsevier, 34(7): 4244–4268. https://doi.org/10.1016/j.jksuci.2020.10.004

[35] Ansari, M.A., A. Crampton, R. Garrard, B. Cai and M. Attallah (2022). A Convolutional Neural Network (CNN) classification to identify the presence of pores in powder bed fusion images. *The International Journal of Advanced Manufacturing Technology*, Springer, 120(7): 5133–5150. https://doi.org/10.1007/s00170-022-08995-7

[36] Valizadeh, M. and S.J. Wolff (2022). Convolutional Neural Network applications in additive manufacturing: A review. *Advances in Industrial and Manufacturing Engineering*, Elsevier, 4: 100072. https://doi.org/10.1016/j.aime.2022.100072

[37] Ahlawat, S., A. Choudhary, A. Nayyar, S. Singh and B. Yoon (2020). Improved handwritten digit recognition using convolutional neural networks (CNN). *Sensors*, MDPI, 20(12): 3344. https://doi.org/10.3390/s20123344

[38] Panahi, M., N. Sadhasivam, H.R. Pourghasemi, F. Rezaie and S. Lee (2020). Spatial prediction of groundwater potential mapping based on convolutional neural network (CNN) and support vector regression (SVR). *Journal of Hydrology*, Elsevier, 588: 125033. https://doi.org/10.1016/j.jhydrol.2020.125033

[39] Paul, E. and R.S. Sabeenian (2022). Modified convolutional neural network with pseudo-CNN for removing nonlinear noise in digital images. *Displays*, Elsevier, 74: 102258. https://doi.org/10.1016/j.displa.2022.102258

[40] WHIG, P (2022). More on convolution neural network CNN. *International Journal of Sustainable Development in Computing Science*, 4(1).

[41] Hakim, W.L., F. Rezaie, A.S. Nur, M. Panahi, K. Khosravi, C.-W. Lee and S. Lee (2022). Convolutional neural network (CNN) with metaheuristic optimization algorithms for landslide susceptibility mapping in Icheon, South Korea. *Journal of Environmental Management*, Elsevier, 305: 114367. https://doi.org/10.1016/j.jenvman.2021.114367

[42] Le, X.-H., H.V. Ho, G. Lee and S. Jung (2019). Application of long short-term memory (LSTM) neural network for flood forecasting. *Water*, MDPI, 11(7): 1387. https://doi.org/10.3390/w11071387

[43] Sahoo, B.B., R. Jha, A. Singh and D. Kumar (2019). Long short-term memory (LSTM) recurrent neural network for low-flow hydrological time series forecasting. *Acta Geophysica*, Springer, 67(5): 1471–1481. https://doi.org/10.1007/s11600-019-00330-1

[44] Sherstinsky, A (2020). Fundamentals of recurrent neural network (RNN) and long short-term memory (LSTM) network. *Physica D: Nonlinear Phenomena*, Elsevier, 404: 132306. https://doi.org/10.1016/j.physd.2019.132306

[45] Song, X., Y. Liu, L. Xue, J. Wang, J. Zhang, J. Wang, L. Jiang and Z. Cheng (2020). Time-series well performance prediction based on Long Short-Term Memory (LSTM) neural network model. *Journal of Petroleum Science and Engineering*, Elsevier, 186: 106682. https://doi.org/10.1016/j.petrol.2019.106682

[46] Hunt, K.M.R., G.R. Matthews, F. Pappenberger and C. Prudhomme (2022). Using a long short-term memory (LSTM) neural network to boost river streamflow forecasts over the western United States. *Hydrology and Earth System Sciences*, HESS, 26(21): 5449–5472. https://doi.org/10.5194/hess-26-5449-2022

[47] Hu, W.-S., H.-C. Li, L. Pan, W. Li, R. Tao and Q. Du (2020). Spatial–spectral feature extraction via deep ConvLSTM neural networks for hyperspectral image classification. *IEEE Transactions on Geoscience and Remote Sensing*, IEEE, 58(6): 4237–4250. https://doi.org/10.1109/TGRS.2019.2961947

[48] Jiang, W (2022). Internet traffic matrix prediction with convolutional LSTM neural network. *Internet Technology Letters*, Wiley Online Library, 5(2): e322. https://doi.org/10.1002/itl2.322

[49] Zhao, C., P. Zhang, J. Zhu, C. Wu, H. Wang and K. Xu (2019). Predicting tongue motion in unlabeled ultrasound videos using convolutional LSTM neural networks, In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, pp: 5926–5930. https://doi.org/10.1109/ICASSP.2019.8683081