Journal of Electrical and Computer Engineering Innovations
(JECEI)
Journal homepage: http://www.jecei.sru.ac.ir

Research Paper

Hybrid CNN–BiLSTM Model with BERT Embeddings for Urgency Detection in MOOC Forums

Mujtaba Sultani [1,2], ***Negin Daneshpour*** [1,*]

[1]*Faculty of Computer Engineering, Shahid Rajaee Teacher Training University, Tehran, Iran.*

[2]*Faculty of Computer Science, Kabul Polytechnic University (KPU), Kabul, Afghanistan.*

Article Information

Article History:

Received 27 November 2025
Reviewed 05 January 2026
Revised 03 February 2026
Accepted 22 February 2026

Keywords:

BERT
Data augmentation
CBiLSTM
Urgent post classification
Mooc
Deep learning
Text classification

*Corresponding Author's Email Address:
ndaneshpour@sru.ac.ir

Abstract

Background and Objectives: Discussion forums in Massive Open Online Courses (MOOCs) enable students to interact with instructors and share educational concerns. However, identifying urgent posts within the vast volume of discussions poses significant challenges. High dropout rates and the need for timely responses to urgent queries highlight the importance of efficient classification systems. This study addresses the binary classification of student posts in the Stanford MOOC Posts dataset into urgent and non-urgent categories, and aims to improve performance in the presence of class imbalance.

Methods: We propose a hybrid deep learning model that integrates BERT-based contextual embeddings with a Convolutional Neural Networks (CNN) and Bidirectional Long Short-Term Memory (BiLSTM) architecture to capture both local textual features and long-term dependencies. To mitigate the class imbalance issue, BERT-based data augmentation technique was employed which enriches minority class samples, and enhances model generalization and urgent post detection. The model's performance was compared against baseline methods, including CNN, LSTM, BiLSTM, and other state-of-the-art models. Evaluation metrics such as F1-weighted score and class-specific F1 scores were used to assess effectiveness.

Results: The model achieved a 93.3% F1-weighted score and an 84.1% F1 score for the urgent class which surpasses the best-performing state-of-the-art model by 0.6% and 0.8%, respectively. The results show the effectiveness of augmentation and hybrid architecture while identifying urgent posts, particularly within imbalanced datasets.

Conclusion: This research introduces a scalable and effective framework for urgent post detection in MOOCs. By leveraging BERT-based augmentation and a CNN–BiLSTM hybrid architecture that integrates contextual and sequential learning, the study provides automated forum analysis, offer timely insights for instructors and course designers to enhance students support, engagement, and retention.

How to cite this paper:
M. Sultani, N. Daneshpour, "Hybrid CNN–BiLSTM model with BERT embeddings for urgency detection in mooc forums," J. Electr. Comput. Eng. Innovations, 14(2): 487-506, 2026.

DOI: 10.22061/jecei.2026.12535.891

URL: https://jecei.sru.ac.ir/article_2542.html

Introduction

Educational technology has been reshaped by the rapid growth of Massive Online Open Courses (MOOCs). These courses aim to make education accessible worldwide and provide access to online resources[1]. More than 58 million users worldwide have enrolled in a MOOC. Currently, more than 220 million learners are participating in MOOCs worldwide [2]. The widespread popularity of MOOCs is also evident from the fact that more than 700 prestigious universities participate in them and offer a variety of courses accessible through platforms such as Coursera, edX, and Udacity. The wide availability of MOOCs signifies a shift in educational approaches that emphasizes the democratization of knowledge and promotes a culture of continuous lifelong learning [3]. Individuals can use these courses to learn new skills, enhance their understanding, and foster professional development according to their preferences [1].

A key aspect of MOOCs is the inclusion of communication platforms, particularly discussion forums. These forums facilitate interaction between learners and instructors, as well as peer-to-peer exchanges [4]. These forums play an important role in supporting different learning approaches that shape the educational experiences of MOOC participants. Moreover, these forums serve as a valuable channel for students to articulate their questions and urgent concerns [5].

However, considering the substantial quantity of MOOC participants and the limited number of instructors, it poses a challenge to effectively track and respond to students' posts and questions. Quick responses to important posts are often necessary to help students overcome obstacles during their learning journey. Failure to provide timely feedback can lead to learner frustration and increase dropout rates [6]. Therefore, it is necessary to develop mechanisms that differentiate urgent posts and ensure that they receive immediate attention and response from instructors [7], [8]. Implementing an effective system to monitor and handle urgent posts would allow instructors to give precedence to their responses and effectively handle the overwhelming volume of submissions. This system would not only optimize instructors' time and attention but also empower them to devote more energy to promote community engagement and provide valuable support [8].

Extensive research has focused on classifying MOOC forum posts into urgent and non-urgent categories. Various word representation techniques and classification approaches have been explored to develop effective models. The goal is to prioritize posts according to their urgency. In [8]-[10], statistical techniques like term frequency (TF), inverse document frequency (IDF), and term frequency-inverse document frequency (TF-IDF) were employed to represent words. However, they neglected the meaning of word order [11], which led to a limitation while capturing the document context. These studies used conventional classification algorithms like Support Vector Machine (SVM) and Nearest Centroid, which require low computational effort but have poor performance because they often rely heavily on manual feature selection [12]. Studies [12]-[15] have used pre-trained models like Google News and Glove to represent words with dense vectors that capture their contextual meaning within the document [16]. These studies used different architectures including multiple CNNs, CNN aggregation, GRU, and attention layers to develop their models. They emphasized including additional representational features to select effective features and assign higher weight to the most important features.

Despite these efforts, challenges remain, particularly in handling imbalanced datasets, which can affect classification performance and lead to bias toward larger classes [17], [18]. Improvements in the classification of imbalanced datasets have been classified into five categories: Data, Algorithms, Cost-Sensitive, Feature Selection, and Ensemble Approaches [19]. Common data-level methods include re-sampling to adjust the number of samples in the dataset. Oversampling involves adding samples, usually by copying samples, while under-sampling involves removing samples, often by random selection. While these methods have shown some effectiveness in data matching, they are not sufficient to completely solve the problem at hand [20]. In this study, we specifically address the data level imbalance and we focused on technique to balance text datasets. We employed BERT-based data augmentation (DA) to solve the data balancing problem [21]. Data augmentation method increases the scale of training data. This approach significantly contributes to mitigate overfitting and improve the robustness of machine learning models, especially for tasks with limited data availability [22].

On the other hand, model generalization, which denotes the ability of a machine-learning model to respond accurately to unseen data, is a critical aspect of model development [22]. However, the presence of a generalization constraint in the literature, poses a significant challenge when applying machine learning models, especially in subdomains or specific contexts where training data may be limited or unavailable. Educational data are inherently diverse and encompass different subjects, learning styles, and student demographics. Therefore, models must have the ability to adapt and perform reliably in different subdomains [22]. By closing the generalization gap,

machine learning models can provide more accurate and reliable insights, recommendations, and support for learners, educators, and educational institutions [22]. In this study, we address this constraint by employing BERT-based data augmentation to balance the dataset and using the BERT pre-trained model for word embeddings, both of which enhance model generalization and performance.

In this study, we employed the BERT model as the embedding layer, fine-tuned by a novel hybrid deep learning approach, combining Convolutional Neural Network (CNN) and Bidirectional Long Short-Term Memory (BiLSTM) layers. We named this hybrid model Convolutional BiLSTM (CBiLSTM), which enables an effective and precise classification of urgent posts. In CBiLSTM, the CNN layers effectively handle the high dimensionality of input texts, and BiLSTM layers explore feature context bidirectionally. The outcomes illustrate that CBiLSTM outperforms conventional deep learning models commonly utilized for urgent post classification using widely recognized Stanford MOOC Post Corpus. Our contributions are as follows:

1. We introduced a novel BERT-based data augmentation technique to balance the dataset, which is the first of its kind in the MOOC context.
2. We leveraged BERT pre-trained model as an embedding layer to enhance context understanding and semantic meaning for more accurate and context-aware representations.
3. We propose the CBiLSTM model that integrates CNN and BiLSTM layers for effective feature extraction and capturing long-term dependencies.
4. Our model outperforms conventional deep learning models in classifying urgent posts which demonstrates the approach's effectiveness and reliability using the Stanford MOOC Posts dataset.

The following sections of this paper provide a structured flow of our research. Section 2 offers a brief overview of related work, while Section 3 delves into the adapted method. In Section 4, we detail the experimental setup, data, and evaluation metrics. Results are discussed in Section 5, followed by future work in Section 6. Finally, Section 7 provides the conclusive results.

Related Work

In the field of online education, MOOCs have become widely adopted due to their flexibility and accessibility. As participation grows, discussion forums have become important centers for communication, knowledge sharing, and collaborative learning [23]. However, the high volume of posts makes it difficult for instructors to respond promptly, especially to urgent questions.

Consequently, researchers have focused on the classification of urgent posts within MOOC discussion forums to enhance the effectiveness and efficiency of instructor-learner interactions. MOOC post-classification employs both traditional machine learning and deep learning algorithms [14].

In [8], metadata and linguistic features were used in conjunction with models such as AdaBoost to identify urgent posts. While these approaches offered modest improvements, they depended heavily on handcrafted features and struggled to capture semantic nuances. Similarly, [24] developed a multi-dimensional classification framework for urgency, sentiment, and confusion. Although effective within specific courses, the method lacked generalizability across domains. [25] performed an analysis of over 100,000 discussion posts on Coursera. They used linear regression in combination with a Gradient Lifting Decision Tree (GBDT) to classify MOOC discussion posts. In particular, this model uses features that are independent of course content and achieved an overall accuracy of 85%. These models, however, continued to rely on shallow representations that cannot fully interpret contextual meaning.

Deep learning approaches have addressed some of these limitations. The convolutional LSTM architecture in [26] captured both local and sequential features, and achieves 86.6% accuracy for urgent post classification. Yet, the use of static embeddings limited their ability to model contextual word meanings. Similarly, [14] combined CNN, GRU, and Char-CNN layers with pre-trained Google News word vectors to handle noisy text, but the embedding remained context-independent and insufficient for capturing subtle urgency cues. Additional studies investigated confusion detection or relevance classification using logistic regression, bag-of-words, or SVMs [9], [27], but these approaches continued to face challenges with semantic ambiguity and domain transferability.

Khodeir [15] introduced a Bi-GRU model enhanced with BERT embeddings for urgent post classification. Although this work demonstrated improved performance through contextual representations, the architecture relied on a single recurrent component and lacked mechanisms to fully capture both local and long-range semantic patterns. Similarly, El-Rashidy et al. [12] proposed weighted BERT features combined with multi-CNN models to improve MOOC post classification. While effective, this approach focused primarily on convolutional feature extraction and did not integrate sequential modeling which limits model ability to encode temporal dependencies that are often important in urgency-related discourse. Table 1 presents an overview of previous research results.

Table 1: Overview of past study results

Authors	Dataset	Approach /Model	Embedding Layer	Results/Findings
[12]	Stanford MOOC Posts	Feature aggregation model based on CNN	BERT	The proposed model first aggregates feature to capture data-driven relationships among token features as well as their representation. Then CNN is implemented to enhance the accuracy of text context interpretation. Finally, these combined features are utilized for post text classification. The model achieved 92.7% F1-weighted scores.
[15]	Stanford MOOC Posts	Bi-GRU	BERT	The model is created based on BERT as embedding layer and achieved 91.9% F1-weighted scores.
[14]	Stanford MOOC Posts	Hybrid CNN + GRU as well as Char-CNN	Google-news	This hybrid model was proposed to tackle with the noise of spelling errors and emoticons. The model achieved an impressed 91.8% F1-weighted scores.
[8]	Stanford MOOC Posts	AdaBoost (Decision Tree)	LIWC + Term Frequency	This study tried traditional classification algorithms namely, NB, SVM, RF, AdaBoost, and LR. The best 88% F1-Score was achieved by AdaBoost.
[25]	Stanford MOOC Posts	Linear Regression + gradient lifting decision tree (GBDT)	Google Word2Vec	This model classified sentiment, urgency and confusion. This model achieved 86.6% accuracy in classifying urgent posts.
[26]	More than 100,000 Coursera threads	Hybrid of CNN + LSTM		The proposed model uses features that are independent of course content and achieves an impressive overall accuracy of 85%. They also obtained that most of the posts were not related to the course contents.
[24]	Stanford MOOC Posts	NB, SVM, RF	TF-IDF with unigram features	Proposed classification model considered urgency, sentiment and confusion in different domains. This model gave good result within the domain but is difficult to generalize across different domains.
[9]	Stanford MOOC Posts	Logistic Regression	Bag of words with unigram features	A two-stage model is introduced: initially, the model identifies instances of confusion, followed by the application of a recommendation system to suggest brief video clips aimed at resolving that confusion. This model obtained 77% F1-scores.

Method

We present a new method to classify urgent posts in MOOC discussion forums which integrates Convolutional Neural Network (CNN) and Bidirectional Long Short-Term Memory (BiLSTM) layers based on BERT. CBiLSTM model aims to accurately classify MOOC forum posts as urgent or non-urgent. The proposed approach follows a multi-stage process, which involves data preprocessing, BERT-based data augmentation to enrich the training set with contextualized samples, BERT-based word embedding to facilitate a deeper understanding of the text, and the comprehensive description of the CBiLSTM model. The method formulation flow is illustrated in Fig. 1.

A. Preprocessing

In the preprocessing phase of the Stanford MOOC Posts dataset, a series of essential steps are undertaken to ensure data consistency and quality. Firstly, all URLs (Uniform Resource Locators) were removed from the text to eliminate any potential noise and irrelevant information. To standardize word forms, contractions like "won't" and "can't" are replaced with expanded versions, such as "will not" and "can not" to facilitate uniformity in text representation. Similarly, symbols like question marks and exclamation marks were substituted with a specific word to promote a cohesive approach to text analysis.

To enhance text readability, abbreviations like "'re", "n't", "'s", and others were transformed into their corresponding full words ('are', 'not', 'is', etc.) to streamline the text for further analysis. Further, symbols such as slashes, dollar signs, and others are eliminated to ensure that special characters do not interfere with the text's semantic meaning.

To facilitate lemmatization, the Spacy 'en_core_web_ls' model is applied which groups inflected forms into their base or dictionary forms to lead to a more coherent and meaningful representation of words. Stop words were retained in the dataset, as their inclusion was found to improve the classification results [28].

The 'course_display_name' metadata feature is integrated with student posts which display the course's name and domain within the MOOC Posts dataset. By incorporating 'course_display_name' into the posts, a considerable improvement in the achieved results was observed [14].

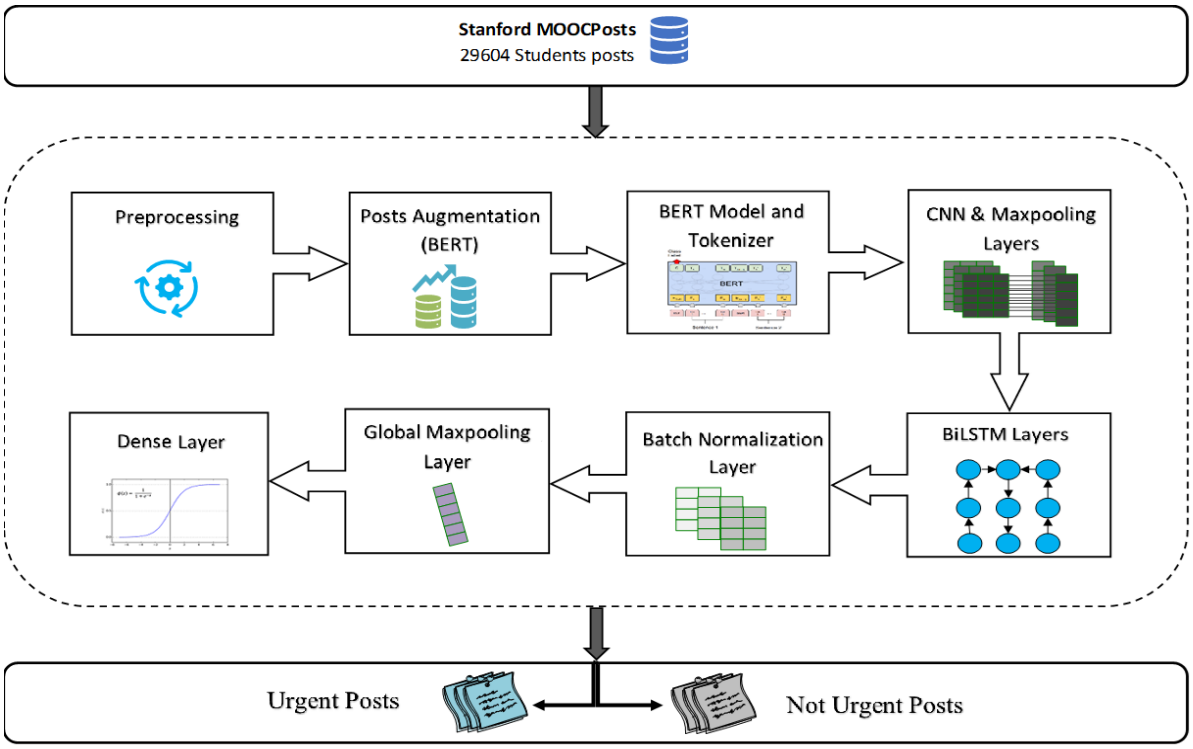


Fig. 1: The method formulation flow.

B. Addressing Data Imbalance

Text data imbalance is a prevalent challenge in natural language processing tasks, where certain classes or categories within a dataset are significantly underrepresented compared to others. Machine learning models can be adversely affected by this imbalance which leads to biased predictions and reduced accuracy, especially for minority classes [29]. Synthetic data methods like SMOTE [30] and AdaSyn [31] effectively address statistical data imbalances. However, when applied to textual data, they face challenges with overfitting and noise. Although GANs like CycleGAN [32] have shown promise in generating synthetic numerical and image data, their suitability for textual data, including grammar, context, and semantics, requires further evaluation.

We employed BERT (Bidirectional Encoder Representations from Transformers) to augment the training dataset. Its strong language understanding and contextual modeling capabilities improve the quality of the augmented data, thereby enhancing overall model performance. In the context of text augmentation using BERT, as seen in Fig. 2, the initial step is tokenization which breaks the input sentence into discrete units like words. Subsequently, (MASK) tokens are inserted at random positions within the sentence which results in a partially masked sentence. After augmentation, the sentence is passed through the BERT model which leverages contextual information from the surrounding tokens to predict the appropriate replacements for the (MASK) tokens.

Table 2 presents the original samples alongside their corresponding augmented variations.

Prior to augmentation, the training distribution consisted of 4,063 urgent vs. 15,934 non-urgent samples. We generated 11,871 synthetic urgent samples, increasing the urgent class size to 15,934, matching the majority class. This process expanded the training set from 19,997 to 32,186 instances. Although some generated sentences contained minor grammatical distortions, we retained them because prior studies show that such noise does not harm and may even improve model robustness when semantic meaning is preserved.

Kobayashi [40] demonstrated that contextual augmentation with slight syntactic noise maintains or enhances performance, while Wei and Zou [41] reported that transformer-based models tolerate minor grammatical irregularities without degrading classification accuracy. Therefore, all augmented samples were used during training.

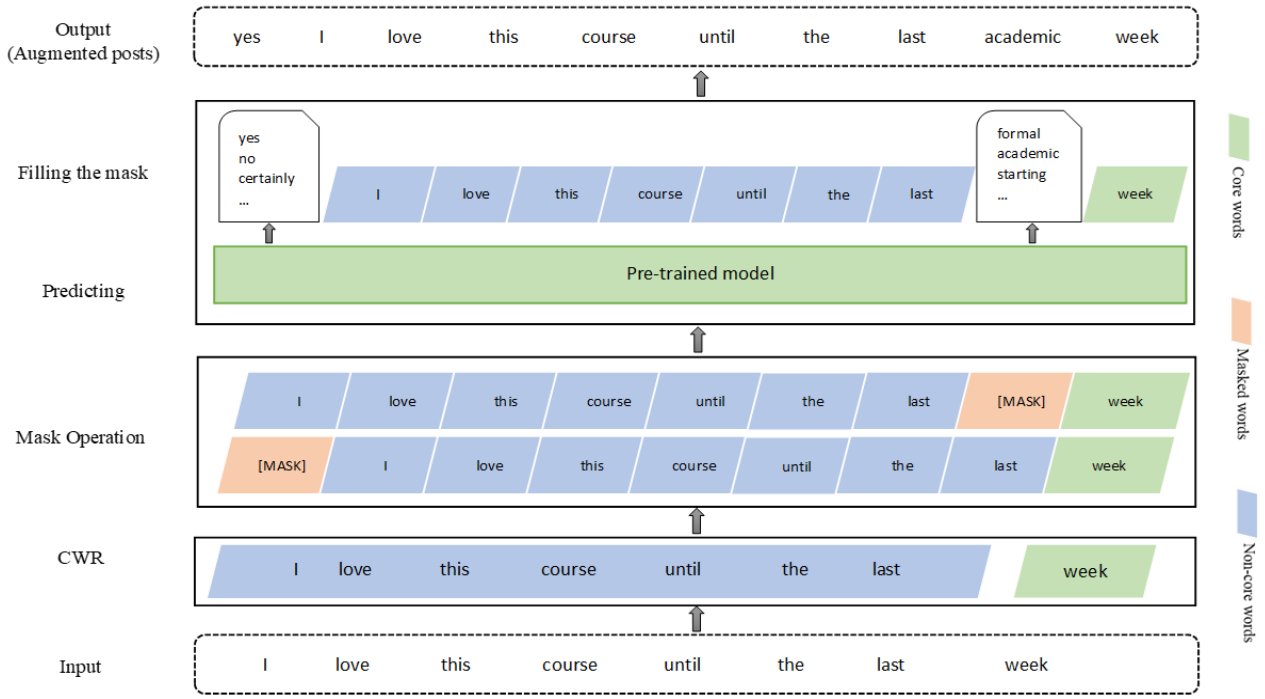


Fig. 2: Internal process of BERT text augmentation.

Table 2: Original posts alongside their corresponding augmented variations

No	Original posts	Augmented posts	Urgency
1	I love this course until the last week when try to submit work and all I get be the stupid try again come up have write in the education education one one five number how to learn math	yes I love this course until the last academic week and when try to submit work applications and instead all I get be the more stupid try again come up have write out in the joint discussion that this be happen education education one one five number how next to learn math	Urgent
2	I have the same issue take a look at the update on some platform issue discussion topic they be fix some bug education education one one five number how to learn math	I have the same issue take to a quick look then at from the news update when on some platform more issue discussion in topic they be fix some bug computer education education one one five plus number how to learn math	Not Urgent

BERT-generated variations are introduced to the urgent class samples to address the data imbalance issue and lead to notable improvements in model performance and generalization. Fig. 3 illustrates the distribution of both urgent and not-urgent classes following the augmentation process.

Tokenization is performed using the pre-trained BERT model, specifically the 'pre-trained bert-base-uncased tokenizer', which is case-insensitive and offered by the transformer's library.

Algorithm 1 shows the augmentation process steps.

C. Embedding Layer

In NLP and neural network language models, the word embedding layer is pivotal. It is responsible for transforming words or tokens in a text into dense numerical vectors in a high-dimensional space. Before BERT, Word2Vec, and Glove were widely used in NLP tasks for traditional word embeddings. BERT, as introduced by [33], is a transformer-based model, unlike traditional approaches, BERT learns contextual embeddings. It is pre-trained on a massive corpus using next-sentence prediction and masked language modeling tasks. This pre-training enables BERT to understand the semantic meaning and syntactic structure of words in various contexts. BERT-based word embeddings not only capture the semantic meaning of individual words but also their contextual significance in sentences or documents. BERT, as shown in Fig. 4, uses special tokens such as '(CLS)' and '(SEP)' to handle variable-length sequences of text. The '(CLS)' token represents the classification token and is used to achieve a fixed-size vector representation for the entire input sequence, which can be used for downstream classification tasks. The "(SEP)" token separates different sentences in the input when dealing with sentence-level tasks [33].

Algorithm 1*: BERT-based dataset augmentation*

Input:

- ✓ Original dataset of text samples: *D*
- ✓ BERT-based augmentation instance: *augmenter*
- ✓ Number of augmentation repetitions (*optional*): *repetitions*
- ✓ Number of augmented samples per minority class sample (*optional*): *samples*

Output: Augmented dataset of text samples: A

Algorithm:

Step 1: Load the original dataset (*D*).
Step 2: Select minority class samples (*minority_samples*) from the original dataset (*D*).
Step 3: Generate augmented texts (*augmented_texts*) for each minority class sample (*minority_samples*) using the augmenter.
Step 4: Create an augmented dataset (A) containing the original labels and generated augmented texts.
Step 5: Repeat steps 3 and 4 for the specified repetitions (*if applicable*).
Step 6: Append the augmented A to the original D.
Step 7: Shuffle the entire dataset (*A*) to create a balanced and diverse dataset.

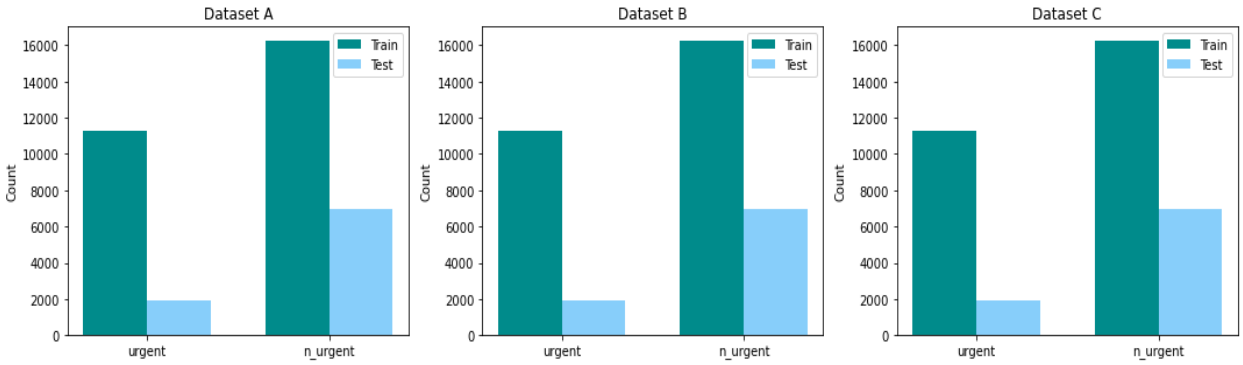


Fig. 3: Distribution of urgent and not urgent classes following the augmentation process.

In text classification, BERT encapsulates the entire sequence by utilizing the final hidden state of the special token (CLS). There are three distinct fine-tuning techniques: 1) Training the Entire Architecture: All layers of the BERT model, including task-specific layers, are updated during fine-tuning, which allows maximum flexibility and adaptation to the specific task. 2) Training Some Layers While Freezing Others: Only a subset of BERT layers is updated, while others are frozen, which strikes a balance between flexibility and computational efficiency. 3) Freeze the Entire Architecture: All layers, including pre-trained and task-specific, are kept frozen which makes BERT function as a fixed feature extractor for subsequent tasks [15].

In this study, a pre-trained BERT model was loaded and used for tokenization. The BERT model was kept frozen during fine-tuning, while only the CBiLSTM components were trained to learn from its representations.

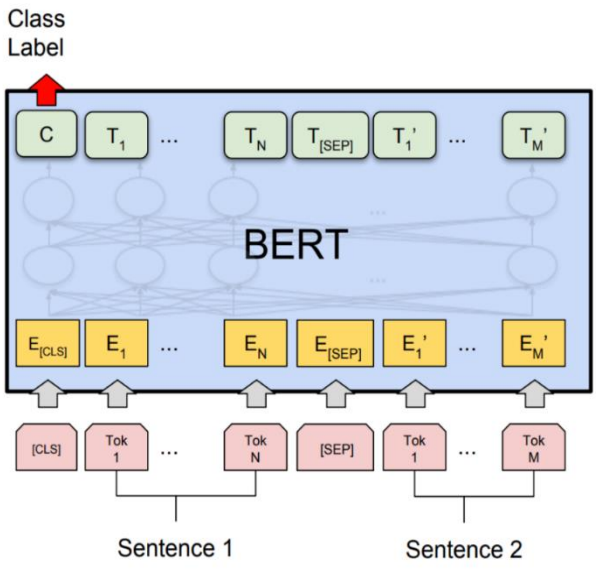


Fig. 4: BERT (Language Model) [33].

D. Convolutional BiLSTM (CBiLSTM) Model

CNN and RNN are widely used models in sentiment analysis and text classification. Each model has distinct strengths: CNN excels at extracting local features by analyzing the spatial relationships within the text but cannot learn sequential correlations, while RNN is adept at capturing sequential correlations and extracting global features [34], [35]. However, traditional RNNs suffer from issues like gradient explosion or vanishing gradient when dealing with long sequences of data. To overcome these limitations, LSTM [35], an extension of RNN, was introduced. LSTM employs memory cells to handle gradient issues and capture long-term dependencies. In a traditional LSTM, data propagates through the network unidirectionally, as depicted in Fig. 5-a. While this allows the LSTM to model past dependencies effectively, it may not fully leverage future context, which is important in text classification tasks.

LSTM uses a forget gate (f_{gt}), input gate (i_{gt}), and output gate (o_{gt}), constructed with a sigmoid layer to control the information flow within its cells [36]. The current cell state of the LSTM is represented as c_t. The forget gate decides what information to discard, determined by (1):

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \quad (1)$$

where, w_f and b_f illustrate the forget gate's weights and bias, while x_t corresponds to the input sequence and h_{t-1} signifies the previous output state.

Next, the LSTM layer utilizes its input gate to determine which elements from the current input x_t should be incorporated into the present cell state c_t. This process involves the use of both sigmoid and tanh layers. The sigmoid layer determines updates to the current cell state, as shown in (2). On the other hand, the tanh layer generates a new vector, denoted as c'_t using (3), composed of the updated values. W_i and b_i represent the weights and bias of the input gate respectively.

$$i_{gt} = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \quad (2)$$

$$c'_t = \tanh(W_c \cdot [h_{t-1}, x_t] + b_c) \quad (3)$$

Next, LSTM updates the previous cell state c_{t-1} with the new cell state c_t by multiplying forget gate f_{gt} values with c_{t-1}, followed by the addition of the newly computed candidate values scaled by the i_{gt}, as displayed in (4).

$$c_t = fg_t \cdot c_{t-1} + c'_t \cdot ig_t \quad (4)$$

As a result, the LSTM employs its output gate, represented by (5) and (6):

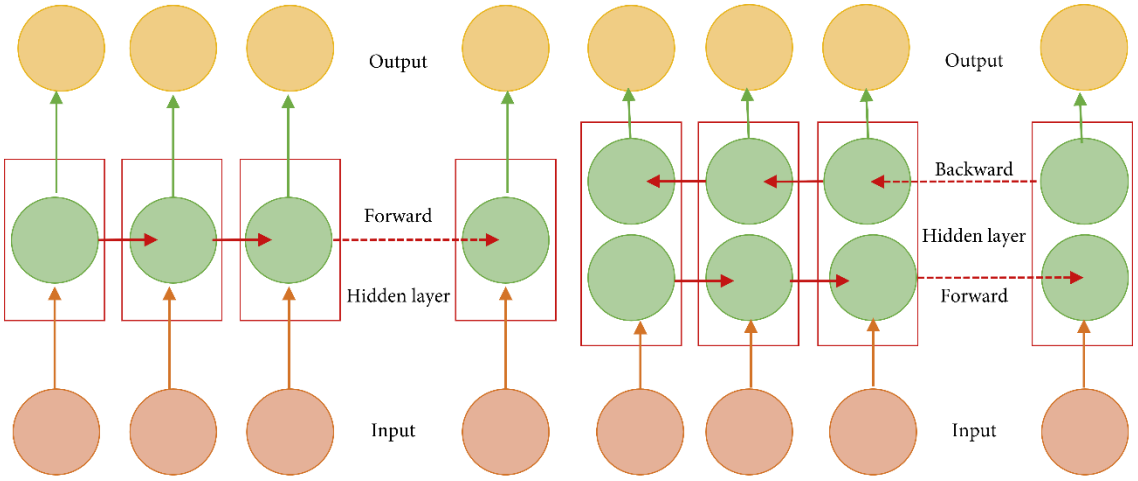
$$Og_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \quad (5)$$

$$h_t = og_t \cdot \tanh(c_t) \quad (6)$$

The final hidden state h_t is subsequently passed to a fully connected layer for further processing.

Moreover, Bi-LSTM [37] enhances LSTM's capabilities by incorporating two LSTM layers which simultaneously process the information bidirectionally, as depicted in Fig. 5-b.

The forward LSTM processes the input sequence from the first step to the last, while the backward LSTM processes it in reverse, from the last step to the first. This enables Bi-LSTM to better capture bidirectional dependencies in the data, making it particularly effective in various text classification tasks.



A: LSTM ***B: Bi-LSTM***

Fig. 5: Depiction of LSTM model (A) alongside a Bi-LSTM model (B) [39].

We propose CBiLSTM, a hybrid deep learning model that utilizes BERT for word embedding. The model architecture comprises two main components: CNN and BiLSTM layers. In the first step, we pass the BERT-based embeddings to the CNN structure. Following the BERT embedding layer, a series of one-dimensional convolutional layers are applied. These layers employ kernel sizes of 3, 4, and 5, each with 128 filters and ReLU activation functions. These convolutions are designed to identify local patterns and features within the embedded text. After each convolutional layer, max-pooling operations with varying pool sizes (8, 6, and 3) are performed to reduce dimensionality and retain the most significant features. A dropout layer with a 0.2 rate is added after the first convolutional layer to address overfitting.

Furthermore, the feature vectors are passed through a bidirectional LSTM layer. The model incorporates three Bi-LSTM layers with hidden units of 256, 128, and 64, respectively, each set to return sequences (return sequences=True). Bi-LSTMs capture contextual information bidirectionally which enables the model to understand the sequential dependencies within the text. To further process the output from the BiLSTM, another dropout layer with a rate of 0.1 is inserted to further enhance generalization. A batch normalization layer is applied to stabilize training by normalizing activations. Following the batch normalization layer, a global max-pooling layer is used which aggregates the most important features across the entire sequence by creating a fixed-length representation of the text data. The model concludes with a dense output layer featuring a single neuron using a sigmoid activation and provides a probability value between 0 and 1. This output layer classifies the input into either the urgent or not urgent classes.

For model training, the Adam optimizer is employed alongside a binary cross-entropy loss function. During model training, several callbacks are employed, including model checkpointing, early stopping, and learning rate reduction. These strategies ensure model convergence and prevent overfitting. Table 3 shows the detailed key parameters and threshold for the proposed model structure.

Table 3: Key parameters and threshold for the proposed model structure

Layers	**Parameters and Threshold**
BERT_base_uncased	Pre-trained model, 512 token limit, 768 features per token
Multi-Layer CNN	Number of layers: 3 Number of CNN filters: 128 – 128 – 128 Kernel sizes: 3 – 4 – 5 Activation: Relu
MaxPooling1D Layers	Number of layers: 3 Pool sizes: 8 – 6 – 3
Multi-Layer BiLSTM	Number of layers: 3 Number of hidden units: 256 – 128 – 64 Bidirectional (Return sequences): true
Dropout	Number of layers: 2 Dropout rate: 0.2 – 0.1
Dense Layer	Number of units: 1 Activation: Sigmoid
General training parameters	
Early Stopping	Patience: 3 Monitor: val_loss Min_delta: 0 Mode: min Verbose: 1
Reduce_lr	Monitor: val_loss Min_lr: 0.000001 Patience: 1 Mode: min Verbose: 1 Min_delta: 0.01
Loss	Binary_crossentropy
Optimizer	Adam
Epochs	10
Batch_size	128

Experiments

This section provides insights into the dataset, experimental setup, and evaluation metrics employed in this study.

A. Dataset and Experimental Setup

In this research, the experiments were conducted on the Stanford MOOC Posts dataset [28], a benchmark corpus introduced by [9]. The corpus comprises 29,604 anonymized learner forum posts from 11 Stanford University public online courses. These posts are divided into Humanities/Sciences, Medicine, and Education distinct domains, each containing 9723, 10001, and 9878 posts respectively. Each post is manually labeled across various dimensions, including assessing its urgency ranked on a scale from 1 to 7. To create a binary classification task for urgent post-identification, the class labels were adjusted. Posts with urgency scores of 4 or higher were categorized as "urgent," while those with scores below 4 were labeled as "not urgent." This binary classification scheme ensures that approximately 20% of posts are classified as urgent which allows instructors to promptly address critical cases, save 80% of their time, and enable efficient management of urgent posts.

Following the approach of previous studies [8], [12], [15], the dataset was categorized into three groups using course and domain names:

- Group A: The baseline scenario, where the training and test datasets remained independent across courses or domains. The whole dataset is divided into three distinct subsets.
- Group B: In this case, data division was determined by the course name. Several courses were excluded from the training phase. Specifically, courses from the Humanities and Medicine domains, such as Stat Learning (Winter 2014), Statistics in Medicine, and Managing Emergencies: What Every Doctor Must Know?, were selected for testing. As the Education domain had only one course, 33% of its posts were reserved for testing in a stratified manner.
- Group C: In this case, the domain was reserved for testing, and the classifier was trained on posts from the Medicine and Education domains. The evaluation was performed on posts from the Humanities domain, which was held out for testing. The choice of Humanities as the domain for testing was arbitrary.

As mentioned in embedding layer sub section, we conducted BERT-based data augmentation on the training sets of all three groups to address the dataset's class imbalance issue. This allows our model to learn more effectively and improves its performance while classifying urgent posts. By achieving a balanced dataset, this study aims to enhance the generalization and robustness of our model which enables the model to handle various real-world scenarios with improved precision and recall.

We employed TensorFlow and Keras libraries and utilized Python version 3.10, to build and train our proposed model. Prior to the data being used by the model, a series of preprocessing steps were carried out to ensure data quality and consistency. To facilitate the tokenization process and word embedding, we applied the pre-trained 'bert-based-uncased' model from the transformer's library. Special tokens were added to format the input sequences appropriately. As the 'bert-based-uncased' model has a constraint of a maximum input size of 512 tokens, we truncate sequences to the defined maximum length to ensure the compatibility of our data with the BERT model. As mentioned in convolutional BiLSTM (CBiLSTM) Model sub section , after a wide range of experiments, we selected the optimizer, number of layers, and number of hidden units of both CNN and BiLSTM components. These decisions were made after thorough experimentation and optimization to achieve the best performance.

B. Evaluation Metrics

In this study, we use six evaluation metrics to evaluate the model's performance: precision, recall, F1-score, F1-Weighted, Learning curve (LC), and Precision-Recall Curve (PRC). Precision measures the fraction of accurately predicted positive results by the model, while recall represents the proportion of relevant positive results correctly predicted. F1-score, a balanced metric, blends precision and recall by taking their harmonic mean to provide a single performance measure. Equations (7), (8), and (9) present the equations for these performance metrics, illustrating their mathematical formulations.

$$PR = TP / (TP + FP) \quad (7)$$

$$RC = TP / (TP + TN) \quad (8)$$

$$F1\text{-score} = (2 \cdot PR \cdot RC) / (PR + RC) \quad (9)$$

TP (True Positives), represents correctly predicted positive cases, TN (True Negatives), denotes correctly predicted negative cases, and FP (False Positives), indicates instances where the model incorrectly predicts positive cases.

Additionally, the F1-weighted score is employed to address class imbalances in the dataset by computing the weighted average of the F1 score for each class.

Learning curves are mostly employed as a diagnostic tool in machine learning, especially for incremental

learning algorithms like deep learning [38]. These curves provide a comprehensive evaluation of model performance by considering both the training and validation datasets.

They yield two insightful curves: The Training Learning Curve and the Validation Learning Curve. The former offers a glimpse into how effectively the model is "learning" from the training data, while the latter delves into how proficiently the model is "generalizing" its knowledge.

In the context of learning curves, it's common to employ minimization scores, like loss.

Smaller scores indicate more effective learning, with an ideal score of 0.0 signifying perfect learning of the training dataset. Regularly inspecting these learning curves during training serves as a powerful tool for analyzing learning-related issues like underfitting or overfitting. In our experimental setup, we employ early stopping, a technique that involves continuous monitoring of validation set errors. Whenever an improvement is observed, we capture a snapshot of the model parameters at that point. Upon termination of the training algorithm, we retain and use these saved parameters instead of the latest ones. This approach contributes to enhancing the generalization capabilities of deep neural networks.

ROC and PR curves are valuable tools for assessing probability predictions in binary classification problems [15]. ROC curves are graphical representations used to illustrate the trade-off between true positive rates and false positive rates at various classification thresholds. In contrast, PR curves present the trade-off between precision and recall at different classification thresholds. When dealing with imbalanced datasets, where one class significantly outweighs the other, PR curves become more important. In such scenarios, the Precision-Recall Plot proves to be a more informative tool for assessing binary classifiers compared to the ROC plot.

To summarize model performance and facilitate comparisons between different classifiers, we turn to the Area Under the Curve (AUC) metric. AUC metric summarizes the skill of the model. The precision-recall curve's baseline, denoted as $y = P/(P + N)$, where P is positive and N is negative, acts as a reference for a no-skill classifier. Such a classifier cannot distinguish between classes and predicts either a random outcome or a constant class for all instances.

Result and Discussion

This section features an in-depth performance analysis, comparing the proposed model with baseline architectures, including standalone CNN, LSTM, and Bi-LSTM models.

The baseline models provide foundational insights into the individual contributions of convolutional and sequential modeling components. It is important to note that we evaluate the performance of baseline models on augmented datasets (A, B, and C) and subsequently compare their results to our proposed model. Additionally, we compare our model performance against state-of-the-art models that have recently emerged.

In this study, a balanced dataset refers to a dataset in which the classes contain an approximately equal number of samples, whereas an imbalanced dataset contains uneven class distributions, often leading to biased model performance.

A. Performance Comparison: CBiLSTM Vs. Baseline Models

In the pursuit of enhancing urgent post-classification performance, we conducted a comprehensive comparative analysis between our primary model, CBiLSTM, and foundational baseline models, namely CNN, LSTM, and BiLSTM.

The learning curve analysis depicted in Figs. 6 to 9 for datasets A, B, and C shows how the models are learning and generalizing. The data visualizations clearly depict that employing BERT as an embedding layer result in a substantial decrease in the training time required to meet the early stopping criteria. Additionally, the graphical representations indicate that our CBiLSTM model, when incorporating BERT as the embedding layer, achieves early stopping with minimal loss in fewer training iterations which results in improved efficiency and cost-effectiveness.

A closer inspection of the learning curves in Figs. 6 to 9 reveals several observations.

First, the CBiLSTM model consistently reaches its minimum validation loss in fewer epochs compared to the standalone CNN, LSTM, and BiLSTM models. This behavior indicates that the integration of BERT embeddings accelerates convergence by providing semantically rich contextual representations at the input level.

As a result, the model requires fewer training iterations to learn discriminative patterns within MOOC posts.

Second, the smaller gap between training and validation losses for CBiLSTM particularly in Groups B and C suggests stronger generalization and reduced overfitting, even in cross-course and cross-domain scenarios.

In contrast, the baseline models exhibit larger discrepancies between training and validation curves, which indicates a weaker ability to transfer learned representations across diverse course contexts. These observations highlight the role of contextualized embeddings in stabilizing learning dynamics and improving generalization efficiency.

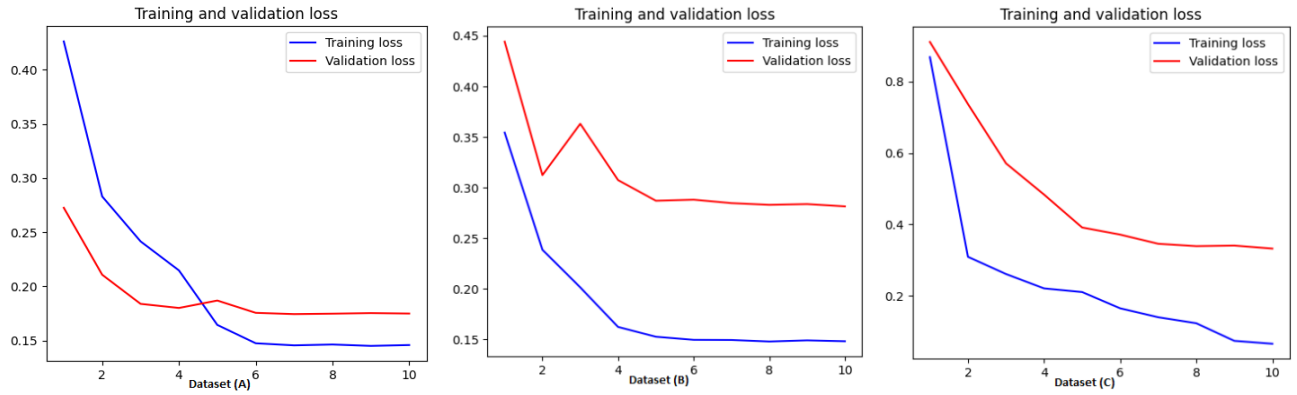


Fig. 6: Learning and validation curves for CNN on datasets A, B, and C where the minimum loss values are 0.17 (Epoch 7), 0.287 (Epoch 10), and 0.331 (Epoch 10).

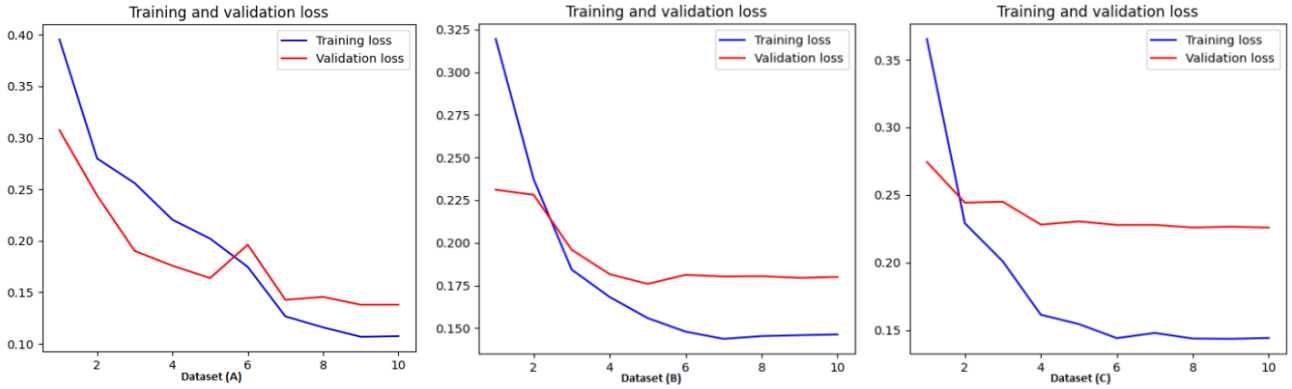


Fig. 7: Learning and validation curves for LSTM on datasets A, B, and C where the minimum loss values are 0.142 (Epoch 9), 0.172 (Epoch 5), and 0.229 (Epoch 8).

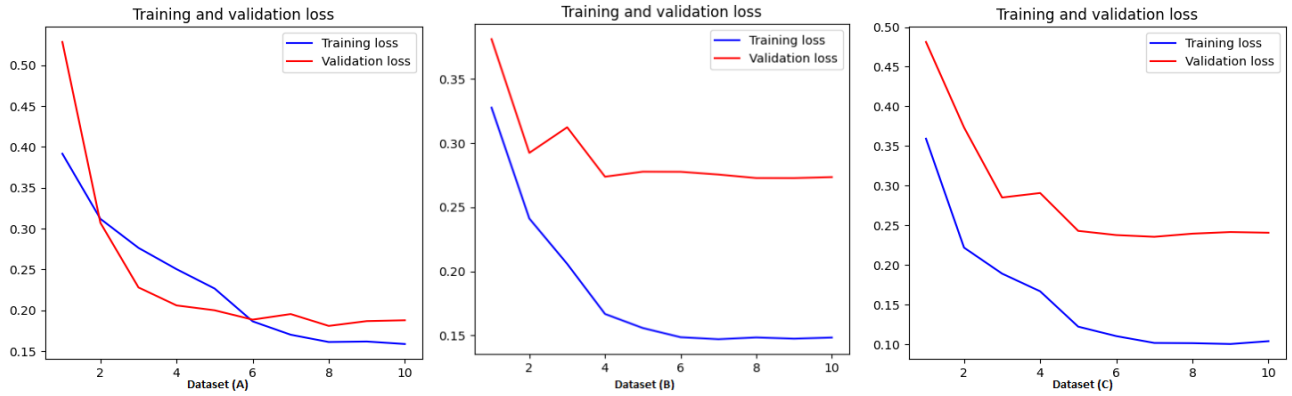


Fig. 8: Learning and validation curves for BiLSTM on datasets A, B, and C where the minimum loss values are 0.181 (Epoch 8), 0.272 (Epoch 9), and 0.236 (Epoch 9).

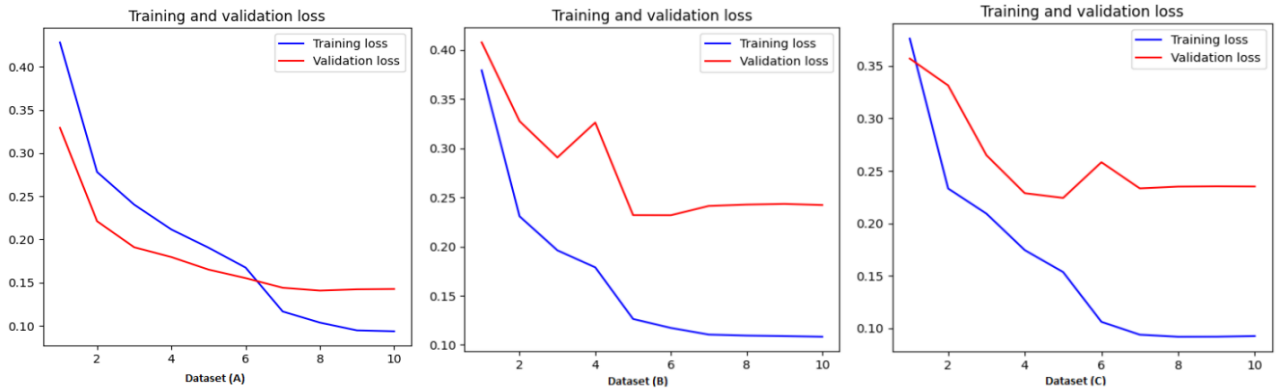


Fig. 9: Learning and validation curves for CBiLSTM on datasets A, B, and C where the minimum loss values are 0.147 (Epoch 8), 0.232 (Epoch 5), and 0.226 (Epoch 5).

The Precision-Recall curves in Figs. 10 to 13 illustrate the balance between true positive rate and positive predictive value. The AUC is employed as a quantitative metric to show the model performance. Furthermore, we employ a baseline for the precision-recall curves to show the balance between urgent and non-urgent classes. Notably, across all three datasets (A, B, and C), CBiLSTM achieved the highest AUC values and surpassed the baseline models. The analysis of the precision–recall curves shows that the proposed CBiLSTM maintains high precision even as recall increases, which demonstrates its robustness in detecting urgent posts especially under class imbalance. Unlike the baseline CNN, LSTM, and BiLSTM models, whose curves drop sharply as recall grows, the CBiLSTM curve declines gradually, and indicates fewer false positives at higher recall thresholds. This behavior reflects the strength of BERT's contextual embeddings combined with bi-directional sequential modeling, and enables the model to better distinguish subtle urgency cues in MOOC posts. The advantage is even more pronounced in Group C (cross-domain evaluation), where domain shift typically weakens classifier performance. However, the CBiLSTM retains a significantly larger AUC-PR margin above the no-skill baseline. These observations confirm that the model not only detects urgent posts more accurately but also sustains consistent performance across heterogeneous course and domain settings.

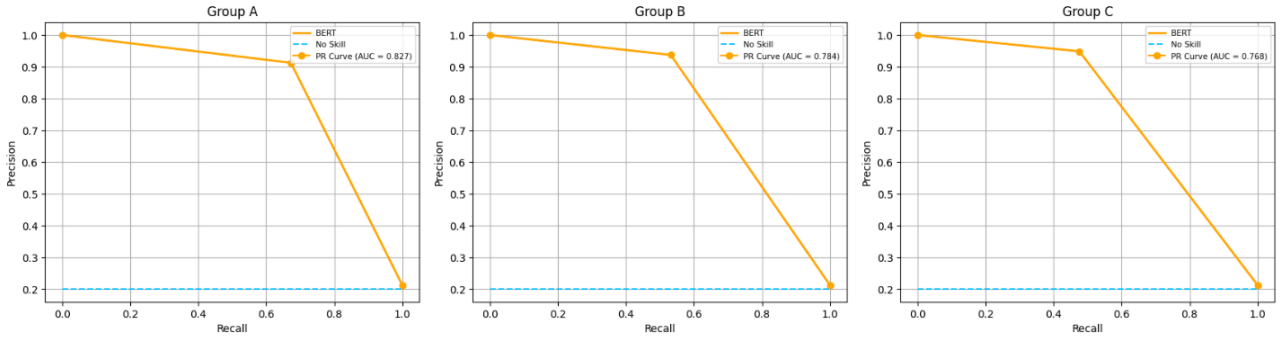


Fig. 10: PR curves for CNN model using A, B, and C datasets.

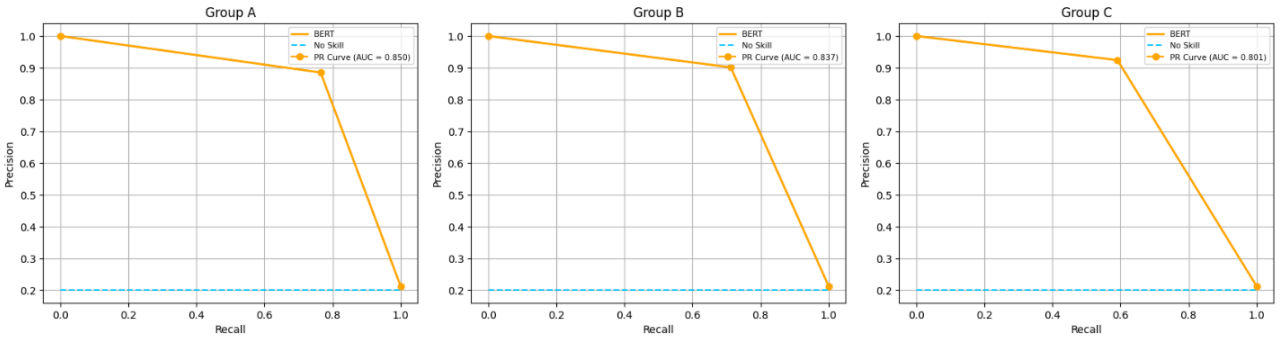


Fig. 11: PR curves for LSTM model using A, B, and C datasets.

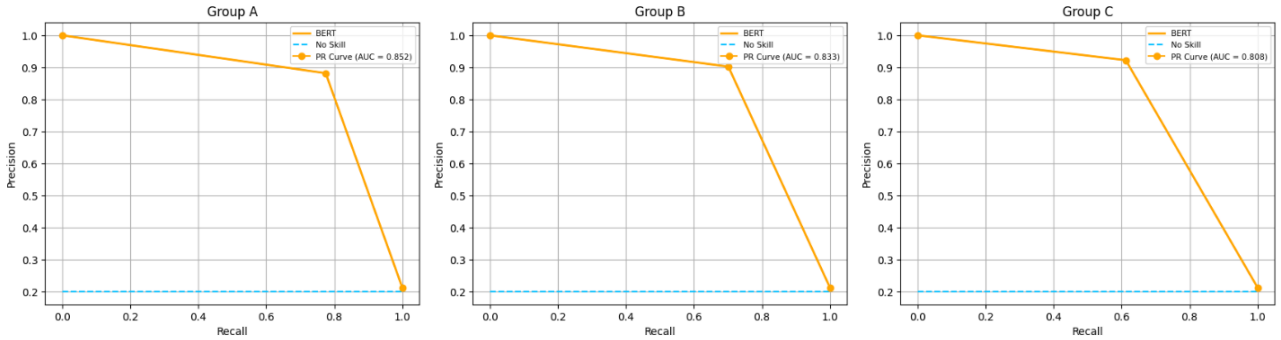


Fig. 12: PR curves for BiLSTM model using A, B, and C datasets.

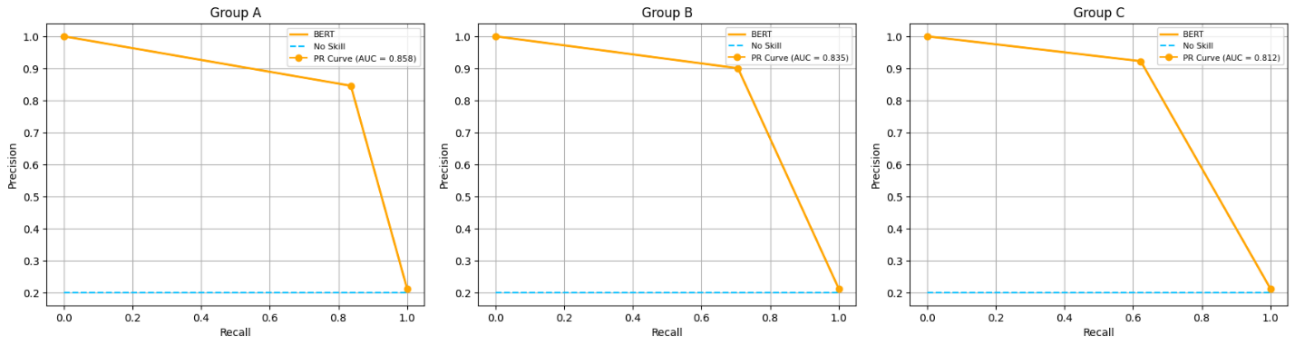


Fig. 13: PR curves for CBiLSTM model using A, B, and C datasets.

The comparative analysis of performance metrics, including Precision (ability to correctly identify positive cases), Recall (ability to capture all positive causes), F1-scores (a balanced measure of precision and recall), and F1-weighted scores, between the proposed CBiLSTM model and baseline models is presented in Tables 7, 8, and 9 for datasets A, B, and C, respectively. Notably, CBiLSTM outperforms the baseline models in all three scenarios.

B. Performance Comparison: Balanced vs imbalanced datasets

To investigate the effect of dataset balance, we compared the model's performance on both imbalanced and balanced datasets. Notably, balancing the data using BERT augmentation led to a significant improvement in model performance, as detailed in Table 4. This finding supports the notion that balanced datasets can enhance the effectiveness of deep learning models for binary classification tasks.

Table 4: Performance comparison of the CBiLSTM model on imbalanced vs. BERT-balanced datasets for Group A, B, and C. Results reflect class-specific precision, recall, F1-scores, and weighted F1 under identical training settings

Dataset	Urgent			Not Urgent			Weight F1 (%)
	P (%)	R (%)	F1 (%)	P (%)	R (%)	F1 (%)	
Imbalanced (A)	85.2	82.6	83.8	93.3	96.5	94.8	92.5
Imbalanced (B)	84.4	76.8	80.4	90.4	93.1	91.7	89.3
Imbalanced (C)	75.7	84.3	79.7	93.6	90.8	92.1	89.5
Balanced (A)	84.6	83.6	84.1	95.6	95.9	95.7	**93.3**
Balanced (B)	86.8	77.4	81.8	91.5	95.4	93.4	**90.2**
Balanced (C)	76.4	85.2	80.6	95.5	92.2	93.8	**90.9**

C. Optimizing CNN architecture

To extract key local features from the text data, a Convolutional Neural Network (CNN) was employed within the proposed model. The effect of varying the CNN's architecture on performance was investigated by adjusting the number of layers, kernel sizes, and filter counts. The impact of these changes on the model performance is detailed in Table 5.

D. Optimizing BiLSTM architecture

To capture long-range dependencies and contextual information within the text, a Bidirectional Long Short-Term Memory network (BiLSTM) was incorporated into the proposed model. This layer facilitates the extraction of global features by processing the text in both directions.

We explored the impact of varying the BiLSTM architecture by adjusting the number of layers and hidden units. The detailed results of this investigation are shown in Table 6.

E. Performance Comparison: CBiLSTM vs State-of-the-art Models

The model's performance is compared with state-of-the-art algorithms employing three distinct A, B, and C datasets. [8] presented a MOOC post-classification model using TF, linguistic attributes, metadata, and AdaBoost. TF lacks consideration for word order, potentially affecting context comprehension. Linguistic features, derived from LIWC, may be impacted by misspellings and symbols, potentially influencing AdaBoost's performance. [14] proposed a method that incorporated Google News embeddings, metadata features, and a CNN-Bi-GRU architecture. Google-news embeddings capture a word's fundamental meaning, disregarding its contextual intricacies.

Table 5: Impact of varying CNN depth, kernel configurations, & filter sizes on model performance using Group a dataset

CNN Kernel	Number of CNN filters	Urgent			Not Urgent			Weight F1 (%)
		P (%)	R (%)	F1 (%)	P (%)	R (%)	F1 (%)	
CNN – 2	64	81.6	82.1	81.8	94.1	94.3	94.1	91.5
CNN – 3	64	82.5	83.2	82.8	94.4	94.2	94.2	91.8
CNN – 4	64	82.1	82.8	82.9	95.3	94.9	95	92.5
CNN – 5	64	82.4	82.5	82.4	95	94.1	94.5	91.9
CNN – 6	64	81.8	82.2	81.9	94.7	94.2	94.4	91.8
CNN – 2 – 3	64 – 64	82.1	82.6	82.3	95.4	94.5	94.9	92.2
CNN – 3 – 4	64 – 64	83.7	81.9	82.7	96.3	93.4	94.8	92.2
CNN – 4 – 5	64 – 64	82.9	82.1	82.4	95.8	94.5	95.1	92.4
CNN – 5 – 6	64 – 64	82.8	81.8	81.8	95.3	94.9	95	92.2
CNN – 2 – 3 – 4	64 – 64 – 64	83.1	83.2	83.1	95.6	94.1	94.8	92.3
CNN – 3 – 4 – 5	64 – 64 – 64	83.8	83.1	83.4	95.1	95	95	92.5
CNN – 4 – 5 – 6	64 – 64 – 64	83	82.2	82.5	95.7	94.5	95	92.4
CNN – 2 – 3 – 4 – 5	64 – 64 – 64 – 64	81.1	81.9	81.4	94.3	92.8	93.5	91
CNN – 3 – 4 – 5 – 6	64 – 64 – 64 – 64	81.3	82.1	81.6	94.4	93.4	93.8	91.3
CNN – 2	128	82.9	82.7	82.7	95.3	94.1	94.6	92.1
CNN – 3	128	83.4	83.1	83.2	95.1	94.8	94.9	92.4
CNN – 4	128	82.7	83.1	82.8	95.2	95.3	95.2	92.6
CNN – 5	128	82.9	83.2	83	95.7	94.4	95	92.5
CNN – 6	128	82.2	82.8	82.4	95.1	94.1	94.5	92
CNN – 2 – 3	128 – 128	82.8	83.2	82.9	95.5	95.2	95.3	92.7
CNN – 3 – 4	128 – 128	84.1	82.6	83.3	**96.6**	94.6	95.5	92.9
CNN – 4 – 5	128 – 128	83.7	82.9	83.2	96.3	94.7	95.4	92.9
CNN – 5 – 6	128 – 128	83.4	82.4	82.8	96.1	94.8	95.4	92.7
CNN – 2 – 3 – 4	128 – 128 – 128	83.8	83.5	83.6	96.2	94.4	95.2	92.8
CNN – 3 – 4 – 5	128 – 128 – 128	**84.6**	**83.6**	**84.1**	95.6	**95.9**	**95.7**	**93.3**
CNN – 4 – 5 – 6	128 – 128 – 128	83.7	82.9	83.2	95.9	95.1	95.5	92.9
CNN – 2 – 3 – 4 – 5	128 – 128 – 128 – 128	81.6	82.2	81.8	94.6	93.1	93.8	91.3
CNN – 3 – 4 – 5 – 6	128 – 128 – 128 – 128	81.8	82.4	82.1	94.3	93.8	94	91.5
CNN – 2	256	82.4	82.5	82.4	94.9	93.6	94.2	91.7
CNN – 3	256	82.8	83.1	82.9	94.7	94.2	94.4	92
CNN – 4	256	82.4	83.2	82.7	95.1	95	95	92.4
CNN – 5	256	83.1	82.6	82.8	95.2	93.9	94.5	92
CNN – 6	256	81.7	82.5	82.1	94.4	93.8	94.1	91.5
CNN – 2 – 3	256 – 256	82.1	82.9	82.4	95.1	95	95	92.3
CNN – 3 – 4	256 – 256	83.8	82.1	82.9	95.9	94.2	94.9	92.4
CNN – 4 – 5	256 – 256	83.2	82.5	82.8	96.1	94.3	95.1	92.5
CNN – 5 – 6	256 – 256	83.5	81.9	82.6	95.5	94.1	94.7	92.2
CNN – 2 – 3 – 4	256 – 256 – 256	83.2	83	83	95.8	94.2	94.9	92.4
CNN – 3 – 4 – 5	256 – 256 – 256	83.9	83.1	83.4	94.8	95.1	94.9	92.5
CNN – 4 – 4 – 6	256 – 256 – 256	82.9	82.3	82.5	95.4	94.6	94.9	92.3
CNN – 2 – 3 – 4 – 5	256 – 256 – 256 – 256	82.1	81.8	81.9	94.1	93	93.5	91
CNN – 3 – 4 – 5 – 6	256 – 256 – 256 – 256	81.4	81.9	81.6	93.8	93.4	93.5	91.1

Table 6: Effect of BiLSTM architecture depth and hidden-unit configuration on performance using Group A dataset. Results demonstrate how sequential modeling capacity influences contextual feature extraction

Number of Layers	Number of hidden units	Urgent			Not Urgent			Weight F1 (%)
		P (%)	R (%)	F1 (%)	P (%)	R (%)	F1 (%)	
1	64	80.7	**85.8**	83.1	94.7	94.4	94.5	92.1
2	64 – 64	82.4	84.2	83.2	95.3	94.5	94.8	92.4
3	64 – 64 – 64	82.6	84.8	83.6	95.5	94.3	94.8	92.5
4	64 – 64 – 64 – 64	81.8	82.4	82.1	94.6	94.1	94.3	91.7
1	128	81.3	85.4	83.2	95.2	94.6	94.8	92.4
2	128 – 128	82.7	84.3	83.4	95.6	94.3	94.9	92.5
3	128 – 128 – 128	82.6	84.4	83.4	95.3	94.9	95	92.6
4	128 – 128 -128 – 128	81.9	83.7	82.7	94.3	94.1	94.1	91.7
1	256	83.1	83.4	83.2	94.6	94.5	94.5	92.1
2	256 – 256	82.8	83.3	83	94.9	95.4	95.1	92.5
3	256 – 256 – 256	80.4	85.4	82.8	95.4	94.3	94.8	92.2
4	256 – 256 – 256 – 256	79.9	85.2	82.4	95.1	93.4	94.2	91.7
1	512	82.2	84.1	83.1	93.8	94.6	94.1	91.8
2	512 – 512	81.4	83.9	82.6	95.1	94.3	94.6	92.1
3	512 – 512 – 512	81.1	84.2	82.6	94.6	96.1	95.3	92.6
4	512 – 512 – 512 – 512	80.3	84.8	82.4	93.4	**96.2**	94.7	92.1
2	64 – 128	83.9	82.5	83.1	95.2	95.5	95.3	92.7
2	128 – 256	**84.7**	82.9	83.7	95.7	95.2	95.4	92.9
2	256 – 512	84.2	82.9	83.5	95.3	95.4	95.3	92.8
3	64 – 128 – 256	84.6	83.6	**84.1**	95.6	95.9	**95.7**	**93.3**
3	128 – 256 – 512	84.3	83.1	83.6	95.1	95.6	95.3	92.8
4	64 – 128 – 256 – 512	83.4	82.8	83	94.6	95.2	94.8	92.3

The CNN and Bi-GRU layers within the architecture were utilized to extract long-term dependencies within post text.

[15] introduced a model that utilized preprocessing, BERT embeddings, and Bi-GRU techniques. Preprocessing encompassed the removal of stop words and special characters, which can enhance contextual comprehension but may detrimentally, affect classification accuracy.

The Bi-GRU component formed the classification model and emphasized its ability to extract long-term dependencies among words effectively.

Recently, [12] introduced a four-stage model that includes coding and vectorization using pre-trained BERT, a feature aggregation model to capture data-based relationships, a CNN-based model for improved text understanding, and classification of post-text using composite features.

Table 7: Performance comparison between the CBiLSTM and state-of-the-art models using Group A (baseline scenario with full cross-course independence). Metrics include class-wise and weighted F1 scores

Model	Urgent			Not Urgent			Weight F1 (%)
	P (%)	R (%)	F1 (%)	P (%)	R (%)	F1 (%)	
Adaboost [8]	77	65	70	91	95	93	88
Geo et al. [14]	83.4	77.2	80.1	94.8	95.4	95.1	91.8
Bi-GRU [15]	80.8	81.5	81.2	94.9	94.7	94.8	91.9
BERT + CNN Agg [12]	83.6	83	83.3	95.3	95.5	95	92.7
CNN	80.7	80.4	80.5	94.7	94.8	94.8	91.8
LSTM	82.2	80.7	81.4	94.7	95.2	95	92.1
Bi-LSTM	83.7	81.3	82.5	94.9	95.6	95.2	92.5
CNN + BiLSTM	**84.6**	**83.6**	**84.1**	**95.6**	**95.9**	**95.7**	**93.3**

Table 8: Performance comparison across models under Group B (cross-course evaluation), where several courses are held out entirely for testing. This table emphasizes generalization to unseen courses

Model	Urgent			Not Urgent			Weight F1 (%)
	P (%)	R (%)	F1 (%)	P (%)	R (%)	F1 (%)	
Adaboost [8]	80	65	72	90	95	92	88
Geo et al. [14]	80.7	79.7	80.2	93.8	**95.8**	**94.8**	**91.3**
Bi-GRU [15]	76.0	**84.7**	80.1	**95.7**	92.7	94.2	91
BERT + CNN Agg [12]	81.6	80.7	80.9	93.2	93.3	93.2	90
CNN	83.3	73.8	78.3	90.3	94.3	92.2	88.3
LSTM	85.7	77.8	81.5	91.9	95.1	93.5	90.2
Bi-LSTM	85.1	76.8	80.7	91.6	94.9	93.2	89.8
CNN + BiLSTM	**86.8**	77.4	**81.8**	91.5	95.4	93.4	90.2

Table 9: Performance comparison for Group C (cross-domain evaluation), where the Humanities domain is held out for testing. Results highlight model robustness across domain shifts

Model	Urgent			Not Urgent			Weight F1 (%)
	P (%)	R (%)	F1 (%)	P (%)	R (%)	F1 (%)	
Adaboost [8]	80	57	67	87	**95**	91	85
Geo et al. [14]	**80.7**	73.1	76.7	90.7	94.5	92.6	88.4
Bi-GRU [15]	76.1	83.1	79.4	94.7	92.1	93.4	90
BERT + CNN Agg [12]	72.4	**86.2**	78.7	**96**	91.1	93.5	90.3
CNN	74.2	83.5	78.6	94.9	91.3	93.1	89.7
LSTM	73.1	85.1	78.6	95.5	91.1	93.2	90
Bi-LSTM	73.1	85.5	78.8	95.7	91.1	93.3	90.1
CNN + BiLSTM	76.4	85.2	**80.6**	95.5	92.2	**93.8**	**90.9**

Despite the endeavors of state-of-the-art models, the highest attained F1-weighted scores stand at 92.7%, with an 83.3% F1 score specifically for urgent class classification, as depicted in Tables 7, 8, and 9. Notably, none of these investigations have addressed the issue of dataset imbalance within the StanfordMOOC Posts dataset, a factor known to impact classification performance and potentially introduce bias favoring larger classes [17], [18].

Our proposed model addresses imbalance by balancing the training dataset using a pre-trained BERT model which significantly influenced the classification performance. Our proposed model has achieved a better result, boasting a 93.3% F1-weighted score and 84.1% F1 score for the urgent class, as demonstrated in Table 7. This represents a notable improvement of 0.6% and 0.8% over the best-performing state-of-the-art model using the group A dataset. Moreover, as shown in Table 8, on the group B dataset, our model obtained an F1-urgent score boost of 1.6% compared to [14] and a 0.9% gain over the state-of-the-art model in [12]. Additionally, on the group C dataset, the CBiLSTM model outperformed the best state-of-the-art model [12] by 0.6% in overall F1-weighted score, as well as [14] by 1.2% in the urgent post classification.

CBiLSTM model outperformed state-of-the-art algorithms using various test scenarios, and obtained higher weighted F1 scores and a balanced precision-recall for urgent posts classification (see Tables 7, 8, and 9).

Although the CBiLSTM model does not include explicit attention mechanisms, qualitative analysis of the learned features reveals patterns aligned with human-understandable urgency cues. The CNN layers capture local discriminative n-gram patterns, while the BiLSTM highlights longer contextual dependencies. In particular, the model assigns higher importance to terms expressing temporal sensitivity (e.g., “urgent,” “immediately,” “deadline”), failure states (e.g., “stuck,” “error,” “not working”), and help-seeking cues (e.g., “please advise,” “need help”). These insights suggest that the model’s predictions are consistent with domain knowledge, reinforcing confidence in the classification results [7], [15].

The performance gains of the proposed model can be attributed to the complementary contributions of its three main components. Empirically, the BERT embeddings provided the largest single boost in accuracy because they supply rich contextual representations that reduce ambiguity in short and noisy MOOC posts. The CNN layers then enhanced performance by extracting stable local features that improved robustness to spelling variations and informal phrasing. Finally, the BiLSTM component contributed additional improvements by modeling sequence-level dependencies that are essential for distinguishing subtle urgency patterns spread across multiple clauses. Together, these components form a hierarchy in which BERT strengthens semantic understanding, CNN stabilizes local features, and BiLSTM refines global context. Despite these benefits, the architecture introduces limitations such as increased computational cost and dependency on pre-trained language models, which may be constraining for low-resource deployment scenarios.

Future Work

The proposed CBiLSTM model which incorporated a contextual embedding layer (BERT) and BERT-based data augmentation, produced a better result within the StanfordMOOC Posts datasets. Nevertheless, opportunities remain for further improving the model's performance. In particular, the contextual understanding of MOOC forum posts could be enhanced by leveraging advanced NLP techniques, including transformer-based architectures, to capture richer contextual information and provide more contextually relevant responses. Second, there is room for innovation in data augmentation, especially for scenarios with limited urgent posts. Exploring novel techniques to diversify and refine synthetic data can lead to more robust classification models.

Conclusion

This paper presented a novel approach to classify urgent MOOC forum posts and addressed the challenges posed by the increasing volume of students and posts in online courses. Our model, CBiLSTM, leverages the power of BERT-based contextual embeddings and a hybrid architecture that integrates CNN with BiLSTM layers. Our method includes a multi-stage process, from data preprocessing to the integration of course metadata, and accurately tackles the issue of data imbalance through BERT-based data augmentation. Balancing the dataset through BERT-based augmentation played an important role in enhancing the model's performance. This step effectively addressed the challenges posed by class imbalance and led to more accurate and robust classification results. We have shown the effectiveness of our model by comparing it to baseline models, such as standalone CNN, LSTM, and BiLSTM, as well as state-of-the-art approaches in the domain. Across various datasets representing different scenarios, our CBiLSTM model outperformed baseline models and demonstrated remarkable improvements over existing state-of-the-art algorithms. Notably, our model obtained higher F1-weighted scores and balanced precision recall for urgent post-classification.

Furthermore, the proposed framework demonstrates strong potential for deployment in real MOOC platforms, where timely identification of urgent posts can significantly enhance learner support and instructor responsiveness.

Author Contributions

Mujtaba Sultani: Conception, Data Curation, Methodological Approach, Validation, Formal Analysis, Investigation, Drafting the Main Manuscript.

Negin Daneshpour: Conception, Methodological Approach, Manuscript Review & Editing, Project Management.

Acknowledgment

The authors would like to thank Shahid Rajaee Teacher Training University (SRTTU) for their support and providing the opportunity to publish this research work.

Conflict of Interest

The authors state that there are no conflicts of interest related to this publication.

Abbreviations

AUC	Area Under the Curve
BERT	Bidirectional Encoder Representations from Transformers
Bi-LSTM	Bidirectional Long Short-Term Memory
CBiLSTM	Convolutional Neural Network + Bidirectional Long Short-Term Memory
CNN	Convolutional Neural Network
DA	Data Augmentation
GBDT	Gradient Lifting Decision Tree
LSTM	Long Short-Term Memory
MOOCs	Massive Open Online Courses
NLP	Natural Language Processing
SVM	Support Vector Machine
URLs	Uniform Resource Locators

References

[1] C. Zhenghao, B. Alcorn, G. Christensen, N. Eriksson, D. Koller, E. J. Emanuel "Who's benefiting from MOOCs, and why," Harvard Business Review. 25(1): 2-8, 2015.

[2] D. Shah, "A Decade of MOOCs: A Review of Stats and Trends for Large Scale Online Courses in 2021," EdSurge, 2021.

[3] D. Shah, "Monetization over massiveness: Breaking down MOOCs by the numbers in 2016," Class Central, 2016.

[4] C. Zhang, H. Chen, C. W. Phang, "Role of instructors' forum interactions with students in promoting MOOC continuance,'' J. Global Inf. Manage., 26(3): 105–120, 2018.

[5] Y. Feng, D. Chen, Z. Zhao, H. Chen, P. Xi, "The impact of students and TAs' participation on students' Academic performance in MOOC," in Proc. 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM): 1149–1154, 2015.

[6] K. S. Hone, G. R. El Said, "Exploring the factors affecting MOOC retention: A survey study," Comput. Educ. 98: 157–168, 2016.

[7] M. Sultani, N. Daneshpour, "Extracting urgent questions from mooc discussions: A BERT-based multi-output classification approach," Arabian J. Sci. Eng., 50: 1169-1190, 2025.

[8] O. Almatrafi, A. Johri, H. Rangwala, "Needle in a haystack: Identifying learner posts that require urgent response in MOOC discussion forums,'' Comput. Educ. vol. 118, pp. 1–9. doi: 10.1016/j.compedu.2017.11.002

[9] A. Agrawal, J. Venkatraman, S. Leonard, A. Paepcke, "YouEDU: Addressing confusion in mooc discussion forums by recommending instructional video clips," Int. Edu. Data Mining Soc., 2015.

[10] C. y. Chang, S. J. Lee, C. H. Wu, C. F. Liu, C. K. Liu, "Using word semantic concepts for plagiarism detection in text documents," Inf Retr. J., 24: 298-321, 2021.

[11] J. Xue, Y. Chen, "The principle and implementation of sentiment analysis system," in Proc. International Conference on Artificial Intelligence and Security: 28-39, 2022.

[12] M. A. El-Rashidy, A. Farouk, N. A. El-Fishawy, H. K. Aslan, N. A. Khodeir, "New weighted BERT features and multi-CNN models to enhance the performance of MOOC posts classification," Neural Comput. Appl., 35: 18019-18033, 2023.

[13] X. Sun, S. Guo, Y. Gao, J. Zhang, X. Xiao, J. Feng, "Identification of urgent posts in MOOC discussion forums using an improved RCNN," in 2019 IEEE World Conference on Engineering Education (EDUNINE): 1-5, 2019.

[14] Z. X. Guo, X. Sun, S. X. Wang, Y. Gao, J. Feng, "Attention-based character-word hybrid neural networks with semantic and structural information for identifying of urgent posts in MOOC discussion forums," IEEE Access, 7: 120522-120532, 2019.

[15] N. A. Khodeir, "Bi-GRU urgent classification for MOOC discussion forums based on BERT," IEEE Access. 9: 58243-58255, 2021.

[16] F. Almeida, G. Xexéo, "Word embeddings: A survey," arXiv preprint arXiv:1901.09069, 2019.

[17] J. Wei, K. Zou, "Eda: Easy data augmentation techniques for boosting performance on text classification tasks," arXiv preprint arXiv:1901.11196, 2019.

[18] X. Guo, Y. Yin, C. Dong, G. Yang, G. Zhou, "On the class imbalance problem," in Proc. 2008 Fourth International Conference on Natural Computation: 192-201, 2008.

[19] D. Ramyachitra, P. Manikandan, "Imbalanced dataset classification and solutions: A review," Int. J. Comput. Bus. Res. (IJCBR), 5(4), 1-29, 2014.

[20] T. Luis, B. Paula, R. P. Ribeiro, "A survey of predictive modeling under imbalanced distributions," ACM Comput. Surv. 49(2): 1-50, 2016.

[21] V. Kumar, A. Choudhary, E. Cho, "Data augmentation using pre-trained transformer models," arXiv preprint arXiv:2003.02245, 2021.

[22] M. Birjali, M. Kasri, A. Beni-Hssane, "A comprehensive survey on sentiment analysis: Approaches, challenges and trends," Knowledge-Based Syst., 226, 107134, 2021. doi: 10.1016/j.knosys.2021.107134

[23] T. R. Liyanagunawardena, A. A. Adams, S. A. Williams, "MOOCs: A systematic study of the published literature 2008-2012," Int. Rev. Res. Open Distrib. Learn., 14(3): 202-227, 2013.

[24] A. Bakharia, "Towards cross-domain MOOC forum post classification," in Proc. the third (2016) ACM Conference on Learning @ Scale: 253-256), 2016.

[25] L. Feng, G. Liu, S. Luo, S. Liu, "A transferable framework: Classification and visualization of mooc discussion threads," in Proc. Neural Information Processing: 377-384, 2017.

[26] X. Wei, H. Lin, L. Yang, Y. Yu, "A convolution-LSTM-based deep neural network for cross-domain MOOC forum post classification," Information. 8(3), 92, 2017.

[27] Y. Cui, A. F. Wise, "Identifying content-related threads in MOOC discussion forums," in Proc. the Second (2015) ACM Conference on Learning @ Scale: 299-303, 2015.

[28] A. Agrawal, A. Paepcke, "The stanford MOOC Posts dataset," Accessed: Dec, 15, 2020.

[29] S. Shaikh, S. M. Daudpota, A. S. Imran, Z. Kastrati, "Towards improved classification accuracy on highly imbalanced text dataset using deep neural language models," Appl. Sci., 11(2), 869, 2021.

[30] N. V. Chawla, K. W. Bowyer, L. O. Hall, W. P. Kegelmeyer, "SMOTE: synthetic minority over-sampling technique," J. Artif. Intell. Res., 16: 321-357, 2002.

[31] H. He, Y. Bai, E. A. Garcia, S. Li, "ADASYN: Adaptive synthetic sampling approach for imbalanced learning," in Proc. 2008 IEEE International Joint Conference on Neural Networks (IEEE world congress on computational intelligence): 1322-1328, 2008.

[32] A. Almahairi, S. Rajeshwar, A. Sordoni, P. Bachman, "Courville A (2018) Augmented cyclegan: Learning many-to-many mappings from unpaired data," in Proc. International conference on machine learning: 195-204, 2018.

[33] J. Devlin, M. W. Chang, K. Lee, "Google, KT, Language, AI: BERT: pre-training of deep bidirectional transformers for language understanding," in Proc. NAACL-HLT: 4171-4186, 2019.

[34] Y. Luan, S. Lin, "Research on text classification based on CNN and LSTM," in Proc. 2019 IEEE International Conference on Artificial Intelligence and Computer Applications (ICAICA): 352-355, 2019.

[35] J. Du, C. M. Vong, C. P. Chen, "Novel efficient RNN and LSTM-like architectures: Recurrent and gated broad learning systems and their applications for text classification," IEEE Trans. Cybern., 51(3): 1586-1597, 2020.

[36] C. Olah, "Understanding lstm networks," available at: https://colah. github. io/posts/2015-08-Understanding-LSTMs. 2020.

[37] G. Liu, J. Guo, "Bidirectional LSTM with attention mechanism and convolutional layer for text classification," Neurocomputing. 337: 325-338, 2019.

[38] M. J. Anzanello, F. S. Fogliatto, "Learning curve models and applications: Literature review and research directions,'' Int. J. Ind. Ergonom. 41(5): 573–583, 2011.

[39] R. L. Abduljabbar, H. Dia, P. W. Tsai, "Unidirectional and bidirectional LSTM models for short-term traffic prediction," J. Adv. Transp., 1-16, 2021.

[40] S. Kobayashi, "Contextual augmentation: Data augmentation by words with paradigmatic relations," arXiv:1805.06201, 2018.

[41] J. Wei, K. Zou, "EDA: Easy data augmentation techniques for boosting performance on text classification tasks," arXiv:1901.11196, 2019.

Biographies

Mujtaba Sultani received his B.Sc. degree at Kandahar University, Afghanistan in 2019, and his M.Sc. degree at Shahid Rajaee Teacher Training University (SRTTU), Tehran in 2024. He is an assistant professor at Kabul Polytechnic University (KPU). His research interest primarily includes data mining, information and knowledge management, machine learning, natural language processing, sentiment analysis, and language comprehension.

- Email: Mujtaba.cs01@kpu.edu.af
- ORCID: 0009-0009-4267-703X
- Web of Science Research ID: NA
- Scopus Author ID: NA
- Homepage: NA

Negin Daneshpour is an Associative Professor in the faculty of Computer Engineering at Shahid Rajaee Teacher Training University, Tehran, Iran. She received her B.Sc. degree in Computer Engineering-Hardware from Shahid Beheshti University, Tehran, Iran, in 1999, and her M.Sc. and Ph.D. degrees in Computer Engineering - Software from Amir Kabir University of Technology, Tehran, Iran, in 2002 and 2010 respectively. Her research interests include data mining, data preprocessing, information management, and natural language processing.

- Email: ndaneshpour@sru.ac.ir
- ORCID: 0000-0003-3951-4060
- Web of Science Research ID: NA
- Scopus Author ID: NA
- Homepage: NA