Multi-label Text Classification on Textcnn fused Bilstm with Attention Mechanism

Aihua Duan*, RODOLFO C. RAGA JR

***Abstract* - This paper proposed a deep learning model for multi-label text classification to effectively manage and utilize the network text information and realize the automatic labeling of text content. The neural network word vector model GloVe is used to obtain the semantic features of the text data. The model fusion of recurrent neural network and convolutional neural network is performed, and the attention mechanism is introduced into BiLSTM to form the BiLSTM_Attention neural network model. Experimental results show that the BiLSTM_Attention model structure combines the advantages of the TextCNN model and can better understand the semantic information. The Attention mechanism is more reasonable for text feature extraction, so the model focuses on the features that contribute more to the text classification task, and the classification effect is better.**

***Index Terms*—Bi-LSTM; Attention; TextCNN; Multi-label**

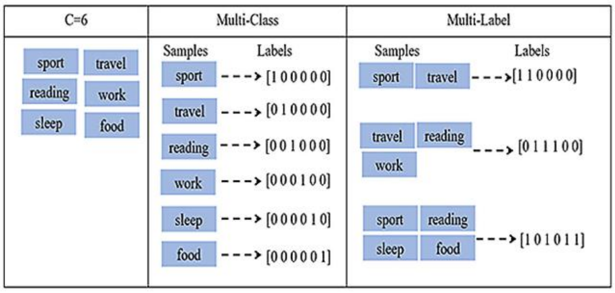
I. Introduction

In the contemporary age of information proliferation, individuals are inundated with copious amounts of textual information. To efficiently manage this textual data and assist individuals in accessing the information they require, the demand for text classification technology has escalated significantly. Multi-Label Text Classification (MLTC) technology has emerged as a pivotal area of research, encompassing various applications such as information retrieval [1], conversational behavior classification [2], topic recognition [3], emotion analysis [4], and question-answering systems [5].

As a significant and demanding endeavor in Natural Language Processing (NLP), MLTC has found extensive applications. For instance, a news article may encompass various topics simultaneously, such as technology, economics, and digital trends. This task involves assigning multiple labels to the text, making it more complex than single-label text classification. Table I illustrates both multi-category text classification and MLTC.

While multi-label classification is an interesting concept, its practical implementation is far from trivial and widely explored. The traditional approach to multi-label text classification is binary relevance (BR). This method transforms a multi-label learning problem into multiple (with the same number of categories) binary classification problems.

TABLE I Comparison Between Multi-class and Multi-label Text Classification



C=6	Multi-Class		Multi-Label	
	Samples	Labels	Samples	Labels
sport travel	sport	[1 0 0 0 0 0]	sport travel	[1 1 0 0 0 0]
reading work	travel	[0 1 0 0 0 0]	travel reading work	[0 1 1 1 0 0]
sleep food	reading	[0 0 1 0 0 0]	sport reading sleep food	[1 0 1 0 1 1]
	work	[0 0 0 1 0 0]		
	sleep	[0 0 0 0 1 0]		
	food	[0 0 0 0 0 1]		

This method has been developed to stack numerous binary classifiers into chains in a certain sequence to overcome the limitations of first-order methods employing binary classifiers. This method structures the chain structure based on information such as prior knowledge and label dependencies. Generally, each successive classifier is developed from the predictions of its predecessor, so establishing a methodology that can leverage higher-order label dependencies. However, the complicated structure of the classifier chain increases exponentially with the quantity of classes. Furthermore, the initial predictions are essential, as the chain structure establishes captured label dependencies that are significantly influenced by preceding predictions.

The other category is algorithmic adaptation. Modifying typical binary classifiers so that they can be directly utilized for multilabel issues. Examples include multi-label k-nearest neighbors, multi-label decision trees, and support vector machine ranking. However, most algorithmic adaptation methods are still inadequate, especially when compared to novel approaches based on deep learning, as they are limited to modeling first- or second-order label dependencies.

Currently, common models rely on deep-learning classification approaches. In contrast to standard machine learning approaches that involve manual feature extraction, deep learning enables computers to automatically learn and extract features, resulting in resource savings and higher performance, displaying exceptional advantages in jobs. While existing mainstream approaches effectively address feature extraction restrictions and unknown semantic linkages, the task of deeply examining global and local semantic relationships within text remains unresolved. Therefore, there is a compelling need to remedy model shortcomings in MLTC tasks and boost classification accuracy to permit practical implementations in scenarios such as text information categorization.

Manuscript received Dec 15, 2023; revised Dec 7, 2024.

Aihua Duan is a PhD candidate in Philosophy in Computer Science at the College of Computing and Information Technologies National University, Manila, 1008, Philippines. (e-mail: duana@students.national-u.edu.ph)

RODOLFO C. RAGA JR is a professor at the College of Computer Studies and Engineering José Rizal University, Mandaluyong City 1550, Philippines. (e-mail:rodolfo.raga@jru.edu)

This paper is organized as follows: In the next portion, suitable multi-label learning approaches are discussed from the literature. In the Methods section, this paper will review all the methods applied and introduce two kinds of error functions. In addition, this paper will explore the TextCNN embedding method, LSTM, which combines attention to obtain a representation of the document themes. In the next chapter, the paper presents experimental findings. Specifically, the paper analyzes (1) the merger of the TextCNN parallel and BiLSTM_Attention models. (2) performance comparison between eight models. (3) The influence of various gamma error function values on model outcomes (4) The precision, recall, and F1 of the six models mentioned under Micro and Macro in the paper.

II. Literature Review

The current state-of-the-art deep learning techniques for MLTC primarily involve Recurrent Neural Networks (RNNs), Convolutional Neural Networks (CNNs), and attention-based models. RNN-based text classification models treat text as a sequence of words and derive semantic features by considering the structural information in the sequence and the interdependence between contexts for downstream classifiers. However, conventional RNN models have limitations in effectively retaining long text sequences. Among the various RNN variants, Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) are the most commonly used model architectures designed to capture long-term dependencies more effectively. Yang [7] introduced the Sequential Generation Model (SGM) model, which utilized a Bidirectional Long Short-Term Memory (Bi-LSTM) network in the encoder structure and incorporated an out-of-order prediction module in the decoder structure to address the error accumulation issue resulting from sequential prediction in the Seq2Seq model. Concurrently, You[8] proposed the Attention Extreme Multi-Label (XML) model, which extracted text semantic features based on the Bi-LSTM network, enhanced text semantic features using the attention mechanism, and employed the label tree for grouping labels, thereby resolving the issue of excessive computation under the substantial quantity of labels.

RNNs are known for their ability to extract temporal features, whereas CNNs excel at capturing spatial features and local context. Kim [9] introduced a CNN-based text classification model (TextCNN) that incorporated a layer of multi-scale convolution following the training of word vectors based on the Word2Vec model, resulting in promising results. Subsequently, Kurata [13] enhanced CNN architecture by introducing a hidden layer and initializing the neural network using label co-occurrence information, which led to improved accuracy in multi-label classification compared to random initialization. This marked the first instance of incorporating co-occurrence information into a CNN.

Based on tALBERT-CNN, Liu [10] proposed a method for multi-text classification. The method uses LDA topic model and ALBERT model to obtain topic vectors as well as semantic context vectors for each word (document), adopts a certain fusion mechanism to obtain deep topics, and extracts semantic representations of the documents and multi-label features of the text via TextCNN model to train a multi-label classifier. Experimental results on a standard dataset show that the proposed method is feasible to extract multi-label features from documents and outperforms existing state-of-the-art text classification algorithms using multi-label methods.

Yang [11] designed a convolutional neural network (CNN) model based on threshold learning, used a convolutional neural network (CNN) as a feature extractor, and introduced two threshold learning mechanisms: adaptive threshold (AT) and implicit threshold (IT). Among them, AT-CNN uses an adaptive threshold to predict the confidence of each label based on different classes and uses this confidence as a threshold to select positive labels. IT-CNN uses an implicit threshold, predicts the number of positive labels in each sample, and selects the top k scores from the multi-label module as the final category. Method The proposed method achieves good results on the multi-label text classification task on the MIMIC-III database. AT-CNN and IT-CNN models outperform other baseline models in performance and run more efficiently.

Yang [12] proposed optimized Binary Relevance combined with the multi-label learning model of Convolutional Neural Networks (BR-CNN), this paper uses a variety of deep learning architectures including convolutional neural networks (CNN), Long Short-Term memory networks (LSTM), and Gated Recurrent units (GRU) and optimizes their BR transformations. This paper compares the deep learning BR method with the MLTC method based on label dependency information and the traditional BR method. It is found that BR-CNN has superior performance on four datasets of AAPD, Reuters-21578, MIMIC-III, and RCV1-v2.

Lu [13] proposed a CNN-Bi-LSTM-Attention model for Chinese short texts. They designed the method to extract the meaning of labels, the CNN layer to extract the local semantic features of the text, the BiLSTM layer to fuse the context features and local semantic features of the text, and the attention layer to select the most relevant features for each label. Experimental results show that the proposed method is effective under the commonly used multi-label evaluation metrics.

In addition to RNNs and CNNs, the study also suggests the utilization of Graph Neural Networks (GNNs) and Attention to explore relationships among words, documents, or tags for acquiring more comprehensive text features. Among the various types of GNNs, Graph Convolutional Networks (GCNs) and their variations are widely favored due to their efficiency and compatibility with other neural networks, leading to significant achievements in various applications. GCN functions as a convolution operation, leveraging the connections between neighboring nodes in the graph structure, as well as the dependency syntax tree or word co-occurrence information to capture pertinent internal text details.

Liang [14] proposed representation combines three different sources of information, namely the input text itself, label-to-text relevance, and label-to-label relevance. A dual attention mechanism is used to combine the first two information sources, and a graph convolutional network is used to extract the third information source, which is then used to help fuse the features extracted from the first two information sources. Extensive experiments are conducted on a public dataset of privacy leak posts on Twitter, and the results show that the proposed privacy leak detection method significantly and consistently outperforms other state-of-the-art methods in all key performance metrics.

Wu [15] proposes a multi-perspective contrast model (MPCM) based on the Attention mechanism to integrate labels and documents, and uses the contrast method to enhance the label information semantics and relevance perspective of two texts. We introduce contrastive global representation learning and positive label representation alignment techniques to improve the model's ability to perceive accurate labels. Experimental results show that the proposed algorithm achieves good results on AAPD and RCV1-V2 datasets.

To perform text classification more effectively, Liu [16] introduced a text-label joint attention mechanism. In this approach, their proposed representation combines three different sources of information, namely the input text itself, label-to-text relevance, and label-to-label relevance. A dual attention mechanism is used to extract the third information source, which is then used to help fuse the features extracted from the first two information sources. This model in two multi-label classification datasets (AAPD, Reuters-21578) demonstrated the superiority over the considered baseline methods.

Gao [17] proposed a label-aware network to obtain label relevance and text representation. Since two adjacent labels or words in the graph have similar relationships, and the structure of the graph is also conducive to the representation of labels, a heterogeneous graph is constructed from words and labels, and the label representation is learned using MAP-ath2vec. Each part of the text contributes differently to label inference, so bidirectional attention flow is exploited for label-aware text representation in two directions: from text to label and from label to text. Experimental evaluation shows that the proposed method outperforms various baseline methods on both offline benchmarks and real online systems.

Zhao [18] integrated variational continuous label distribution learning into MLTC models. This integration allows the attention to be directed towards the overall distribution of the complete label set, rather than concentrating only on specific labels with the highest response values. Consequently, this strategy effectively addresses the challenge of class imbalance.

Li [19] developed an Attention Network incorporating external knowledge, label embedding, and a comprehensive attention mechanism. Experimental results demonstrate that this approach surpasses the current state-of-the-art MLTC method.

III. Methodology

In this study, the Bi-LSTM Attention model and TextCNN model will be utilized to conduct MLTC experiments. The Bi-LSTM model is adept at handling sequential structures and can take into account the contextual information of the sentence, albeit with a trade-off in terms of overall processing speed. On the other hand, the TextCNN model is agnostic and exhibits a strong ability to extract surface-level textual features. TextCNN primarily extracts features utilizing a filter window, which may limit its long-distance modeling capability and may not be sensitive to word order. To address these individual limitations, this research proposes the fusion of Bi-LSTM with TextCNN. This integration introduces an attention mechanism that allows the model to concentrate on text features that significantly impact text classification results. The model structure diagram is depicted in Fig. 1.

A. *Embedding layer*

This study primarily focuses on the application of tokenizers to words, with prominent technologies including TF-IDF, word2Vec, GloVe, ELMo, and BERT.

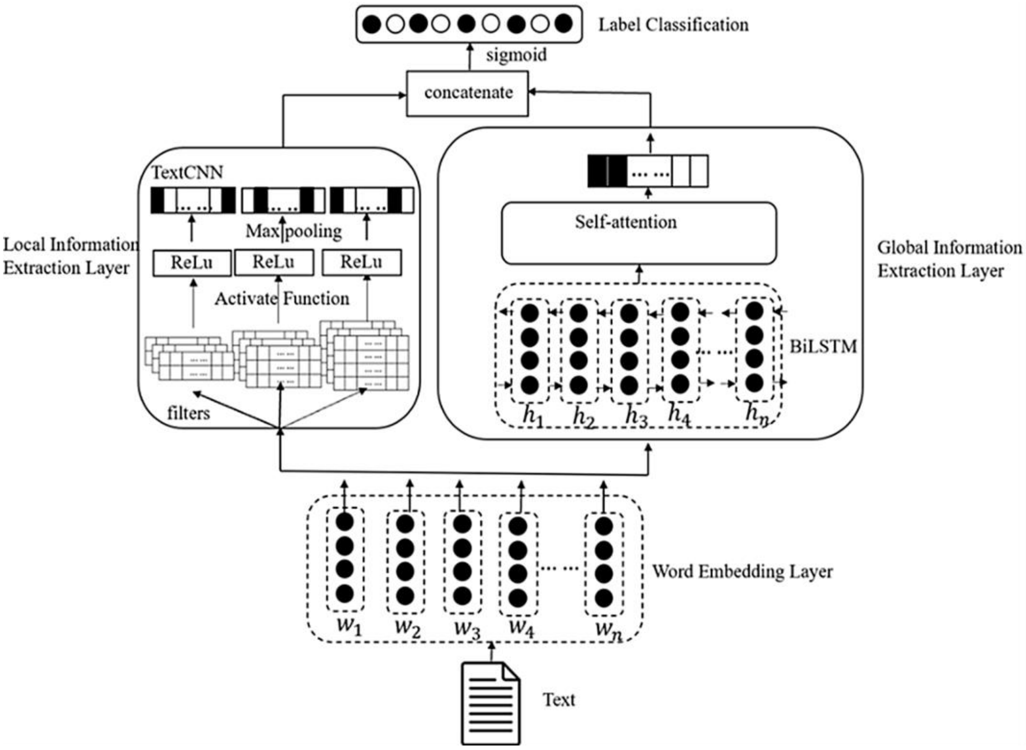


Fig. 1. Structure of Proposed Model

Various tokenizers exhibit different impacts on machine learning performance. GloVe offers several advantages compared to other word embedding techniques like Word2Vec. It excels in semantic and grammatical tasks and demonstrates superior ability in capturing linear word relationships. Furthermore, GloVe's training procedure is relatively simple and enables efficient training on extensive corpora.

B. Attention mechanism

The attention mechanism [8] serves as a method to address the challenge of mimicking human attention in swiftly identifying valuable information from a vast dataset. Given the constraints posed by computational power and optimization algorithms, integrating an attention mechanism can enhance the neural network model's capacity to manage information overload and enhance its information processing capabilities. Within the RNN model, this mechanism is employed to address the issue of information loss bottleneck resulting from the conversion of a lengthy sequence into a fixed-length vector (Fig. 2).

The Bi-LSTM model is employed to extract global features from the input data, with a focus on capturing semantic information within the text more effectively through context information. In the Bi-LSTM model, the current hidden layer h_{t-1} at time t is obtained through a weighted sum of the forward hidden layer h_t and the backward hidden layer h_{t-1}. The calculations are presented as follows:

$$\overrightarrow{h_t} = LSTM(x_t, \vec{h}_{t-1}) \qquad (1)$$

$$\overleftarrow{h_t} = LSTM(x_t, \overleftarrow{h}_{t-1}) \qquad (2)$$

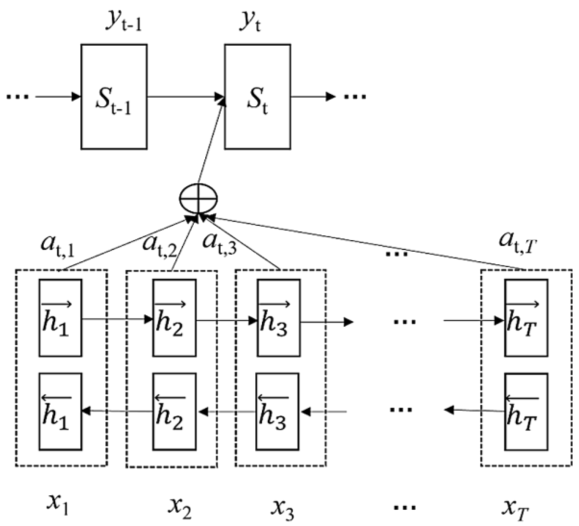


Fig. 2. Attention Mechanism with Bi-LSTM

$$h_t = w_t\overrightarrow{h_t} + v_t\overleftarrow{h_t} + b_t = [\overleftarrow{h_t}, \overrightarrow{h_t}] \quad (3)$$

where x_t is the input of the current hidden layers; $\vec{h}_{t-1}$ is the forward hidden layer state at time (t-1); $\overleftarrow{h}_{t-1}$ is the backward hidden layer state at time t-1 in Equation (2); w_t and v_t are the relative weight values of the pre-hidden layer and post-hidden layer corresponding to BiLSTM at time t, respectively, b_t is the bias value of the hidden layer state at time t in Equation (3).

The output matrix $\boldsymbol{H} = [h_1, h_2, \dots, h_t]$ of BiLSTM model is fitted into the hidden layer of the attention mechanism to obtain the attention initial state matrix $\boldsymbol{S} = [s_1, s_2, \dots, s_t]$. According to the importance of each feature in S, a weight is assigned to each feature, and the different weight coefficients a_t are multiplied and accumulated with their corresponding

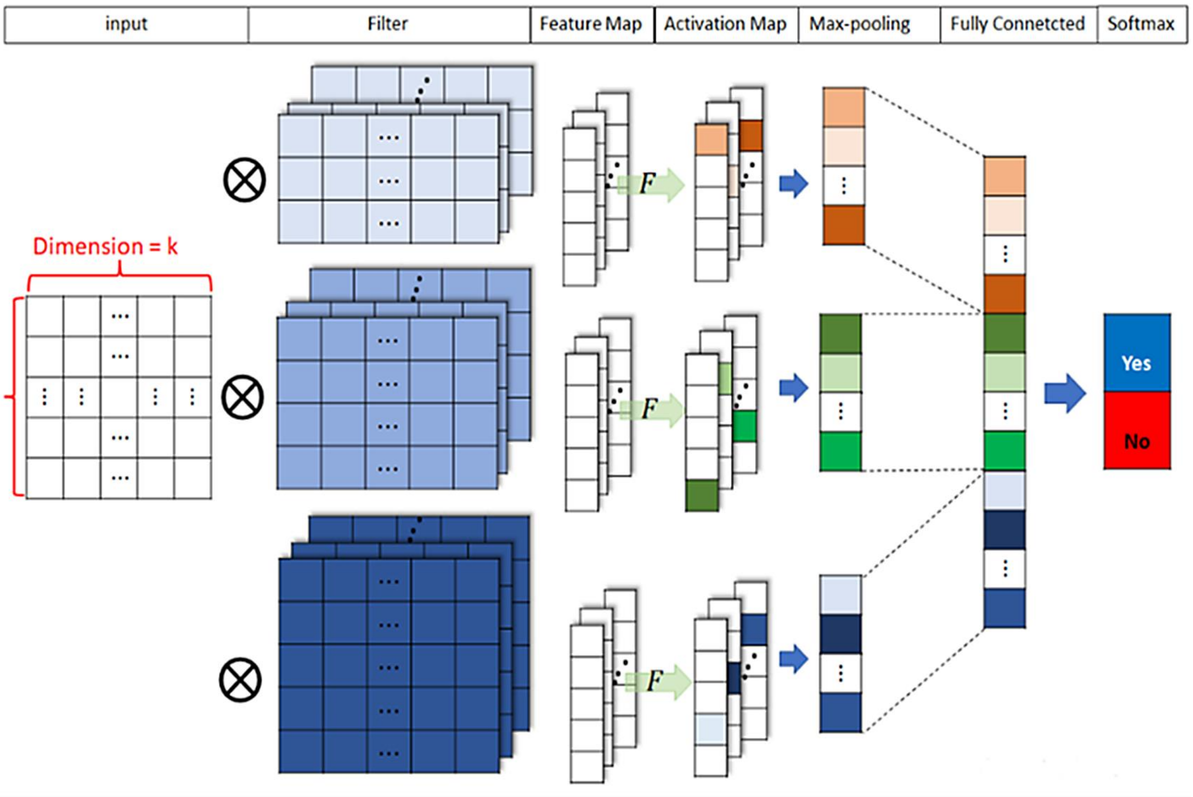


Fig.3. Diagram of TextCNN model

initial state vector in Equation (5). Subsequently, the output vector Y of the attention layer is obtained in Equation (6). The equation is expressed as follows:

$$e_t = \tanh(v_t S_t + b_t) \quad (4)$$

$$a_t = \frac{\exp(e_t)}{\sum_1^t \exp(e_i)} \quad (5)$$

$$\mathrm{Y} = \sum_{t=1}^{T} a_t S_t \quad (6)$$

where v_t is the weight matrix; b_t is the bias quantity; and e_t is the energy value determined by the state vector s_t in Equations (4) and (6).

The attention mechanism is a technique utilized in artificial neural networks to simulate cognitive attention. This technique assigns varying weights to different parts of the input data, enabling the neural network to concentrate on the most crucial aspects of the data. Primarily an RNN, it derives the primary meaning of the article by understanding the contextual connections within the article. LSTM model utilizes gate structures to replicate the forgetting and memory processes of the human brain, effectively mitigating issues such as gradient vanishing or exploding during prolonged sequence training. Additionally, the bidirectional LSTM enhances information retrieval capabilities.

In the process of text representation, the output vectors of each time step are directly summed and then averaged. This approach assumes equal importance for each input word in the text, which may not always hold. Proper allocation of attention resources is crucial when combining these output vectors, requiring different weights to be assigned to each vector to prioritize the most relevant classification results based on the text vector characteristics. The attention mechanism assigns a weight to each vector, enabling a weighted average of all output vectors. These weights are determined by the contribution of each term to the output result of the text content, thereby reducing the impact of irrelevant words and enhancing computational efficiency. Integrating the attention mechanism into the MLTC model can lead to a more comprehensive explanation of text features, ultimately improving the accuracy of classification results.

C. TextCNN

TextCNN represents a traditional text classification model. Kim employed various sizes of sliding windows for the convolutional pooling operation on the input text vector to capture local features of the text sequence for aggregation and filtering. Additionally, Kim extracted semantic information from the text at various levels of abstraction, resulting in a high-level feature vector representation of the text. The model's structure is illustrated in Fig. 3.

TextCNN consists of four main components: the input layer, convolutional layer, pooling layer, and output layer. In this model, for a text input of length n, the convolutional layer extracts text features by employing h sliding windows of varying sizes to convolve the text input vector. The convolutional feature values are generated by the convolution kernel at position i.

$$S = f(w \cdot T_{i:i+\hbar-1} + b), w \in R^{h \times k} \quad (7)$$

Where k is the word vector dimension corresponding to each word in the text sequence; w is the convolution kernel with dimension size h × k; $T_{i:\ i+h-1}$ is the sliding window consisting of row i to row i+h-1 of the input matrix; b is the bias parameter; and f is the nonlinear mapping function. The pooling layer uses a 1-MaxPool maximum pooling strategy to extract the maximum feature value from each sliding window.

$$C_i = \max\{S\} = \max(S_1, S_2, \ldots, S_{n-h+1}) \quad (8)$$

The concatenation layer combines all the pooled feature values to obtain the high-level feature vector of the text.

$$C = [C_1, C_2, \ldots, C_{n-h+1}], C \in R^{n-h+1} \quad (9)$$

Where n is the number of words in the text sequence and C is the text feature vector trained by the TextCNN module with a-dimension size of i+h-1. After completing the convolution pooling operation, the fully connected neural network layer is linked to the downstream task to facilitate the prediction of text labels.

If the task involves binary classification, the Softmax function is employed as the classification function (Fig. 3).

For MLTC, the sigmoid function is frequently utilized as the activation function for the output layer, while the binary Cross-Entropy (BCE) function is employed as the loss function. Specifically, the sigmoid activation function is applied to each output node in the final classification layer. Subsequently, the cross-entropy loss function is computed for each output node about its corresponding label. The equation can be expressed as follows:

$$BCE(x)_i = -y_i log f_i(x) + (1 - y_i)\log(1 - f_i(x))] \quad (10)$$

where *x* is the input; C is the number of classification classes; *i* is ranging from [1, C]; and y_i is the accurate label corresponding to the *i*th category.

For multi-label classification tasks, the utilization of Focal Loss (FL) has been shown to consistently exhibit enhanced performance in mitigating classification imbalances [20]. FL involves the multiplication of a modulating factor to BCE with a tunable focusing parameter γ ≥ 0. This approach assigns greater loss weight to instances that are challenging to classify, particularly those predicted with low probabilities compared to the ground truth [21]. In the context of multi-label classification, FL can be formally defined as follows:

$$L_{FL} = \begin{cases} -\alpha(1 - p_i^k)^{\gamma} \log(p_i^k) & if\ y_i^k = 1 \\ -\ (1 - \alpha)\ (p_i^k)^{\gamma} \log(1 - p_i^k) & otherwise \end{cases} \quad (11)$$

where α and γ are the coordinates to control.

The selected model evaluation metrics include accuracy and Micro-F1. The equation for accuracy is given as follows:

$$\text{Accuray} = \frac{TP + TN}{TP + TN + FP + FN} \quad (12)$$

True Positive (TP) refers to cases where the actual sample value is Positive, the sample is input into the prediction model, and the model output value is also Positive, indicating correct predictions by the classification model. True Negative (TN) occurs when the actual sample value is Negative, the sample is fed into the prediction model, and the model output value is also Negative, representing correct predictions. False Positive (FP) happens when the actual sample value is Negative, the sample is fed into the prediction model, and the model output value is Positive, indicating incorrect predictions. False Negative (FN) denotes cases where the actual sample value is Positive, the sample is fed into the prediction model, and the model output value is Negative.

In this paper, we use Micro/Macro-F1, Micro/Macro-P and Micro/Macro-R as the evaluation indicators for performance comparison, which are specifically defined as follows:

$$Micro - F1 = \frac{\sum_{i=1}^{C} 2TP_i}{\sum_{i=1}^{C} 2TP_i+FP_i+FN_i} \tag{13}$$

$$Macro - F1 = \frac{1}{C}\sum_{i=1}^{C} \frac{2TP_i}{2TP_i+FP_i+FN_i} \tag{14}$$

Where i denotes the ith class label, TP_i, FP_i, FN_idenote the true positive examples, false positive examples, and false negative examples, respectively.

$$Weigthted - F1 = \sum_{i=1}^{C} w_i\, F1_i \tag{15}$$

where w_i represents the proportion of class i in the total sample.

IV. Experiment

This paper divides the corpus into a training set, a validation set, and a test set. The training set is utilized for model training, the validation set aids in model selection and hyperparameter tuning, and the test set assesses the model's performance on unseen data (Fig. 4).

This study conducts experiments using the Rueters-21578 text classification corpus, comprising 10,788 British Reuters financial news texts that are partitioned into training and testing documents. The training set consists of 7,769 records, while the test set comprises 3,019 records. The document length ranges from a minimum of 11 words to a maximum of 8,459 words, with an average word count of 749 words per document. The dataset contains a total of 90 classes, with each document having a maximum of 15 labels and a minimum of one label. On average, each document is associated with 1.2336 labels. The complete dataset is presented in Table II.

TABLE II
Dataset Description

Data and Labels	Number
Training Data	6215
Validation Data	1554
Test Data	3019
Total labels	90
Avg label per text	1.2336

The labels of this dataset are severely imbalanced, and Figure 5 shows the number of samples on each label.

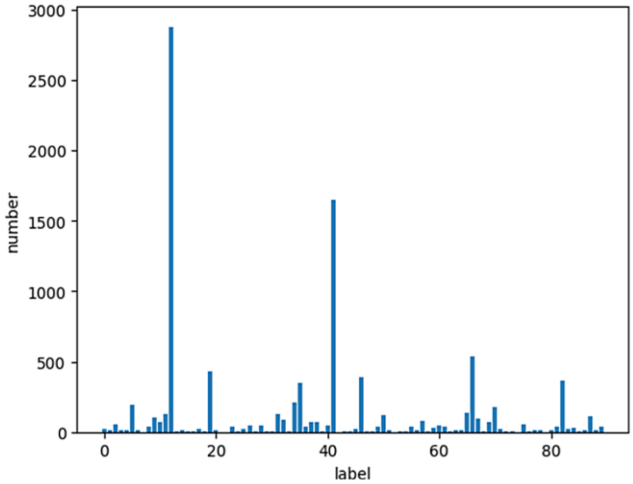


Fig.5. Distribution Diagram of Labels

The proposed model was developed using the Python language and the Tensorflow+Keras deep learning framework. Parameter selection plays a crucial role in deep learning models. TextCNN model utilizes three filters with sizes 3, 4, and 5. To mitigate overfitting, a dropout rate of 0.5 is employed, and the batch size is set to 30, which is more appropriate for facilitating the convergence of the model's gradient descent. The optimal

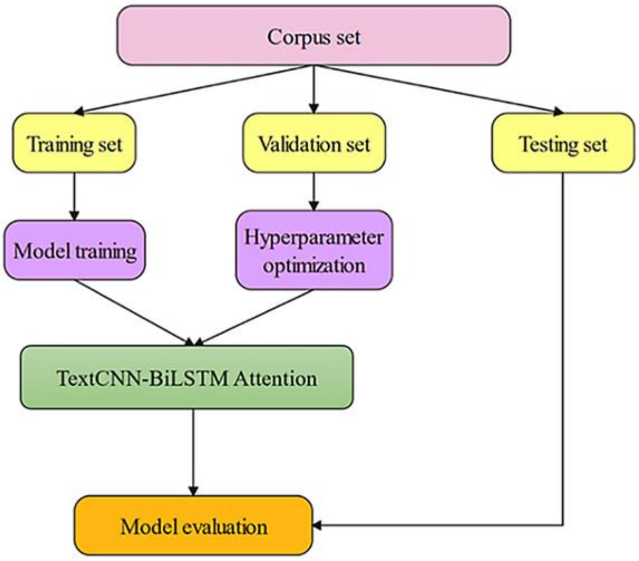


Fig. 4. Corpus Segmentation model

parameter configurations are outlined in Table III.

TABLE III
PARAMETERS SETTING OF TEXTCNN WITH BILSTM-ATTENTION MODEL

Parameters	Values
Word vector dimension	300
Max length of article	500
Filters in TextCNN	[3,4,5]
Dropout	0.5
Batch size	30
Epochs	100
Activate Function	Sigmoid
Loss	Focal Loss
Optimizer	Adam

The paper compares word embeddings using Word2Vec and GloVe, and models using Binary Relevance, Label Powerset, Classifier Chain, TextCNN, Bi-LSTM, and Bi-LSTM_Attention. The study concludes that combining GloVe word embeddings with the TextCNN parallel with the Bi-LSTM_Attention model yields the best training and test set results.

TABLE IV COMPARATIVE RESULTS OF DIFFERENT LEARNING MODELS

No.	Model	Accuracy
1	Tf-idf BinaryRelevance	46.59%
2	Tf-idf LabelPowerset	64.80%
3	Tf-idf ClassifierChain	46.91%
4	Vector TextCNN_	83.87%
5	Vector BiLSTM	82.68%
6	Vector TextCNN in parallel with BiLSTM	86.72%
7	GloVe TextCNN	84.89%
8	GloVe BiLSTM Attention	85.26%
9	GloVe TextCNN in parallel with BiLSTM Attention **(Proposed model)**	**87.88%**

Fig. 6 illustrates the variation in accuracy of six models across the training set. This research indicates that the Bi-LSTM model has significantly improved during training and can acquire global information.

However, it still exhibits limitations in capturing local information. TextCNN demonstrates a faster and more stable increase in classification accuracy, providing more detailed information in multi-label document text classification. Bi-LSTM_Attention model represents an enhancement of Bi-LSTM model. Combining it with the TextCNN model results in a rapid increase in accuracy and demonstrates superior generalization ability on the test set.

It was used in the test set to evaluate the model's generalization capability, incorporating various word vector embeddings and combining simple models such as TextCNN, Bi-LSTM, and Attention with Bi-LSTM. The results highlight the following key aspects:

(1) Bi-LSTM-Attention TextCNN model embedded with GloVe outperforms other experimental models, demonstrating its effectiveness for MLTC. Specifically, the accuracy of the GloVe_TextCNN_Bi-LSTM-Attention model is 0.0299 and 0.0262 higher than that of the standalone TextCNN and Bi-LSTM-Attention models, respectively.

(2) The experimental results indicate that using FL is more advantageous than using BCE within the same model. FL prioritizes difficult samples, addressing low classification accuracy in categories with fewer samples. Importantly, FL improves overall model performance by mitigating the issue of imbalanced samples, not just those

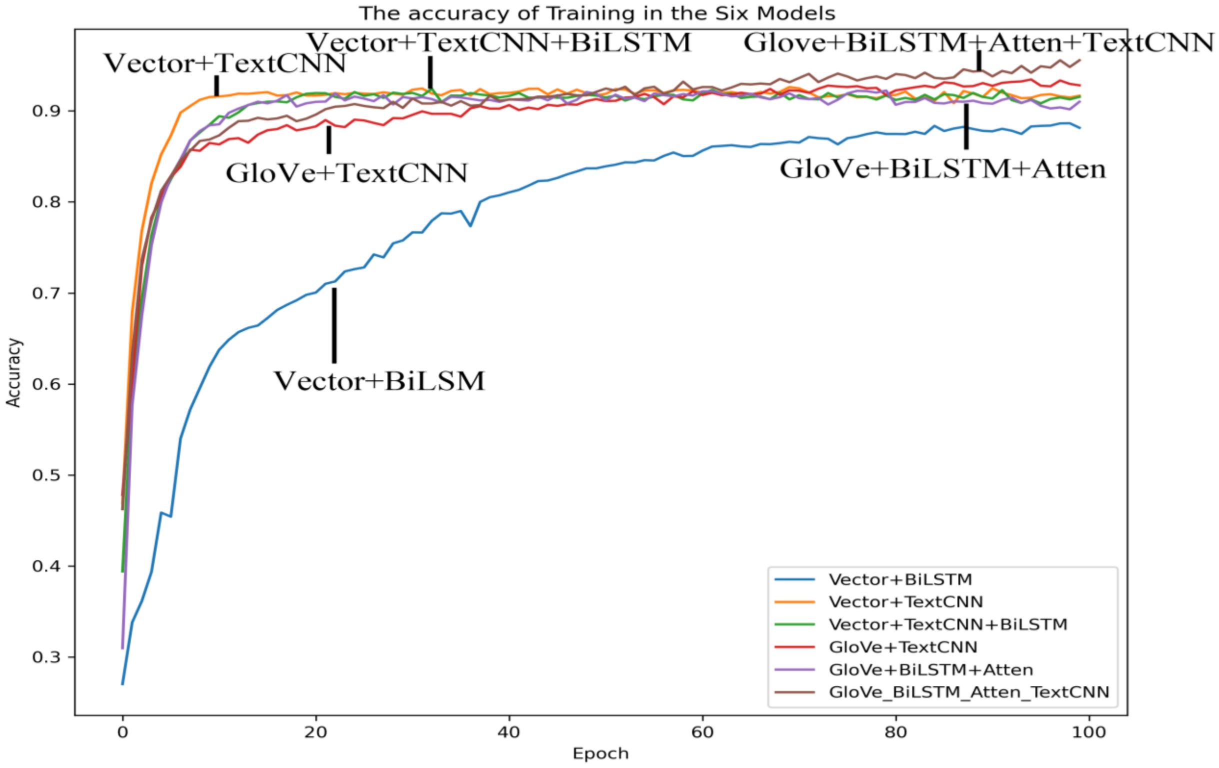


Fig. 6. Training Data Accuracies of Six Models

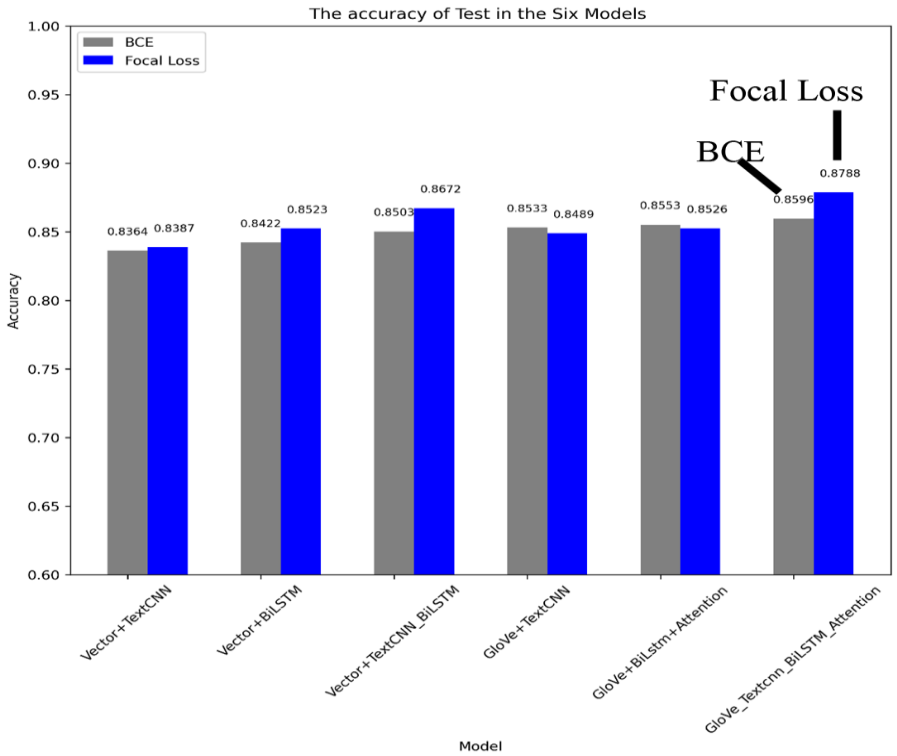


Fig. 7. Test Data Accuracy Across Six Models

with fewer instances.

(3) This paper not only examined the fundamental performance of the six aforementioned models but also executed a comparative experiment assessing their performance using Binary Cross-Entropy (BCE) and Focal Loss on the test set, revealing that Focal Loss exhibited superior performance, particularly on datasets with significantly imbalanced labels. The outcomes are depicted in Figure 7.

Due to the significant variance in the number of labels in the text, a comparative test was conducted using batch sizes of 200, 50, and 30. The results indicated that the model's accuracy is consistently high when utilizing smaller batch sizes. The results are listed in Table V.

In this model, FL serves as a balanced loss function to enhance classification accuracy, particularly in scenarios with a long-tail distribution of label data. Enhancing the model's capacity to learn tail label classification, can further improve performance. This study coordinated two parameters, α, and γ, within the FL framework for control purposes. Specifically, α=0.25 and γ=1 were employed to optimize results. Table VI compares evaluation metrics for varying values of γ=2, 1, and 0.5.

TABLE V ACCURACY OF DIFFERENT DEEPLEARNING MODELS

Embed ding	model	Batch=200	Batch=50	Batch=30
Vector	TextCNN_BCE	74.30%	74.96%	83.64%
	TextCNN_Focal	74.89%	75.36%	83.87%
	BiLSTM- BCE	75.49%	74.73%	84.22%
	BiLSTM-Focal	75.79%	79.13%	85.23%
	TextCNN BiLSTM_BCE	78.17%	75.65%	85.03%
	TextCNN BiLSTM_Focal	77.61%	76.71%	86.72%
GloVe	TextCNN_BCE	73.15%	75.46%	85.33%
	TextCNN_Focal	75.37%	75.85%	84.89%
	BiLSTM Attention_BCE	82.61%	83.01%	85.53%
	BiLSTM_Attention_Focal	81.88%	83.97%	85.26%
	BiLSTM-attention TextCNN_BCE	**83.67%**	83.4%	85.96%
	BiLSTM-attention TextCNN_Focal **(Proposed model)**	82.58%	**84.80%**	**87.88%**

TABLE VI MODEL EVALUATION WITH DIFFERENT γ

	Accuracy	F1-micro	F1-weigthed	F1-sample
γ=2	86.88%	83.95%	79.73%	82.84%
γ=1	**87.88%**	**86.66%**	**82.42%**	**86.84%**
γ=0.5	86.58%	84.55%	80.76%	83.35%

TABLE VI MODEL EVALUATION ON MACRO, MICRO, AND WEIGHTED

model	Micro			Macro			weighted		
	P	R	F1	P	R	F1	P	R	F1
TextCNN	89.63%	70.89%	84.14%	51.97%	23.24%	29.61%	85.75%	70.89%	74.69%
BiLSTM	86.94%	76.23%	85.51%	30.24%	22.69%	c	78.77%	76.29%	76.68%
BiLSTM with TextCNN	92.00%	70.03%	87.19%	49.45%	19.25%	21.60%	86.26%	70.03%	74.30%
GloVe TextCNN	89.59%	69.42%	82.94%	46.66%	20.80%	26.33%	84.10%	69.42%	73.27%
GloVe BiLSTM attention	87.73%	81.33%	84.61%	48.02%	36.67%	39.66%	84.55%	81.33%	82.10%
Ours	89.63%	83.89%	86.66%	51.95%	23.24%	32.61%	86.55%	82.23%	82.42%

To evaluate each model more comprehensively, we used the Macro, Micro, and Weighted indicators for evaluation analysis, Table VI is as follows. From the perspective of evaluation indicators, Macro-F1≪Micro F1, the results show a serious imbalance of categories in this data set. Micro-F1 and Weighted-F1 can reflect that the correctness of various classifications is consistent. The precision, recall, and F1 are relatively better in our proposed model. Because they considered the proportion of samples under each category, such indicators are more reasonable. Overall, the accuracy and other evaluation metrics are not particularly high. There are two reasons: first, the data is seriously imbalanced, and it is not easy to learn and classify the samples of some small categories correctly. The second is the challenge of the difficulty of multi-label classification itself. The uncertainty of the number of labels determines the difficulty of this classification task.

V. Conclusion

This paper proposed a multi-label text classification model using a balance loss function. This model used GloVe for word embedding, TextCNN mined local information, and BiLSTM combined with the self-attention mechanism mined global information. These two aspects of information obtain rich text semantic information to accelerate the model convergence speed and improve the model classification performance. Finally, the FL balance loss function is used to complete the training, and the classification accuracy is improved by focusing on tail labels. The model is applied to the Reuter-21578 dataset and compared with other methods. The experimental results show that the proposed model outperforms other models in terms of evaluation indexes, verifying the effectiveness of the model. Future work will consider mining the correlation information between labels to improve the classification performance.

References

[1] Chang, W. C., Yu, H. F., Zhong, K., Yang, Y., & Dhillon, I. S. “Taming pre-trained transformers for extreme multi-label text classification”. In Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining. pp. 3163-3171. August 2020.

[2] Nissa N K, Yuliant E. “Multi-label text classification of Indonesian customer reviews using bidirectional encoder representations from transformers language model”. Int. J. Power Electron. Drive Syst, vol.13, pp. 5641-5652,2023.

[3] Liu, Y., Cheng, H., Klopfer, R., Gormley, M. R., & Schaaf, T. “Effective convolutional attention network for multi-label clinical document classification”. In Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pp. 5941-5953, November 2021.

[4] Liu, X., Shi, T., Zhou, G., Liu, M., Yin, Z., Yin, L., & Zheng, W. “Emotion classification for short texts: an improved multi-label method.” Humanities and Social Sciences Communications, vol.10, no.1, pp.1-9,2023.

[5] Zhu, P., Yuan, Y., Chen, L., Wu, H. “Question answering on agricultural knowledge graph based on multi-label text classification”. In International Conference on Cognitive Systems and Signal Processing, Singapore: Springer Nature Singapore, pp. 195-208, December 2022.

[6] Haojin Hu, Mengfan Liao, Chao Zhang, and Yanmei Jing, “Text classification based recurrent neural network”. In 2020 IEEE 5th Information Technology and Mechatronics Engineering Conference, pp. 652-655, June 2020.

[7] P. Yang, X. Sun, W. Li, S. Ma, W. Wu, and H. Wang, ‘‘SGM: Sequence generation model for multi-label classification”, 2018, arXiv:1806.04822.

[8] You, R., Zhang, Z., Wang, Z., Dai, S., Mamitsuka, H., & Zhu, S. “Attentionxml: Label tree-based attention-aware deep model for high-performance extreme multi-label text classification”. Advances in Neural Information Processing Systems, 32. 2019.

[9] Yoon Kim. Convolutional neural networks for sentence classification. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). pp.1746–1751.2014.

[10] Liu W, Pang J, Li N, et al. Research on multi-label text classification method based on tALBERT-CNN. International Journal of Computational Intelligence Systems, vol.14, no.1, 201, December 2021.

[11] Yang Z, Emmert-Streib F. Threshold-learned CNN for multi-label text classification of electronic health records. IEEE Access 3309157, September 2023.

[12] Yang, Z., & Emmert-Streib, F.“Optimal performance of Binary Relevance CNN in targeted multi-label text classification”. Knowledge-Based Systems, 284, 111286, 2024.

[13] Lu G, Liu Y, Wang J, et al. CNN-BiLSTM-Attention: A multi-label neural classifier for short texts with a small set of labels[J]. Information Processing & Management, vol.60, no.3, May 2023, 103320.

[14] Liang Zhanbo, Jie Guo, Weidong Qiu, Zheng Huang, Shujun Li. “When graph convolution meets double attention: online privacy disclosure detection with multi-label text classification”. Data Mining and Knowledge Discovery, vol.38, no.3, pp.1171-1192, 2024.

[15] Wu Tianxing, Yang Suqun. “Contrastive Enhanced Learning for Multi-Label Text Classification”. Applied Sciences, vol. 14, no. 19,8650, 2024.

[16] Liu Minqin, Liu Lizhao, Cao Junyi, Du Qing. “Co-attention network with label embedding for text classification”. Neurocomputing, 471, pp. 61-69, 2022.

[17] Guo Hao, Li Xiangyang, Zhang Lei, Liu Jia, Chen Wen. “Label-aware text representation for multi-label text classification”, ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, pp.7728-7732, 2021.

[18] Zhao, X., An, Y., Xu, N., & Geng, X. “Variational Continuous Label Distribution Learning for Multi-Label Text Classification”. IEEE Transactions on Knowledge and Data Engineering.2023.

[19] Li, B., Chen, Y., & Zeng, L. “Kenet: Knowledge-Enhanced DOC-Label Attention Network for Multi-Label Text Classification”. In ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) pp. 11961-11965. April 2024.

[20] Huang, Y., Giledereli, B., Kksal, A., Arzucan zgür, & Ozkirimli, E. “Balancing methods for multi-label text classification with long-tailed class distribution”. Association for Computational Linguistics.2021.

[21] Chen, Lin, Zhang, X., Shang, M, "MCFL: multi-label contrastive focal loss for deep imbalanced pedestrian attribute recognition," Neural Computing and Applications. vol.34, no.19, pp. 16701-16715,2022.

Aihua Duan (F’79) was born in Qinhuangdao, China. She received a B.S. degree in computer science and technology from Hebei University of Economics and Business, Shijiazhuang, China, in 2002, and an M.S. degree in computer application from Soochow University, Jiangsu, China, in 2005. She is currently pursuing a Ph.D. degree with the College of Computing and Information Technologies, National University, Manila, Philippines.

From 2005 to 2024, she is a Lecturer at Anhui University of Finance and Economics. Bengbu City, Anhui Province, China. Her research interests include machine learning and natural language processing.

RODOLFO C. RAGA JR. received a Ph.D. degree in computer science from De La Salle University, Manila, Philippines, in 2013. He is a Professor at the College of Computer Studies and Engineering, José Rizal University, Mandaluyong City 1550, Philippines. His research interests include natural language processing, educational data mining, learning analytics, and academic e-learning.