EAI Endorsed Transactions
on Scalable Information Systems
Research Article EAI.EU

Word Embedding for Text Classification: Efficient CNN and Bi-GRU Fusion Multi Attention Mechanism

Yalamanchili Salini[1*], Poluru Eswaraiah[2], M. Veera Brahmam[3] and Uddagiri Sirisha[4]

[1,*] Information Technology, V R Siddhartha Engineering College, A.P ., India, 520007
[2,3] School of Computer Science&Engineering, V I T − AP University, Amaravati, A.P ., India.
[4]ComputerScience&Engineering, P V P Siddhartha Institute of Technology, Andhra Pradesh, India, 520010

Abstract

The proposed methodology for the task of text classification involves the utilization of a deep learning algorithm that integrates the characteristics of a fusion model. The present model is comprised of several attention-based Convolutional Neural Networks (CNNs) and Gate Recurrent Units (GRUs) that are organized in a cyclic neural network. The Efficient CNN and Bi-GRU Fusion Multi Attention Mechanism is a method that integrates convolutional neural networks (CNNs) and bidirectional Gated Recurrent Units (Bi-GRUs) with multi-attention mechanisms in order to enhance the efficacy of word embedding for the purpose of text classification. The proposed design facilitates the extraction of both local and global features of textual feature words and employs an attention mechanism to compute the significance of words in text classification. The fusion model endeavors to enhance the performance of text classification tasks by effectively representing text documents through the combination of CNNs, Bi-GRUs, and multi-attention mechanisms. This approach aims to capture both local and global contextual information, thereby improving the model's ability to process and analyze textual data. Moreover, the amalgamation of diverse models can potentially augment the precision of text categorization. The study involved conducting experiments on various data sets, including the IMDB film review data set and the THUCNews data set. The results of the study demonstrate that the proposed model exhibits superior performance compared to previous models that relied solely on CNN, LSTM, or fusion models that integrated these architectures. This superiority is evident in terms of accuracy, recall rate, and F1 score.

Received on 20 June 2023; accepted on 30 August 2023; published on 26 September 2023

Keywords: Text categorization, Deep learning, Convolution neural network (CNN), Gate recurrent unit (GRU), Attention

doi:10.4108/eetsis.3992

1. Introduction

Kim used Convolutional Neural Networks (CNN) for sentence-level classification in reference [1], where pre-processed word vectors were fed into an English classification model. Similarly, Xing et al. [2] employed CNNs to tackle the Twitter polarity opinion problem. Although CNNs have significantly advanced text categorization, their preference for local characteristics over word context can impact classification accuracy. To overcome this limitation, CNN-based capsule networks and dynamic routing were developed by [3], which outperformed traditional CNNs in classification tasks. [4] suggested Recurrent Neural Networks (RNNs) for text categorization, which consider the text's word structure. However, RNNs are susceptible to gradient dispersion issues. Short-term memory (LSTM) can address this problem but is computationally expensive, memory-intensive, and relies heavily on previous knowledge. Bidirectional Gated Recurrent Units (BiGRU) combine the advantages of Gated Recurrent Units (GRUs) and bidirectional LSTMs, compressing the BiLSTM structure into two gates (update and reset) and solving the problem of semantic

*Corresponding author. Email: yalamanchilisalini@gmail.com

information. BiGRUs consider the meaning of words in context and can converge faster due to their lower number of parameters. When working with lengthy text sequences, the cyclic neural network RNN runs into issues with gradient disappearance and explosion, despite the fact that it has been recommended for its capacity to closely correlate before and after text feature terms. [5] proposed the Long-Term Memory Network (LSTM) in 1997 as a solution to this problem. This network consists of three gates that are designed to improve the processing of long texts and text levels.[6] later introduced a bi-directional LSTM neural network model for text categorization, while [7] created the GRU theoretical model as a more efficient alternative to LSTM. In this work, GRU will collect context semantic information of text feature words and mitigate the adverse effects of CNN's inability to extract context features. The Attention mechanism was first utilized in natural language processing by [8]. Wu proposed a new method for analyzing the emotional content of Chinese text. His method used a self-attention and BiLSTM model that operated on word vectors. Li expanded on Wu's work by proposing a Self-Attention+Bi-LSTM+CNN model that used sentence vectors. Later, Li introduced the BiGRU and Attention mechanism to hierarchically model sentences and documents. This study will use self-attention, drawing on Li's theoretical framework, as a way to enhance attentional focus on salient information.[9]has presented a cyclic convolution neural network that integrates Bi-directional Recurrent Neural Network (Bi-RNN) and Convolutional Neural Network (CNN). Zhang, She, and Li have proposed hybrid models that combine Long Short-Term Memory (LSTM) with CNN, CNN with LSTM, and Bi-directional LSTM with CNN, respectively. Wang proposed the Attention-based bidirectional long-term memory convolution layer (AC-BiLSTM), which comprises the convolutional neural network (CNN), the bidirectional long-term memory (BiLSTM) attention mechanism, and the convolution layer. Luo proposed the use of Latent Dirichlet Allocation (LDA) as a method for text representation. Ma proposed a new text classification model that integrates Convolutional Neural Network (CNN) and Bidirectional Gated Recurrent Unit (BiGRU) with a fusion attention mechanism. The model demonstrated promising results on standard data sets. The fusion network model that has been proposed for text categorization consists of the following components:

- Word embedding technology is used to convert the feature words of the text into a matrix of word vectors.
- The CNN layer is used to extract local features from the word vectors.
- The BiGRU layer is used to capture long-term dependencies between words.
- The attention mechanism is used to focus on the most important words in the text.
- A softmax layer is used to classify the text into one of the target classes.
- To address the existing issues in the domain of sentiment classification of short texts, this study puts forward a text classification model that combines CNN and BiGRU through feature fusion. Additionally, a multi-attention mechanism is incorporated in the proposed model based on prior research.
- The two models that make up the model suggested in this study are CNN and BiGRU. In order to determine the emotional polarity of the target keyword in the phrase, the CNN model incorporates the particular target sentiment classification approach [10]. After BiGRU evaluates the emotional polarity at the sentence level, the two features are combined to create a fusion global feature vector.

2. PROPOSED METHOD

This section presents a comprehensive overview of the proposed architecture, which comprises several layers, including the input, embedding, recurrent, output, and classification layers (softmax) [11]. Embedded words are used as inputs in our method, and a GRU layer is used to extract the lexical characteristics. A classifier layer that completes the final classification follows as the architecture's concluding component as shown in Figure 1.

2.1. Embedding layer

Word embedding is a method of representing words as distributed vectors, which can be trained using deep learning techniques on large data sets to capture their contextual meaning. One-hot representations, on the other hand, suffer from the curse of dimensionality and are not suitable for deep neural networks [12]. Previous research has shown that neural networks can achieve better results with unsupervised pre-training procedures. In the architecture that we have proposed, the sizes of both the GRU hidden layer and the word embedding have been predetermined. We used 200-dimensional Glove word vectors that had been pre-trained by [13], which are considered to be state-of-the-art for a variety of NLP applications. This allowed us to initialize all of the word embeddings. The Glove model makes an attempt to categorize the target word by providing a neural network with inputs consisting

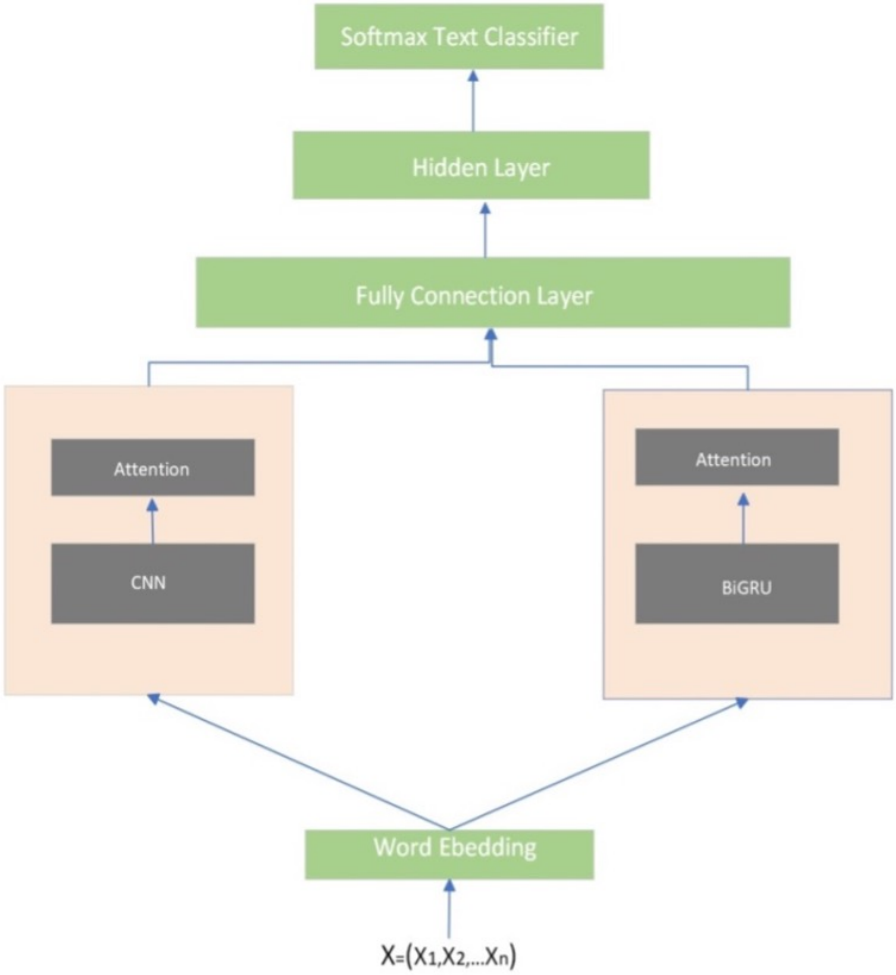

Figure 1. The proposed model architecture

of background phrases. Other pre-trained embeddings, such as those from the competing set called GloVe, have also shown good performance. In our architecture, we use an embedding layer to transform the input words into lower-dimensional vectors and refine them through backpropagation. This embedding layer is essentially the first layer of our feed-forward RNN, where word embedding serves as the weights. Specifically, the initial relationship between two words' probabilities of co-occurrence is based on the information contained in the encoding of the difference vector represented in Figure 2.

$$\left(w_i \middle| w_j\right) = \frac{p(w_{ij})}{p(w_t)} \qquad (1)$$
$$(i = j-1, j-2, -n),\ j+1, j+2, \ldots, j+n$$

In the given sentence sequence $w_1^d, w_2^d, \ldots, w_n^d, n$ where n represents the length of each text sequence and d represents the dimension of the word vector, the input size of this layer is l x b, which refers to the text matrix. The output size is b x l x d, representing the word vector matrix. Here, b represents the number of text batches, l represents the fixed length of the text, and d represents the dimension of the word vector.

2.2. CNN-Attention

Initially, the word vector of the text is fed into the convolution layer to obtain the output features from said layer. The activation function processes the convolution result after the convolution kernel has been applied to the incoming text vector matrix by getting the dot product. This process is employed in this stratum.

$$O_i = w.A\,[i : i+j-1]\, i = 1, 2, \ldots, s-j+1 \qquad (2)$$

The equation represents the convolution operation between a kernel matrix and a matrix of text word vectors with dimensions of width and height. Here, rA

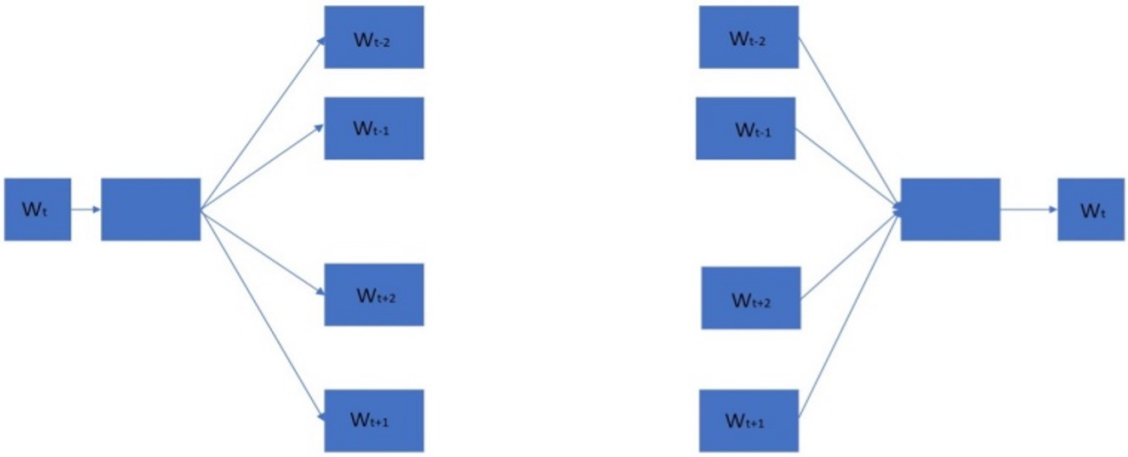

Figure 2. Skip-gram and CBOW structure

[i: j] denotes the unique word vectors for lines i to j of the text.

$$C_i = f(O_i + b) \tag{3}$$

After obtaining the output features from the convolutional layer, they are then fed through a pooling layer to undergo further processing. This particular layer serves to condense the output by extracting the most essential features from the convolutional output as represented in Figure 3.

$$Y_i = \max_p ool(C_i) \tag{4}$$

The following describes the precise Attention execution process:

$$X_i = \tanh(W_s h_t + b_s) \tag{5}$$

$$\propto_i = \frac{e^{(w_u . X_i^T)}}{\sum_{i=1}^{n} e^{(w_u . X_i^T)}} \tag{6}$$

$$u_i = \sum_{i=1}^{n} \propto_i \; _i \tag{7}$$

hi is shorthand for the word vector that is generated by the bi-directional GRU, and Xi is shorthand for the word vector that is generated by the activation function tanh. The average of the word vectors is then used to produce the word vector encoding ui re, where Wu represents the attention weight of the word vector in the sequence vector $s = h_1, h_2, \ldots, h_n$. In order to accomplish this, the arbitrary beginning vector Wu is used.

2.3. GRU-Attention

The GRU-Attention layer takes pre-trained word vectors as input and utilizes the attention mechanism to compute feature vectors with associated weight values using the GRU [14]. A section of the GRU-Attention structure is illustrated in the accompanying Figure 4. Following is a depiction of two-way GRU:

$$\vec{h_t} = G\vec{R}U(w_t) \tag{8}$$

$$ht = GRU(wt) \tag{9}$$

$$h_t[\vec{h_t}, ht] \tag{10}$$

By following the three stages described in steps (8), (9), and (10), the GRU network can generate a global feature that includes the feature word vector along with its weight score, as shown in equation (10).

2.4. Fully connection layer

The input vector for the fully connected layer can be created by combining the GRU vector h with the output vector v obtained from the CNN-attention calculation. The resulting vector F is then computed through attention computation [15]. This serves as the final output of the model and can be determined using the following formula:

$$F = f\left[W_c . (v \oplus h) + b_c\right] \tag{11}$$

The result of the hidden layer can be written as follows, where Wc and bc stand for the complete connection layer's bias vector and weight matrix, respectively:

$$y = f(W_H . F + b_H) \tag{12}$$

HW stands for the complete connection layer's weight matrix, and bH stands for the full connection layer's matrix vector. The Softmax layer receives its input

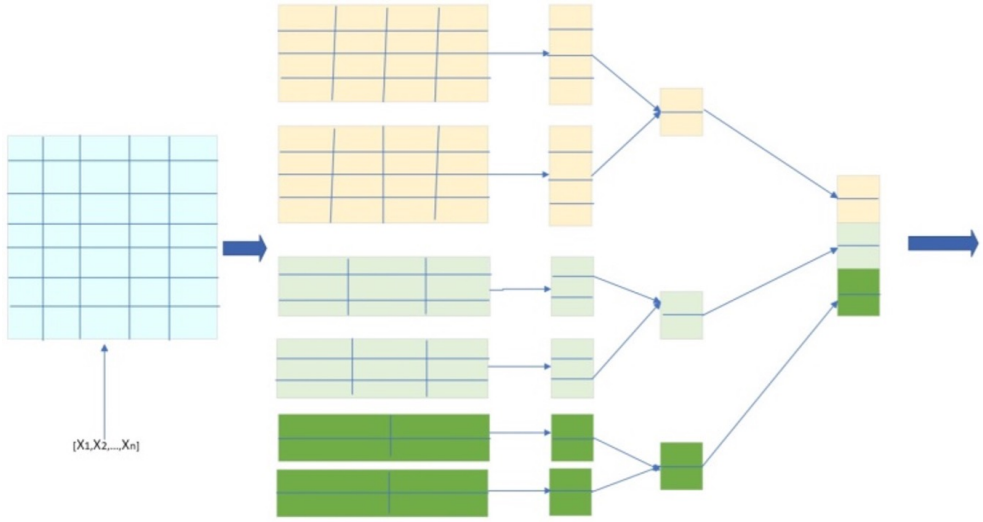

Figure 3. Text –CNN Attention Model

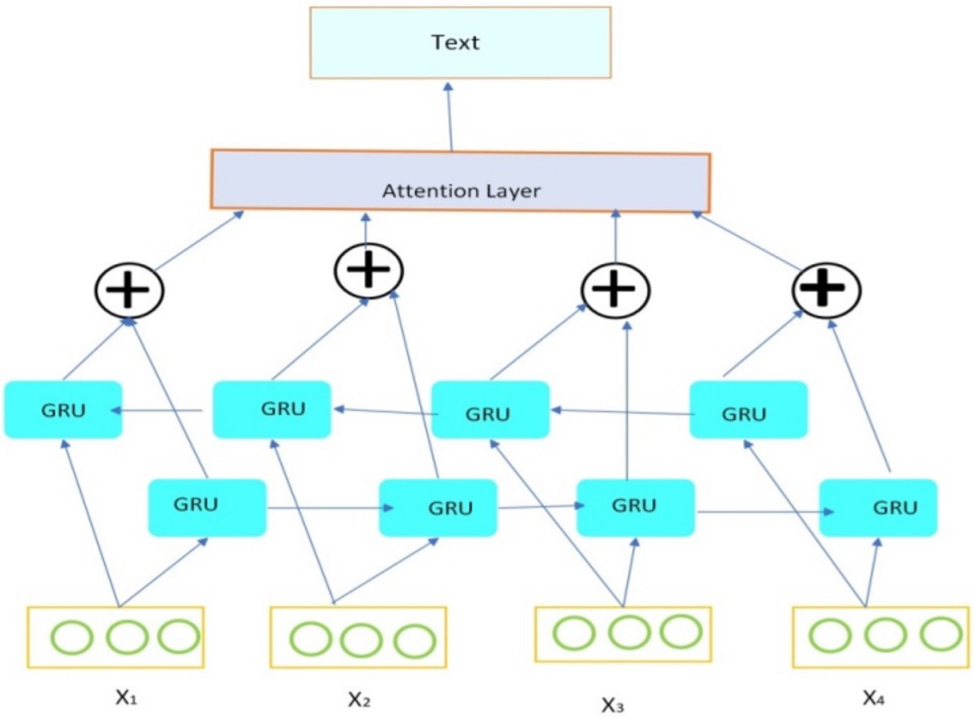

Figure 4. The structure of Bi-GRU attention

from the concealed layer's output. The Softmax layer determines the likelihood of categorizing the text.

$$\hat{y} = softmax(y_i) = \frac{e^{y_i}}{\sum_{i=1}^{n} e^{y_i}} \tag{13}$$

During the training process, the objective is to minimize the loss function when the input label is y and the SoftMax prediction outcome is also y. By denoting the input label as y and the SoftMax prediction outcome as $s = h_1, h_2, \ldots, h_n$, the aim of the training procedure is to achieve the lowest possible value for the loss function.

$$loss = -argmin \sum_{i=1}^{n} p(y_i) \log(p(\widehat{y_i})) \tag{14}$$

The model's parameters are continually updated by computing the loss function, aiming to achieve optimal training performance.

3. RESULTS AND DISCUSSION

3.1. Data set

THUCNews Data set: The THUCNews data set is a Chinese text classification data set consisting of news articles collected from the Chinese internet. It covers a wide range of categories such as politics, finance, sports, entertainment, and technology. Each article is labeled with a specific category, allowing for supervised learning tasks. The data set is significant in size, containing tens of thousands to hundreds of thousands of articles, making it suitable for training and evaluating models for Chinese text analysis tasks.
IMDB Data set: The IMDB data set is a popular collection of English text used for sentiment analysis. It comprises movie reviews sourced from the Internet Movie Database (IMDB). Each review is assigned a binary sentiment label, indicating whether it is positive or negative. This data set is well-balanced, containing an equal number of positive and negative reviews. It serves as a valuable resource for training and evaluating models in sentiment analysis, as it offers a wide range of opinions and sentiments expressed in movie reviews. Refer to Table 1 for an overview of the parameter values.

3.2. Experimental Hyper parameters

Experimental hyperparameters refer to the settings or values chosen for various parameters during the experimentation phase of building machine learning models. These hyperparameters are not learned from the data but are set manually by the practitioner or researcher to optimize the model's performance as represented in Table 2.

3.3. Model evaluatrion standard

Evaluation metrics or criteria, commonly referred to as model evaluation standards, are utilized to evaluate the efficacy and performance of machine learning models. The aforementioned standards offer numerical criteria that enable professionals and scholars to contrast diverse models, opt for the most suitable model for a specific undertaking, and arrive at well-informed determinations regarding model implementation. The following are frequently utilized benchmarks for evaluating models. Table 3 presents the confusion matrix from [16].

1. **Accuracy:** The metric of accuracy evaluates the degree of correctness in the predictions made by a model. It is determined by dividing the number of instances that were classified correctly by the total number of instances. The metric in question is a simple evaluation measure, however, its applicability may be limited in cases where the data set exhibits class imbalance, i.e., when the distribution of classes is uneven.
2. **Precision and Recall:** The assessment of binary classification tasks often involves the utilization of precision and recall as standard evaluation metrics. Precision and recall are two important metrics used in evaluating the performance of a classification model. Precision is defined as the ratio of true positive predictions to the total number of instances predicted as positive. On the other hand, recall is defined as the ratio of true positive predictions to the total number of actual positive instances. The utilization of these metrics is advantageous in cases where there exists a disparity between the positive and negative categories within the data set.
3. **F1-Score:** The F1-score is a metric that quantifies a model's performance by taking into account both precision and recall and is calculated as the harmonic mean of these two measures. It is considered a balanced measure of a model's effectiveness. This approach is frequently employed in scenarios where there is a need for both precision and recall to be accorded equal significance.

$$pre = \frac{TP}{TP + FP} rec = \frac{TP}{TP + FN} F_1 = \frac{2 \times pre \times rec}{pre + rec} \quad (15)$$

3.4. Model comparison and analysis

The process of evaluating and contrasting the efficacy of distinct machine learning models for a specific task is referred to as model comparison and analysis. The present study has opted to utilize a recently proposed model for the purpose of comparing it with the model presented in this paper. The objective is to establish the superiority of the model proposed in this paper. This paper selects three models for comparative analysis, namely CNN-BiGRU, CNN-LSTM-Attention (CLA), LSTM-CNN-Attention (LCN), and CNN+LSTM as shown in Figure 5 and 6.

According to the aforementioned experimental results, the suggested model has varied degrees of improvement in terms of accuracy, recall, and F1 value when compared to CLA, LCA, CNN-BiGRU and LSTM + CNN models[23].

4. CONCLUISION

The purpose of this research is to improve classification performance by developing a text classification model

Table 1. Dataset Information

Dataset	Classification	Testing sample	Training sample
THUCNews Dataset	14	20000	20000
IMDB	2	20000	20000

Table 2. Model parameters

Parameter	Word2vec	CNN Network	LSTM Network
Training algorithm	Word2vec	CNN	LSTM
Window size	5	-	-
Window number	5000	-	-
Word vector dimension	200	200	200
Learning rate	-	0.001	0.001
Convolution kernel height	-	2,3,4	-
Convolution kernel width	-	200	-
Number convolution per size	-	100	-
Dropout	-	0.5	0.5
Activation function	-	ReLu	ReLu
Hidden layer size	-	-	256

Table 3. Confusion matrix

Actual	Negative	Positive
Predicted	FN	TP
False	TN	FP

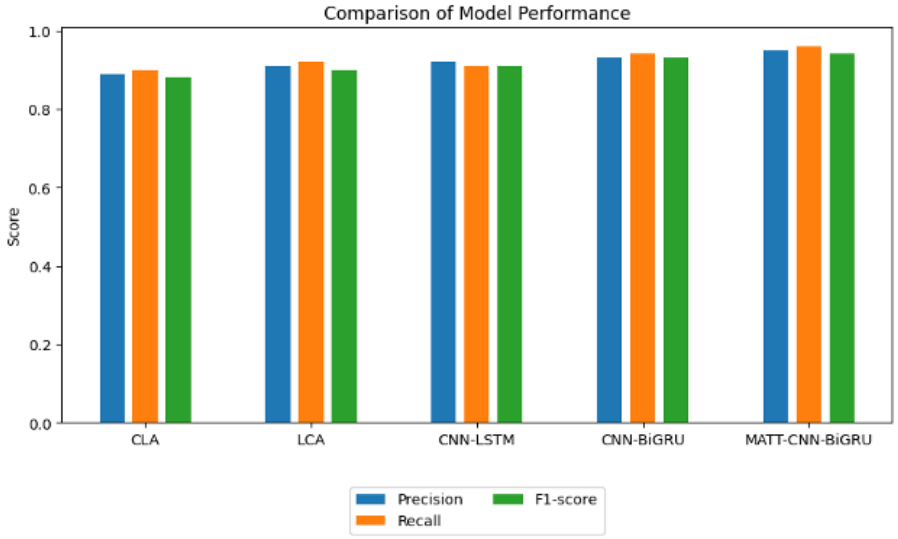

Figure 5. THUC News Dataset

that is based on the CNN+LSTM technique [17],[18], [19]. In order to reduce the amount of time required for training while preserving the model's capacity for accurate classification, the model makes use of a GRU structure rather than LSTM units. In addition, the Attention mechanism was included in order to improve the accuracy of the model's classification. This was accomplished by centering attention on the words that had the greatest influence on the categorization effect.

References

[1] Kim, Y. (2014) Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882* .

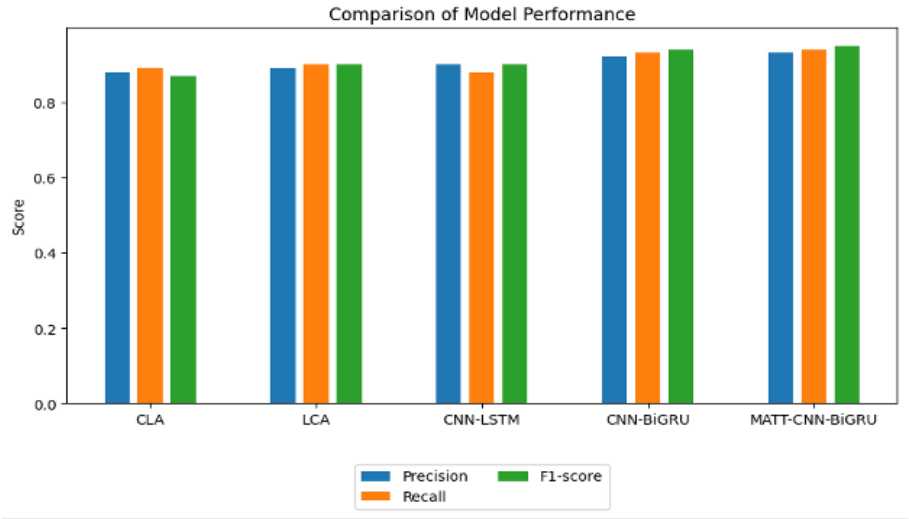

Figure 6. IMDB Experimental Results Comparison

[2] Xing, S., Wang, Q., Zhao, X., Li, T. *et al.* (2019) A hierarchical attention model for rating prediction by leveraging user and product reviews. *Neurocomputing* **332**: 417–427.

[3] Zhao, W., Ye, J., Yang, M., Lei, Z., Zhang, S. and Zhao, Z. (2018) Investigating capsule networks with dynamic routing for text classification. *arXiv preprint arXiv:1804.00538* .

[4] Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S. and Dean, J. (2013) Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems* **26**.

[5] Schmidhuber, J. (2015) Deep learning in neural networks: An overview. *Neural networks* **61**: 85–117.

[6] Graves, A. and Schmidhuber, J. (2005) Framewise phoneme classification with bidirectional lstm and other neural network architectures. *Neural networks* **18**(5-6): 602–610.

[7] Cho, K., Van Merriënboer, B., Bahdanau, D. and Bengio, Y. (2014) On the properties of neural machine translation: Encoder-decoder approaches. *arXiv preprint arXiv:1409.1259* .

[8] Wu, X., Chen, L., Wei, T. and Fan, T. (2019) Sentiment analysis of chinese short text based on self-attention and bi-lstm. *Journal of Chinese Information Processing* **33**(6): 100–107.

[9] Lihua, L. and Xiaolong, H. (2020) Text sentiment analysis based on deep learning [j]. *Journal of Hubei university (natural science edition)* **42**(02): 142–149.

[10] Cai, J., Li, J., Li, W. and Wang, J. (2018) Deeplearning model used in text classification. In *2018 15th international computer conference on wavelet active media technology and information processing (ICCWAMTIP)* (IEEE): 123–126.

[11] Lai, S., Xu, L., Liu, K. and Zhao, J. (2015) Recurrent convolutional neural networks for text classification. In *Proceedings of the AAAI conference on artificial intelligence*, **29**.

[12] Zhang, J., Li, Y., Tian, J. and Li, T. (2018) Lstm-cnn hybrid model for text classification. In *2018 IEEE 3rd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)* (IEEE): 1675–1680.

[13] She, X. and Zhang, D. (2018) Text classification based on hybrid cnn-lstm hybrid model. In *2018 11th International symposium on computational intelligence and design (ISCID)* (IEEE), **2**: 185–189.

[14] Wang, G., Li, C., Wang, W., Zhang, Y., Shen, D., Zhang, X., Henao, R. *et al.* (2018) Joint embedding of words and labels for text classification. *arXiv preprint arXiv:1805.04174* .

[15] Luo, L.x. (2019) Network text sentiment analysis method combining lda text representation and gru-cnn. *Personal and Ubiquitous Computing* **23**(3-4): 405–412.

[16] Salini, Y. and Harikiran, J. (2023) Multiplicative vector fusion model for detecting deepfake news in social media. *Applied Sciences* **13**(7): 4207.

[17] Ma, Y., Chen, H., Wang, Q. and Zheng, X. (2022) Text classification model based on cnn and bigru fusion attention mechanism. In *ITM Web of Conferences* (EDP Sciences), **47**: 02040.

[18] Johnson, R. and Zhang, T. (2014) Effective use of word order for text categorization with convolutional neural networks. *arXiv preprint arXiv:1412.1058* .

[19] Al-kanan, H., Yang, X. and Li, F. (2020) Improved estimation for saleh model and predistortion of power amplifiers using 1-db compression point. *The Journal of Engineering* **2020**(1): 13–18.