Enhancing Multi-Class Text Classification with BERT-Based Models

Haojia Wu
School of Computer Science
The University of Auckland
Auckland, New Zealand
jwen758@aucklanduni.ac.nz

Xinfeng Ye
School of Computer Science
The University of Auckland
Auckland, New Zealand
x.ye@auckland.ac.nz

Sathiamoorthy Manoharan
School of Computer Science
The University of Auckland
Auckland, New Zealand
s.manoharan @auckland.ac.nz

2023 IEEE Asia-Pacific Conference on Computer Science and Data Engineering (CSDE) | 979-8-3503-4107-2/23/$31.00 ©2023 IEEE | DOI: 10.1109/CSDE59766.2023.10487760

***Abstract*— Document classification serves a diverse range of practical applications, significantly enhancing various processes. For instance, it can expedite the identification of categories assigned to research reports submitted by students. By subsequently aligning these categories with the particular interests of university staff members, the reports can then be evaluated by those who possess the most relevant expertise. In this paper, we developed and evaluated several models for carrying out multi-label and multi-class text classification. Our approach revolves around the pre-trained BERT models. We endeavour to augment the efficacy of classification by leveraging latent information drawn from the output and hidden layers of the BERT architecture. We seek to achieve good accuracy and F1 score of the classification process. Our model utilizing the attention mechanism and LSTM to process information generated by BERT outperforms all other models based on our evaluation.**

Keywords—text classification, machine learning

I. Introduction

To uphold the standard of academic projects, a widely adopted approach involves subjecting these projects to evaluation by external examiners outside the academic institution. To facilitate this evaluation process, the students' advisors need to identify suitable external examiners. This is accomplished through an alignment of the research interests of external academics with the thematic focus of the students' projects. However, this introduces a logistical challenge: the process of identifying external examiners can often be time-intensive.

Machine learning has been applied in text classification, a task involving the categorization of textual content into distinct classes based on underlying semantic nuances. In recent years, BERT [1] has risen to become the go-to tool in a multitude of Natural Language Processing (NLP) tasks [2] [3]. BERT's good performance is achieved through its pre-training on vast repositories of textual data. This pre-training equips BERT with the capability to generate embeddings that encapsulate the semantic essence of words. What sets BERT's embeddings apart is their profound contextual understanding, enabling them to accurately capture the intricate interplay of words within their respective contexts. This contextual depth endows the word embeddings with a heightened fidelity to the intended meanings, a virtue that significantly bolsters downstream tasks, such as text classification, which rely on these embeddings to perform.

In our previous work, we developed a framework that helps academics to match their students' reports with potential examiners [4]. One important component in the framework is to classify the categories of the students' reports to allow them to be matched with the research interests of potential examiners. Accurately classifying the categories of the students' reports is a prerequisite for the system to work effectively.

The goal of our current research is to create a machine-learning model adept at discerning the underlying themes and subjects discussed within students' reports. As a result, our model will help academic staff in identifying suitable external examiners for these reports, streamlining the process of reviewer selection. Our task is akin to multi-label text classification, as individual reports often span multiple subject domains. For example, a single report might delve into both computer hardware intricacies and physics principles. Such a paper could logically be labelled with tags like "COMPUTER SCIENCE" and "PHYSICS". Given BERT's well-established reputation as a go-to solution for NLP tasks, we have used BERT to generate the word embedding representing the meaning of the words in reports. This allows us to harness the robust capabilities of pre-trained models like BERT, effectively capitalizing on their intricate understanding of context and semantics to elevate the accuracy and efficacy of our classification efforts.

In this paper, we study and assess multiple machine learning models designed to carry out the task of multi-label text classification. Our study focuses on the utilization of pre-trained BERT models. The models that we developed revolve around harnessing the latent knowledge encapsulated within both the output and concealed layers of the intricate BERT architecture.

The remainder of the paper is structured as follows: Section II introduces background knowledge and related works. Section III delves into the various models we developed for text classification. Section IV outlines the experimental results. The final section presents the conclusions.

II. Background and Related Work

A. The BERT Model

In NLP, the meaning of a word is represented using an n-dimensional vector called *word embedding*. Many models have been developed to generate word representations that take into account of the context in which a word is used, e.g., BERT. BERT consists of several layers of encoding blocks as shown in Fig. 1.

Each encoding block consists of a multi-head self-attention mechanism layer and a feed-forward network layer. When a text with *n* words is given to BERT as the input, first

978-1-6654-5305-9/22/$31.00 ©2023 IEEE

each word in the text is treated as a token, and two special symbols CLS and SEP are added to the start and the end of the text. In BERT, the input text is represented as a sequence of tokens. Each token is a word/sub word/symbol in the text. Each token is represented as a pre-defined vector.

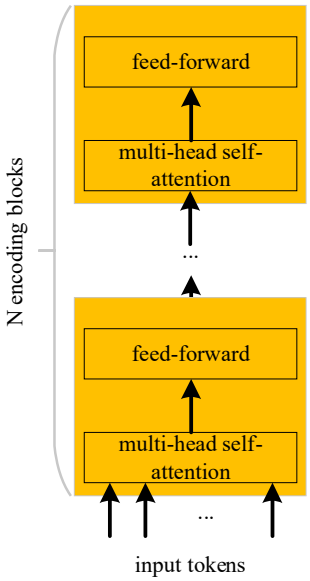

Fig. 1. The BERT Model

The dot-product of the two vectors representing two tokens is used to judge the correlation between the two corresponding tokens. The attention mechanism first projects each token to a n-dimensional space. Then, the relationship between the tokens is computed based on the results of the dot-product operation. Multi-head attention means the attention mechanism is applied several times to each pair of tokens as each token is projected to several n-dimensional space and the correlations between the tokens are computed for each of the n-dimensional spaces. It can be regarded that each n-dimensional space corresponds to a specific contextual feature for the input text. Thus, the multi-head attention mechanism intends to discover the relationship between the tokens under different context.

The results of the multi-head attention are concatenated and processed by a feed-forward neural network. The feed-forward network processes the vector that corresponds to each input token. It can be regarded as adjusting the vector representation of a token according to its relationships with the other tokens.

The vectors representing the tokens in the sentence are send through each of the encoding blocks. Each encoder has an output for each input token. For example, if there are n input tokens, there are n outputs in the encoder where $output_i$ corresponds to $input_i$ where $1 \leq i \leq n$. Each output of an encoding block is a vector. $output_i$ can be regarded as the information that have been encoded for the corresponding input token given to $input_i$. When an encoding block encodes information for an input token, it takes into account of the context in which the token appears in the text. Thus, $output_i$ can be regarded as representing the meaning of the token at $input_i$ in the text given to encoding block. The CLS token in the output of each encoding block can be regarded as the summary information collected by the encoding block based on the input to the encoder. Thus, the value of the CLS token in the output of an encoding block can be regarded as representing the semantics of the text observed by the encoding block at that specific layer.

B. Related Works

Attention-based techniques are used in numerous Natural Language Processing (NLP) tasks. In the work of Pal et al. [1], a feature matrix along with a correlation matrix is employed to capture interdependencies among labels. Subsequently, this information is harnessed to create a classifier. The authors obtained sentence feature vectors through the utilization of BiLSTM and attention mechanisms. These derived vectors are then fed into the classifier to yield the desired outcomes.

Integrating the attention mechanism with Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) architectures enhances the extraction of contextual relationships among words. Liu and Guo [5] introduce a model designed to address the intricate semantics of natural language. This model comprises a convolutional layer for extracting higher-level representations, Bidirectional LSTM to encompass both preceding and succeeding contexts, and an attention mechanism to intensify focus on critical information.

Jang et al. [6] contend that a mere combination of CNN and LSTM might not be the optimal solution due to the substantial number of features requiring consideration for classification. Hence, they integrate the attention mechanism with Bi-LSTM and CNN, effectively harnessing the benefits of both approaches.

Quantifying connections between word pairs in a sentence poses a challenge for RNN models. Handling lengthy input text can also be problematic due to the vanishing gradient issue. To address this, Jing [7] proposes a model that combines RNN with self-attention, mitigating the vanishing gradient problem and gaining better efficacy by extracting more structural potential.

The extraction of meaningful features from intricate semantic information stands as a primary hurdle in text classification. Yu et al. [8, 4] introduce a parallel RNN architecture consisting of LSTM and GRU components that operate in tandem. They generate an attention matrix by computing the similarity between the simultaneously captured context information.

In the pursuit of enhancing RNN and CNN-based models for multi-label text classification, diverse techniques and extensions have been explored. Zheng et al. [9] capture locally pertinent information within the text by amalgamating CNN with BERT. They subsequently feed the original BERT output, coupled with local representation, into a Transformer encoder to yield the ultimate outcome. To address the vocabulary limitations and feature sparsity intrinsic to short text classification, Bao et al. [10] employ BERT to train word vector representations, CNN to capture static features, and BiGRU to encapsulate contextual features.

Given the limitations of distributed word embeddings like Word2Vec and GloVe in handling polysemy, Fan et al. [11] leverage BERT to attain contextualized word embeddings. They subsequently employ a 1-D convolution and max pooling to extract high-level features. The time-intensive nature of training models with numerous parameters prompts Zhou [12] to integrate TF-IDF to eliminate insignificant features, which are then input into the CNN-LSTM model.

The outcomes reveal a minor accuracy reduction in shorter texts.

Xu et al. [13] adopt BERT as an information extractor, using multi-head attention to further distil text information into low-dimensional feature vectors enriched with dense semantic nuances. By concatenating these vectors, they amplify the information entropy of text vectors. To tackle overfitting in small datasets, they propose a loss function incorporating the Flooding mechanism.

The embedding approach of BERT also harmonizes seamlessly with graph convolutional neural networks. Tran et al. [14], the BERT embeddings translate all text content into numerical vectors, which are subsequently fed into a graph convolutional neural network to procure outcomes.

Cai et al. [15] propose a model that utilizes BERT to compute context-aware document representations. They introduce a label graph to capture label structures and correlations. The attention mechanism forges semantic links between labels and words, culminating in a hybrid representation that encapsulates both context and label information.

Song et al. [16] use LSTM and an attention mechanism to aggregate the intermediate embedding of the [CLS] token, ultimately yielding the final output. Lehečka et al. [17] investigates the BERT model enhanced with a pooling layer to utilise the information generated by the BERT model in multi-label text classification. This model is employed for classifying Wikipedia datasets in three distinct languages, with a classification scope spanning 5,200 labels.

BERT also finds utility in sequence-to-sequence multi-label text classification. Yarullin et al. [18] demonstrates that the Sequence Generating BERT model attains comparable performance with a reduced number of training epochs when contrasted with the standard BERT model. The potential synergies between these two models can be further harnessed by their combination, enabling a more profound exploration of latent semantic information.

Transformer-based techniques similarly harness the potency of the attention mechanism. By deploying self-attention mechanisms, transformers adeptly encapsulate overarching dependencies in input text, enabling parallelized processing and adept representation of extended contextual associations. Certain models have sought to extract the complete potential of the Transformer architecture alongside label hierarchies [19]. However, most existing models have been predominantly text-focused, often disregarding latent metadata information and hierarchical labels. To address this gap, the study outlined in [20] reshapes the problem formulation into metadata-aware text classification within an expansive label hierarchy. The researchers introduce the model MATCH, which seamlessly integrates metadata by pre-training text and metadata embeddings in the same space. The attention mechanism is further employed to apprehend interconnections between the two.

A concept stemming from metadata-induced contrastive learning appears as an approach for multi-label text classification [21]. This study predominantly revolves around Large Margin Text Classification (LMTC) in a zero-shot context, effectively reducing the need for extensive human-annotated training data. The researchers introduce an innovative metadata-induced contrastive learning (MICoL) method, which leverages document metadata to streamline the identification of relevant document pairs.

Some researchers have directed their efforts towards fully harnessing the semantic significance of labels [22]. They go beyond utilizing solely word embeddings and incorporate label embeddings as input for the training process. This approach showcases how amalgamating label information can potentially enhance text learning in models. Another more direct integration of label embeddings into BERT is proposed [23]. This involves concatenating the textual content of labels with the original document, separated by a [SEP] token. Separate segment embeddings are assigned to label text and document text. This model achieves good performance while maintaining nearly identical computational costs.

Capturing semantic nuances across different text levels—such as words, sentences, and documents—is important in multi-label text classification. The application of the LDA topic model aids in unearthing latent semantics within documents. As outlined in [24], the LDA topic model is employed to capture the topic vector associated with each word.

Some researchers noticed that many models ignore the interaction between sentences in a text and the attention learned from text-label in the joint space. Dong et al. [25] introduce a method for text classification. This includes the incorporation of a Self-Interaction attention mechanism and label embedding. The attention mechanism facilitates the extraction of more meaningful text representations, while the joint embedding of labels and words leverages the attention acquired from text-label connections.

Certain research focuses intently on the interplay between labels [26]. To better encapsulate label correlations, they present a label-aware network. This approach employs a heterogeneous graph to facilitate label representation learning, drawing from the observation that closely associated labels or words tend to exhibit similarity. The incorporation of bidirectional attention flow components enhances the extraction of contributions from all text segments in both directions, leading to performance improvement [27].

III. THE PROPOSED SYSTEM

We build several machine learning models based on the pre-trained BERT model. This section outlines these models.

A. Basic BERT Model

The structure of the model is depicted in Fig. 2. In this model, we utilize the pre-trained bert-base-uncased version of BERT. The model comprises 12 layers of encoding blocks, with each multi-head self-attention having 12 heads. The embedding dimension is 768. The final embedding of the [CLS] token, generated by the last encoder layer, is fed into a classifier to predict the labels of the input text.

The classifier is a fully-connected linear layer with 768 inputs. The number of outputs is equal to the number of categories. Each output corresponds to a category. A Sigmoid function is applied to each output of the fully connected layer. If the Sigmoid function's value exceeds 0.5, the document is deemed to belong to the respective category. The structure of the classifier remains consistent across all models; they solely differ in the number of inputs. This count is determined by how each model utilizes the various information generated by the BERT model.

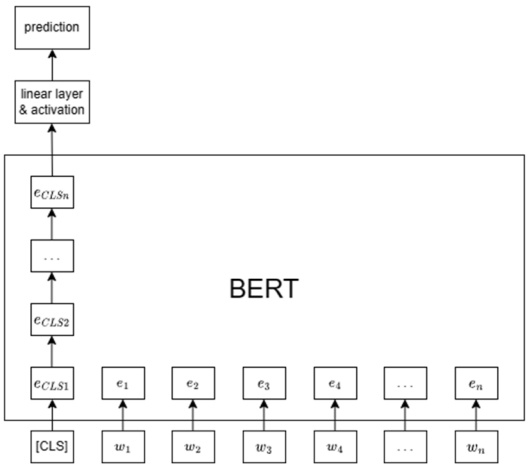

Fig. 2. Basic BERT Model

B. BERT-LSTM-Attention Model

An inherent limitation of the Basic BERT model is its reliance solely on the [CLS] token from the final encoding block for classification purposes. This approach results in the omission of potentially valuable information embedded within the individual word embeddings.

LSTM possesses the capability to capture inter-word relationships within a sentence, processing words sequentially. Nonetheless, LSTM's drawback lies in its potential to lose track of connections between widely separated words. In contrast, the attention mechanism excels at identifying relationships between any two words within a sentence, regardless of their distance. In the BERT-LSTM-Attention model, following the initial processing of input text by the BERT model, the BERT model's outputs undergo sequential processing through an LSTM layer. The outputs at each step of the LSTM model subsequently traverse a multi-head attention layer to adjust token representation, accounting for inter-token relationships. This multi-head attention layer functions similarly to its counterpart within the BERT model. The outputs originating from the multi-head attention mechanism and the [CLS] token extracted from the last layer of BERT's encoder are combined and channeled through a classifier, as illustrated in Fig. 3.

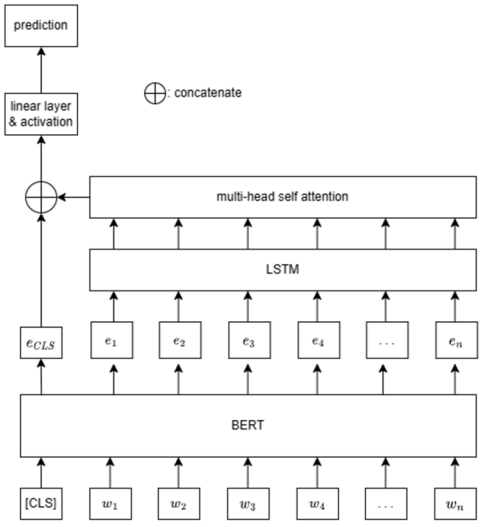

Fig. 3. BERT-LSTM-Attention Model

C. BERT-CLS-Pooling Model

Within the Basic BERT model, the [CLS] token in the last encoding block serves as the classifier's input. This approach can potentially result in information loss due to the underutilization of features gleaned from the lower layers of the encoding blocks. To make full use of the insights garnered by the lower layer encoding blocks, we incorporate an LSTM to amalgamate the lower-layer information.

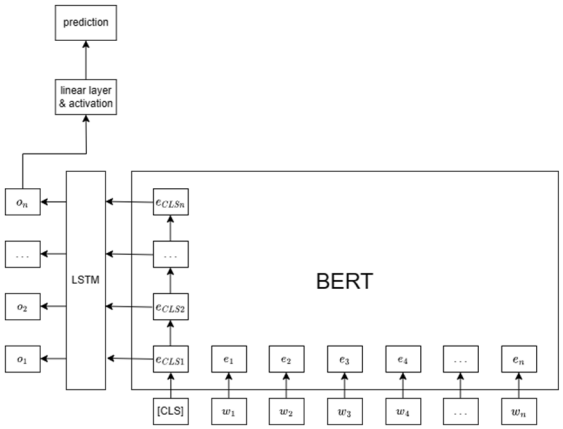

Fig. 4. BERT-CLS-Pooling Model

Let's define the ensemble of [CLS] tokens generated by the BERT encoders as $e_{CLS} = \{e_{CLS}^1, e_{CLS}^2, \dots, e_{CLS}^{12}\}$, encompassing 12 tokens—matching the 12 encoding blocks within BERT. These tokens undergo sequential processing through an LSTM, initiated from the token produced by the first encoding block. The LSTM's output is subsequently fed into the classifier, depicted in Fig. 4. This strategy allows us to extract and leverage insights from the various encoder layers to enhance classification performance.

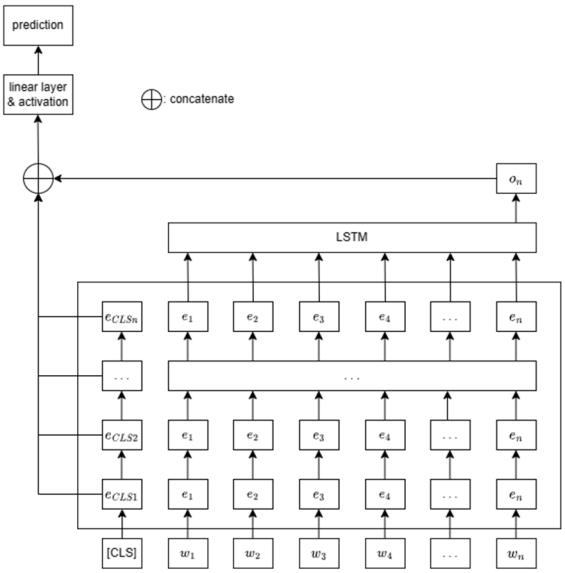

Fig. 5. BERT-LSTM-All Model

D. BERT-LSTM-All Model

The BERT-LSTM-Attention model disregards the features extracted from the text by the lower-layer encoding blocks, while the BERT-CLS-Pooling Model focuses solely on the aggregated information generated by the encoding blocks. In the BERT-LSTM-All model, we encompass all the summary information derived from the encoding blocks, in

addition to the embeddings produced for each word within the text by the BERT model. To aggregate the word embeddings, an LSTM is employed. The resultant pooled information is combined with the summary details originating from all the encoding blocks, specifically the [CLS] tokens generated by each encoding block. This concatenated information is then provided to the classifier. The structural configuration of the model is illustrated in Fig. 5.

E. BERT-All Model

Each encoder within the BERT model houses a multi-head self-attention layer that uncovers relationships among the tokens representing the text. Thus, in the BERT-All model, we consolidate all the outputs from the BERT model with the [CLS] tokens generated across all encoding blocks. Subsequently, this amalgamated information is channeled into the classifier. The distinction between this model and the BERT-LSTM-All model lies in the employment of an LSTM layer in the latter, compressing the BERT model's outputs into a singular vector. Conversely, the outputs of the BERT model within the BERT-All model are directly fed into the classifier. The architectural blueprint of the BERT-All model is depicted in Fig. 6.

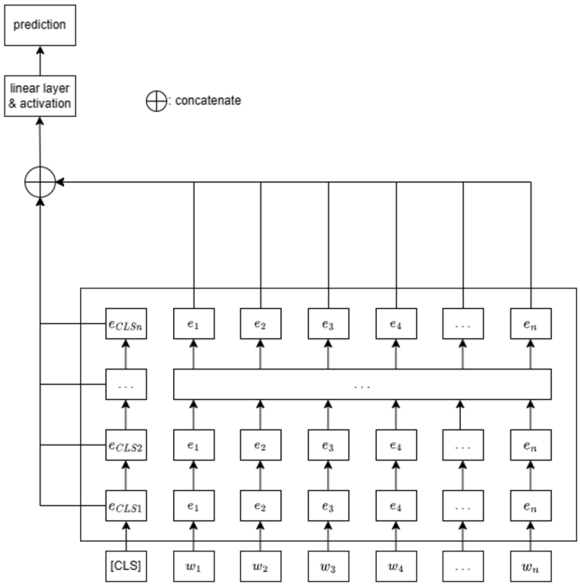

Fig. 6. BERT-All Model

IV. EXPERIMENTS

A. Datasets and Evaluation Metrics

1) Analytics Vidhya Hackthon Dataset. The dataset comprises roughly 12,000 papers, each accompanied by its corresponding title and abstract. Every record is associated with at least one label, encompassing categories such as Computer Science, Physics, Mathematics, Statistics, Quantitative Biology, and Quantitative Finance.

2) arXiv Paper Collection. We gathered abstracts of 26,000 papers sourced from arXiv.org, covering a wide spectrum of research. These papers span 10 distinct categories within Computer Science: Computer Vision, Artificial Intelligence, Computation and Language, Machine Learning, Systems and Control, Neural and Evolutionary Computing, Computational Complexity, Multiagent Systems, Human-Computer Interaction, and Robotics.

The datasets are divided into training, validation, and test sets using a split ratio of 60:10:30. The abstract of each paper is given to the models as the input. We employ Binary Cross-Entropy Loss as the loss function, treating each label as an independent binary classification problem. The utilized BERT model is the pre-trained bert-base-uncased model, wherein each token is represented by a 768-sized vector. The total number of training epochs is set to 50, accompanied by an early stopper with a patience of 4. If the validation loss of a model continues to rise for more than 4 epochs, the training process is halted. Additionally, the learning rate is set to 1e-5.

B. Evaluation Metrics

1) Accuracy: Accuracy is a commonly used metric to evaluate the performance of a classification model. It measures the proportion of correctly classified instances out of the total number of instances in a dataset. It is defined as below:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

TP (True Positives): The number of instances that are correctly predicted as positive.

TN (True Negatives): The number of instances that are correctly predicted as negative.

FP (False Positives): The number of instances that are incorrectly predicted as positive.

FN (False Negatives): The number of instances that are incorrectly predicted as negative.

2) Precision: This is the proportion of true positive predictions out of all positive predictions.

$$Precision = \frac{TP}{TP + FP}$$

3) Recall: This is the proportion of true positive predictions out of all actual positive instances.

$$Recall = \frac{TP}{TP + FN}$$

4) F1 Score: The F1 score is a metric that combines both precision and recall into a single value. It is particularly useful when dealing with imbalanced datasets. The F1 score ranges from 0 to 1, with 1 being the best score. It provides a balance between precision and recall and is particularly useful when the dataset is imbalanced.

$$F1\ score = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

C. Results

Each experiment was conducted five times, and the average results are presented. The performance of the models on the two datasets is illustrated in Tables I and II. Notably, the BERT-LSTM-Attention model demonstrates superior performance compared to other models. While the F1 scores across all models surpass 0.8, the accuracy achieved on the arXiv dataset significantly outperforms that on the Analytics Vidhya Hackathon Dataset. This discrepancy may arise from the fact that the Analytics Vidhya Hackathon Dataset contains a considerably smaller number of samples compared to the arXiv dataset.

TABLE I. PERFORMANCE ON ANALYTICS VIDHYA HACKTHON DATASET

Model	Accuracy	F1 score
Basic BERT	0.680	0.824
BERT-LSTM-Attention	**0.687**	**0.833**
BERT-CLS-Pooling	0.674	0.830
BERT-LSTM-All	0.681	0.819
BERT-All	0.640	0.810

TABLE II. PERFORMANCE ON ARXIV DATASET

Model	Accuracy	F1 score
Basic BERT	0.817	0.816
BERT-LSTM-Attention	**0.824**	**0.826**
BERT-CLS-Pooling	0.807	0.809
BERT-LSTM-All	0.813	0.815
BERT-All	0.737	0.764

V. CONCLUSIONS

In this paper, we have developed machine learning models designed to classify documents into distinct categories. We explore the potential of leveraging the summary information provided by various layers of the encoding blocks in the BERT model, as well as the word embeddings generated by the BERT model, to enhance the performance of classification tasks. We conducted experiments using several models, and the one that combined the attention mechanism and LSTM yielded the most favorable results. This finding underscores the fact that employing LSTM and the attention mechanism can effectively enhance the performance of the model.

REFERENCES

[1] J. Devlin, M. -W Chang, K. Lee, K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pp. 4171–4186*, Minneapolis, Minnesota, 2019.

[2] X. Ye, Y. Lu and S. Manoharan, "Automated Conversion of Engineering Rules: Towards Flexible Manufacturing Collaboration," *Results in Engineering,* vol. 16, p. 100680, December 2022.

[3] X. Ye, Z. Tang and S. Manoharan, "From Audio to Animated Signs," in *9th International Conference on Electrical and Electronics Engineering*, 2022.

[4] J. Wen, S. Manoharan and X. Ye, "Blockchain-based Reviewer Selection," in *2022 IEEE Asia-Pacific Conference on Computer Science and Data Engineering (CSDE)*, 2022.

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

[6] B. Jang, M. Kim, G. Harerimana, S. Kang and J. Kim, "Bi-LSTM Model to Increase Accuracy in Text Classification: Combining Word2vec CNN and Attention Mechanism," *Applied Sciences,* vol. 10, no. 17, pp. 5841,, 2020.

[7] R. Jing, "A Self-attention Based LSTM Network for Text Classification," *Journal of Physics: Conference Series,* vol. 1207, p. 012008, 2019.

[8] S. Yu, D. Liu, W. Zhu, Y. Zhang and S. Zhao, "Attention-based LSTM, GRU and CNN for short text classification," *Journal of Intelligent & Fuzzy Systems,* vol. 39, no. 1, p. 333–340, 2020.

[9] S. Zheng and M. Yang, "A New Method of Improving BERT for Text Classification," in *Intelligence Science and Big Data Engineering. Big Data and Machine Learning*, Springer, 2019, p. 442–452.

[10] T. Bao, N. Ren, R. Luo, B. Wang, G. Shen and T. Guo, "A BERT-Based Hybrid Short Text Classification Model Incorporating CNN and Attention-Based BiGRU," *Journal of Organizational and End User Computing,* vol. 33, no. 6, p. 1–21, 2021.

[11] G. Fan, C. Zhu and W. Zhu, "Convolutional neural network with contextualized word embedding for text classification," in *International Conference on Image and Video Processing, and Artificial Intelligence*, 2019.

[12] H. Zhou, "Research of Text Classification Based on TF-IDF and CNN-LSTM," *Journal of Physics: Conference Series,* vol. 2071, p. 012021, 2022.

[13] E. Xu, D. Qin, J. Huang and J. Zhang, "Multi Text Classification Model Based on BERT-CNN-BiLSTM," in *2022 IEEE 5th International Conference on Big Data and Artificial Intelligence*, 2022.

[14] L. Tran, L. Pham, T. Tran and A. Mai, "Text classification problems via BERT embedding method and graph convolutional neural network," in *2021 International Conference on Advanced Technologies for Communications*, 2021.

[15] L. Cai, Y. Song, T. Liu and K. Zhang, "A Hybrid BERT Model That Incorporates Label Semantics via Adjustive Attention for Multi-Label Text Classification," *IEEE Access,* vol. 8, 2020.

[16] Y. Song, J. Wang, Z. Liang, Z. Liu and T. Jiang, "Utilizing BERT Intermediate Layers for Aspect Based Sentiment Analysis and Natural Language Inference," *arXiv,* vol. 2002.04815, 2020.

[17] J. Lehečka, J. Švec, P. Ircing and L. Šmídl, "Adjusting BERT's Pooling Layer for Large-Scale Multi-Label Text Classification," in *23rd International Conference on Text, Speech, and Dialogue*, 2020.

[18] R. Yarullin and P. Serdyukov, "BERT for Sequence-to-Sequence Multi-label Text Classification," in *Analysis of Images, Social Networks and Texts*, 2021.

[19] J. Zhou, C. Ma, D. Long, G. Xu, N. Ding, H. Zhang, P. Xie and G. Liu, "Hierarchy-Aware Global Model for Hierarchical Text Classification," in *Annual Meeting of the Association for Computational Linguistics*, 2020.

[20] "MATCH: Metadata-Aware Text Classification in A Large Hierarchy," in *Web Conference*, 2021.

[21] Y. Zhang , Z. Shen, C. Wu, B. Xie, J. Hao, Y. Wang, K. Wang and J. Han, "Metadata-Induced Contrastive Learning for Zero-Shot Multi-Label Text Classification," in *ACM Web Conference 2022*, 2022.

[22] C. Zhang and H. Yamana, "Improving Text Classification Using Knowledge in Labels," in *2021 IEEE 6th International Conference on Big Data Analytics*, 2021.

[23] Y. Xiong, Y. Feng, H. Wu, H. Kamigaito and M. Okumura, "Fusing Label Embedding into BERT: An Efficient Improvement for Text Classification," in *Findings of the Association for Computational Linguistics*, 2021.

[24] W. Liu, J. Pang, N. Li, X. Zhou and F. Yue, "Research on Multi-label Text Classification Method Based on tALBERT-CNN," *International Journal of Computational Intelligence Systems,,* vol. 14, no. 1, 2021.

[25] Y. Dong, P. Liu, Z. Zhu, Q. Wang and Q. Zhang, "A Fusion Model-Based Label Embedding and Self-Interaction Attention for Text Classification," *IEEE Access,* vol. 8, 2020.

[26] H. Guo, X. Li, L. Zhang, J. Liu and W. Chen, "Label-Aware Text Representation for Multi-Label Text Classification," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2021.

[27] N. Liu, Q. Wang and J. Ren, "Label-Embedding Bi-directional Attentive Model for Multi-label Text Classification," *Neural Processing Letters,* vol. 53, no. 1, p. 375–389, 2021.