BERT-Based Hybrid RNN Model for Multi-class Text Classification to Study the Effect of Pre-trained Word Embeddings

Shreyashree S, Pramod Sunagar, S Rajarajeswari, Anita Kanavalli
Department of Computer Science & Engineering
M S Ramaiah Institute of Technology (Affiliated to VTU)
Bengaluru, India

***Abstract*—Due to the Covid-19 pandemic which started in the year 2020, many nations had imposed lockdown to curb the spread of this virus. People have been sharing their experiences and perspectives on social media on the lockdown situation. This has given rise to increased number of tweets or posts on social media. Multi-class text classification, a method of classifying a text into one of the pre-defined categories, is one of the effective ways to analyze such data that is implemented in this paper. A Covid-19 dataset is used in this work consisting of fifteen pre-defined categories. This paper presents a multi-layered hybrid model, LSTM followed by GRU, to integrate the benefits of both the techniques. The advantages of word embeddings techniques like GloVe and BERT have been implemented and found that, for three epochs, the transfer learning based pre-trained BERT-hybrid model performs one percent better than GloVe-hybrid model but the state-of-the-art, fine-tuned BERT-base model outperforms the BERT-hybrid model by three percent, in terms of validation loss. It is expected that, over a larger number of epochs, the hybrid model might outperform the fine-tuned model.**

Keywords*—Multi-class text classification; transfer learning; pre-training; word embeddings; GloVe; bidirectional encoder representations from transformers; long short-term memory; gated recurrent units; hybrid model; RNN

I. Introduction

On March 11th, 2020, the World Health Organization (WHO) proclaimed Covid-19 a global pandemic, making human lives increasingly digital. This massive amount of digital data aids data scientists in discovering new patterns and gaining a new perspective on any area of interest. With the rise of Artificial Intelligence (AI) in data science, machines might have the ability to perform all human tasks much better than humans. Natural Language Processing (NLP), a sub-domain of AI, is an interesting research field, in which, Text Classification (TC) is a simple and yet a challenging problem that is well-recognized in the domain. It is a process of categorizing samples of text into few pre-defined categories/classes, which are of two types, viz, binary classification and multi-class text classification (MTC). Applications of TC range from sentiment analysis to topic labelling. Using TC, we can easily categorize emails, social media posts like Tweets etc. to maintain and understand the text better for making any data-driven business decisions.

The approaches to perform TC are rule-based (uses hand-crafted rules), deep-learning techniques (uses neural networks) and hybrid methods. Out of these techniques, the most significant one is the deep learning method because they are powerful and provide good results [1-2]. And this paper concentrates on classifying a Covid-19 twitter dataset of into 15 pre-defined categories. There are two parts for TC, the first part being the feature engineering, where one of its methods called word embedding is used and the second part being the classification. The main objective of this paper is to perform a comparative study on the performance of hybrid classifiers with their respective pre-trained word embedding techniques. This project performs a comparative analysis between 1) hybrid Recurrent Neural Network (RNN) model with the help of either Global Vectors for Word Representation (GloVe) and Bidirectional Encoder Representations from Transformers (BERT) pre-trained word embeddings and 2) BERT-base model. The main reason for choosing hybrid architecture over others is that it helps in boosting the performance of the overall model. With regards to embeddings methods, BERT was mainly chosen because of the following reasons.

1) It provides contextual embeddings.

2) It considers the order of words before providing the embeddings.

3) While other pre-trained embedding models have pre-generated embeddings, BERT has to be trained to generate dynamic embeddings (as it considers context).

4) It generates embeddings for Out-Of-Vocabulary words.

In order to study them, a Covid-19 twitter dataset has been used, that contains approximately two lakhs of tweets and their respective labels. There are 15 different categories of tweets in this dataset.

II. Literature Survey

Shah et al. [3] have developed a text classification system for BBC news by using three traditional machine learning algorithms separately, namely Logistic Regression, K- Nearest Neighbor and Random Forest Algorithms, and compared these models to choose the best one. The classification is divided into four parts, viz text pre-processing, text representation, implementation of classifier and finally the classification of news. Text pre-processing involved removing of stop words and stemming, text representation involved the use of TF-IDF

algorithm to convert the text into suitable format. The comparison between the classifiers has been done in terms of five metrics: Precision, Accuracy, F1-score, Confusion matrix and support. According to the experiment, logistic regression performed the best with 97% accuracy. Kumar et al. [4] have provided the method of text mining using popular machine learning classification algorithms and has also provided a SWOT analysis of these algorithms to summarize the work done so far in the usage of ML classification algorithms on the task of sentiment analysis, one of the major tasks of text classification. Authors have also observed that most of the classification algorithms use bag-of-words for representing text. As sentiment analysis is a significant part of text classification, it can be performed either using machine learning approach or a lexicon-based approach. Harjule et al. [5] have explored both the methods by using SentiWordNet and Word Sense Disambiguation in the former approach and Multinomial Naïve Bayes (MNB), Logistic Regression, SVM and RNN in the latter approach. Along with the above-mentioned classifiers, an ensemble classifier consisting of MNB, SVM and LR is also implemented. Later, these classifiers are being compared. The text pre-processing is done using NLTK that involves casing, removal of stop words, punctuations, URLs and hashtags, POS tagging and tokenization. The datasets used are "Sentiment140" and "Crowdflower's Data for Everyone library". The observations indicate that the RNN model (LSTM) provides better results. Xia Sun et al. [6] have proposed a different approach to SA, where in the context of the text were captured using Bi-GRU and many DL models were used to classify. Among them, CNN+LSTM model outperformed all. The main focus of their work is the discovery of Drop Loss, which focuses on hard examples i.e., texts that are easier to get misclassified. This way, the classification accuracy was improved upon four sentiment datasets viz. MOOC, IMDB-2, IMDB-10 and SST-5. The CNN+LSTM architecture was also used by Giannopoulou et al. [7] to categorize e-books into pre-defined book categories using their table of contents as the text samples. Software As-A Service (SaaS) is one of the popular software delivery models. Customers should be clarified on which SaaS provider is best suited for them, in order to use cloud services. There are many service quality pillars [8] that are to be considered before choosing a right provider. Hence, Raza et al. have performed multi-class text classification on these customer reviews, with service quality pillars as categories. The classification algorithms used are machine learning algorithms and an ensemble of all the ML algorithms. The representation of text is done using TF-IDF technique after text cleaning. The results show that Logistic Regression performs better than all the models, including the ensemble model. There is also the field of citation intent classification that can be benefitted by different word embedding techniques.

Roman et al. [9] have used word embedding techniques like GloVe, InferSent and BERT to classify the citation context with citation intent, on 10 million records of Citation Context Dataset. It has been observed that the method using BERT provides a highest of 89% precision of all. Before BERT or transformer models were discovered, Bi-LSTM architectures were leading in many of the downstream tasks of NLP. Hence, Huang et al. [10] have experimented by combining Bi-LSTM with transformers. Considering the fact that adding more hidden layers to BERT will not improve its performance, the authors have added a Bi-LSTM layer to each of the transformer entity, called TRANS-BLSTM, and have observed that their model provides an F1-score of 94.01% on SQUAD 1.1 development dataset. Hao Wu et al. [11] have proposed a weighted multi-class text classification model where the text is converted to its numerical terms using Word2Vec technique; weights are applied to those vectors using TF-IDF algorithm, and the word vectors are multiplied with these weights to provide the final representation of text. Context is captured using a BiLSTM layer, followed by an Attention layer and a softmax layer to classify. This model has observed to have 91.26% accuracy. Kumar et al. [12] have also used Bi-LSTM layers in their proposed model, SAB-LSTM, where they have applied model and network optimizer with a dropout layer around the Bi-LSTM layer to provide best accuracy when trained on COVID-19 dataset, in comparison with LSTM and Bi-LSTM models individually. Another hybrid model CNN+RNN with attention mechanism was proposed by Guo et al. [13] to perform MTC.

The text classification method also finds its application in the field of medicine prescription, where, it can be detected whether the prescribed medicine has been misused or not. Al-Garadi et al. [14] have experimented in this domain on a Twitter dataset using BERT and its variants but with fusion. The fusion models involved combining the probabilities of each text sample from BERT and its variants using either a logistic regression classifier or a Naïve Bayes classifier. These fusion models were observed to provide higher accuracy than the individual transformer models. Shaik at al. [15] have developed a text classification model that classifies the course learning outcomes (CLOs) and assessment texts into a pre-defined class of Bloom's taxonomy, contributing to the education domain. This model uses Skip-gram word embedding technique and LSTM classifier to perform MTC, which provides an accuracy of 87% on CLOs and 74% on assessment texts. The Skip-gram technique is also used by Aslam et al. [16] to perform MTC on Google Play app reviews using CNN as its classifier with a precision of 95.49%. The CNN is also combined with Bi-LSTM using attention layer having Word2Vec as word embedding technique, proposed by Zhenget al. [17]. Similarly, CNN is combined with GRU layers as an ensemble model to perform MTC on news sources by John et al. [18], to help women select the state they want to travel or relocate to, based on the recent criminal activities. As a better alternative to CNN, CapsNets are used along with Bi-GRU layers as a hybrid model, using Word2Vec technique to perform Text classification by Gangwar et al. [19]. The detection of fake news also is a significant sub-task of MTC, where in IulianIlie et al. [20] have proposed a comparative study of 10 DNN models using GloVe, Word2Vec and FastText word embedding techniques, in which RCNN performed the best. The fastText method is used as a feature extraction method and also as a classification method to classify emails into multiple classes [21]. Aydoğan et al. [22] have performed a comparative study between CNN, LSTM, RNN and GRU networks on Turkish datasets, using word2Vec word embeddings. The results indicate that both

LSTM and GRU perform the best. Sunagar et al. [23] have conducted a comparative study between various deep learning models for the task of MTC on Covid-19 dataset, amongst which, RNN with Bidirectional LSTM performed better. The comparison between ML algorithms was also considered for the task of news topic classification [24] to study the different ML models. Many researchers have also carried out the works like detecting the disease, predicting the end of pandemic [25] and creating a decision support system [26] for Covid-19. Many authors have carried out the research on the features extraction from text and tried to establish how this will help in attaining the better accuracy [27-29].

In the existing system, classification of the text focuses on Sentiment Analysis, Movie Review etc. Due to Covid-19 pandemic, lot of tweets are being generated on various topics like, safety measures, social distancing, advisories, etc. by Government agencies, WHO, Scientists, NGOs and individuals. Classifying these tweets into different categories like Social Distancing, Vaccination, Advisories etc. is one of the motivations for the taking up this project. The recent works do contribute to the accuracy of the models discovered, may it be hybrid, traditional or an ensemble model. But this paper mainly focuses on the fact that, more the number of neural network layers, more the accuracy, with regards to the hybrid RNN model, that is inspired by the work of Sunagar et al. [30]. To boost the accuracy of the model, the BERT embeddings are being considered along with the hybrid RNN model, as they are contextual in nature and support Out-Of-Vocabulary words. And this model is being compared with the fundamental BERT-base model, by considering the importance of pre-trained word embeddings.

III. Proposed Model

Text classification is a challenging yet interesting problem of NLP. It is a method of classifying sample texts into few pre-defined categories. The categories can be two or more in number. If the number of categories is two in number, it is called binary classification. The applications of binary classification are Sentiment Analysis, Spam filtering, Credit Card fraud detection etc. If the number of categories is more than two, then it is called multi-class text classification. The applications of MTC are Product categorization, News categorization, Citation intent classification, E-book classification etc. This project focuses on performing MTC on a COVID dataset consisting of tweets collected from Twitter and Kaggle with 15 unique categories.

A. Architecture of Proposed and Related Models

1) The process of building an MTC model involves two parts: Extraction of Embeddings: In order to perform MTC on the above dataset, the model that is built to do that only understands numerical data and not the raw text. Hence, it is necessary to convert the text samples into numerical vectors. This conversion of raw text data into numerical values is called a word embedding technique [31]. There are two types of word embedding techniques: Frequency-based and Prediction-based.

One of the earliest prediction-based embedding techniques is Word2Vec [32], which is a contextual word embedding method that provides an association between words having similar meaning. There are two models of Word2Vec method to use, in order to create word embeddings. 1) CBOW (Continuous Bag-Of-Words) model – which takes context words as input and tries to predict the target word as output. 2) Skip-gram model – predicts context words given the target word. The disadvantages of Word2Vec model are that it cannot handle out-of-vocabulary words, it relies on local information, requires large corpus to get an optimal solution and word sense is not captured separately. In order to consider the co-occurrence of words in the document, GloVe was invented [33]. This embedding technique was developed by Stanford University that is used to generate embeddings using an unsupervised approach. The training of GloVe model involves the use of global word-word co-occurrence matrix that is points out the number of times each word co-occurs with another.

Fig.1 shows an example of a co-occurrence matrix, where each row consists of unique words in the document and each column denotes the context. Here, the context length in one. E.g., It says that the word “digital” co-occurs with the word “computer” 1670 times for the selected corpus. This large matrix is factorized to provide a lower-dimensional matrix, where, each row of the lower-dimensional matrix acts as word vectors for the respective word. The model has been pre-trained to provide various dimensional word vectors. This project concentrates on using word vectors of dimension 50, where the model is trained on 6 billion tokens taken from both Wikipedia 2014 and Gigaword 5. Hence, the pre-trained word vectors will be inside a text file of name “GloVe.6B.50d.txt”.Unlike Word2Vec model, this method uses global information to construct the embeddings. The main disadvantages of using GloVe model are that it requires large memory to store the co-occurrence matrix, it cannot handle out-of-vocabulary words and word Sense is not supported. These and many other pre-trained models have one disadvantage in common, which is the unidirectional that restricts the power of those models. This led to the discovery of a specific type of transformer networks [34], BERT [35-37]. It is a pre-trained model that considers the left and right context of a word in all layers, while generating embeddings.

In order to pre-train BERT, WordPiece embeddings are used with 30,000 vocabulary size. There are two special tokens inserted into each sentence of BERT’s input, they are [CLS] and [SEP] tokens. [CLS] token represents the beginning of every sequence, which is also a classification token, for the task of NSP. Every sequence in input is separated by [SEP] token. The BERT has a powerful input representation, shown in Fig. 2, which is a combination of three embeddings: Token embeddings: Embeddings that represent each token in the document, Segment embeddings: Embeddings that are used to identify to which segment/sentence does the token belong to and Position embeddings: Embeddings representing the position of each token. There are two pre-training tasks of BERT, shown in Fig. 3, Masked Language Modeling (MLM) and Next Sequence Prediction (NSP).

	aardvark	...	computer	data	result	pie	sugar	...
cherry	0	...	2	8	9	442	25	...
strawberry	0	...	0	0	1	60	19	...
digital	0	...	1670	1683	85	5	4	...
information	0	...	3325	3982	378	5	13	...

Fig. 1. A Sample Co-occurrence Matrix.

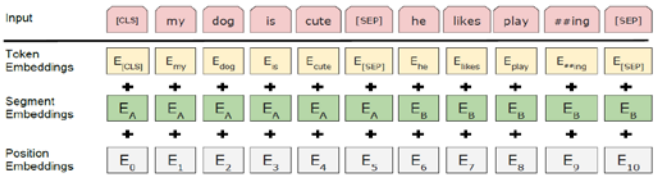

Fig. 2. Input Representation of BERT.

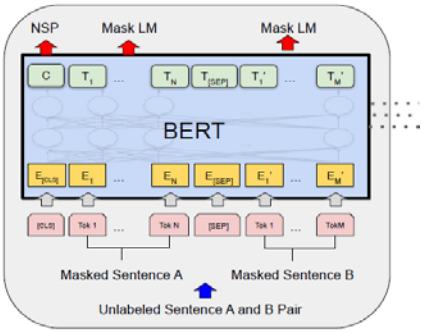

Fig. 3. Pre-training Architecture of BERT.

This paper intends to implement two word embedding methods, 1) GloVe 2) BERT-base architecture, that contains 12 transformer encoder layers, 12 self-attention heads, 768 as hidden size and 110M parameters. GloVe and BERT models are used widely due to the use of transfer learning. It is a method that saves training time of deep learning models for the data scientists. In traditional learning, the model used to be built from scratch [38] [39]. If there is a task of classification of reviews into positive and negative, and a task of classifying spam email, both these tasks were implemented separately by building two independent models from scratch. This might consume a lot of training time in general. But in case of transfer learning, the model built for classifying reviews is used as an initial checkpoint for the task of classifying spam emails. The latter task is implemented by just fine-tuning or adjusting the weights of the former model. The model that is trained for the former task is called the pre-trained model [40-41]. The latter model is called the fine-tuned model. Hence, transfer learning is a method of using the knowledge, gathered while training a model for a task, to train a similar task. Another main advantage of using transfer learning approach is that for the fine-tuning approach, the second similar task need not have a large dataset. This project uses two pre-trained models for generating word embeddings.

2) Classifier: At the early days of neural networks, feed forward networks (FFN) were very popular to perform many tasks. They were known for their accuracy and speed. But they had their own disadvantages:

a) Unable to process sequential data.

b) Current input can't be considered.

c) Cannot remember previous inputs.

RNN (Recurrent Neural Networks) were discovered to overcome the above-mentioned problems. They belong to a class of neural networks which takes previous layer outputs and feeds them as input to the current layer, passing information from the past. This is implemented with the concept of "memory" that keeps information about previous calculations till time step t. The main reason why RNN is used in NLP is because text is a sequential data. But RNNs suffer from vanishing gradient problem, in which, the gradients are so small that the updates of parameters are insignificant. This problem occurs while processing long sequences. Hence, there are two variants of RNN that have been discovered: LSTM (Long-Short Term Memory) and GRU (Gated Recurrent Units). In RNN, in order to add new information, the whole memory context is modified and there is no consideration for important information. To overcome this, LSTM is used [42]. LSTM has the ability to forget or restore information of choice. And it is implemented by three Gating mechanisms. 1) Forget Gate: This gate is used to forget all the insignificant information from the memory context. 2) Input Gate: This gate is used to add or update new information into the memory context. 3) Output Gate: This gate is responsible for selecting important information and passing it out to the downstream network. GRU, on the other hand, is another variant of RNN similar to LSTM, except that it has two Gating mechanisms [43]: 1) Reset Gate: This gate is responsible for deciding how much of previous information should be forgotten. 2) Update Gate: This gate is responsible for deciding how much of previous information should be passed along the network. This paper uses both LSTM and GRU layers to classify tweets. Fig. 4 shows the architecture of the proposed hybrid RNN model.

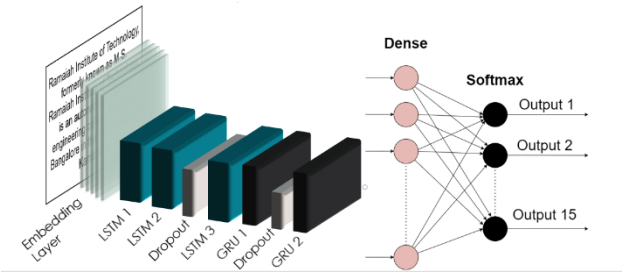

Fig. 4. Architecture Diagram of Hybrid Model.

The input word sequence is provided to the input layer, where tokenization (Word/Sub-word, depending on the embedding technique) takes place and the tokenized data is given to the second layer (Embedding Layer). This layer can use either GloVe or BERT pre-trained word embedding techniques to generate meaningful word embeddings. These embeddings are given to the third layer, which is a combination of LSTM, Dropout and GRU layers. This is the hybrid classifier. The fourth layer is the softmax output layer which gives out the probability for all the classes, with highest probability for the predicted class.

B. Working of Models

Fig. 5 depicts the generic workflow. The dataset used in this project contains tweets collected from Twitter till April 2020. The total number of tweets collected are 260000. There are three attributes in the dataset: Tweets, labels and label ids.

All the tweets are labelled as one of the 15 labels. These tweets, before feeding into the models, have to be cleaned and pre-processed such that it is easier for the models to learn quickly.

Algorithm for proposed model

Input: The COVID dataset

Output: A model trained on the COVID dataset and one of the 15 pre-defined classes for each tweet in the test dataset

1. Import the dataset.
2. Pre-processed_tweets, labels = Data_PreProcessing(dataset).
3. Split the Pre-processed_tweets and labels into training and testing set with a ratio of 80:20.
4. Either perform BERT_Tokenization() or GloVe_Tokenization() depending on the choice of embedding technique.
5. Create an embedding matrix for every word in the vocabulary
6. Build hybrid model as shown in step 7 to step 13.
7. Add an EmbeddingLayer() with weights as the embedding matrix
8. 3 LSTM() Layer
9. Dropout layer
10. 1 LSTM() layer& 1 GRU Layer
11. Dropout layer
12. 1 GRU() Layer
13. Dense() layer with Softmax activation function.
14. Train the model on the training set.
15. Evaluate the model on the test set.

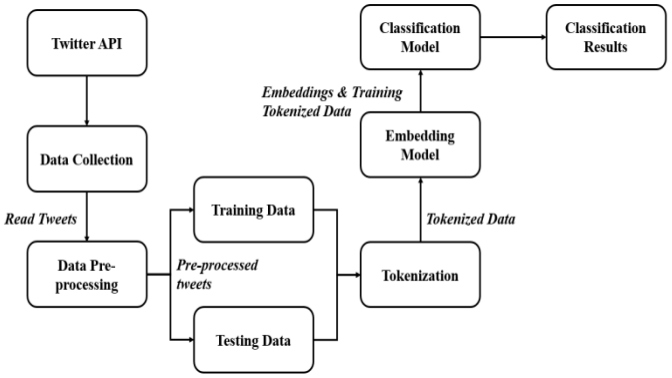

Fig. 5. Generic Workflow.

After the pre-processing, the dataset is split into training and testing set with 80:20 ratio, where each sample from each set is tokenized (GloVe or BERT Tokenization) and is given to the embedding model (GloVe or BERT) to generate the embeddings which is given to the classification model (Hybrid or BERT-base) to produce the results (one of the 15 classes of the dataset). There are two ways to use BERT model. 1) Fine-tuning approach [44] – Using the BERT model as a whole to generate embeddings from 12 encoders and 1 classifier. 2) Embeddings extraction approach – Using BERT model to just extract the embeddings and use a different classifier altogether. This project suggests the use of Hybrid RNN model for this approach as a classifier. The reason being, LSTM is more accurate and complex compared to GRU because of the number of gating mechanisms and in terms of speed, GRU is better than LSTM. Hence, to combine the advantages of both the models this hybrid model has been proposed. For the second approach, the embeddings can be extracted in various ways. We can extract the embeddings from 1) the last layer 2) the last few layers and sum them 3) the last few layers and get an average. This paper uses the second approach. Following is the algorithm for the approach towards the hybrid model.

IV. Results and Analysis

A. Dataset

This paper uses a COVID-19 dataset that is prepared with tweets from twitter and Kaggle in the period 2020-21. The structure of this dataset contains tweets, labels in words and label ids. In terms of MTC, the dataset contains 15 unique classes to categorize the tweets that are: Entertainment, Essential Workers, Facts, General, Government Action, Medical Test & Analysis/Supply, Tribute, Pandemic, Panic Shopping, Political, Self-Care, Social Distance, Stay-At-Home, Taco Tuesday and Telecommuting Life.

B. Experimental Settings

The setup requires Google Colab Pro subscription that is linked to Google Drive, where the dataset is contained. The BERT repository is cloned in the Colab platform, and is used to access files that help in extracting embeddings from BERT's encoder layers. The python files extracted for this purpose are modified for the current use case accordingly. Also, the configuration files of uncased-BERT model, "uncased_L-12_H-768_A-12", is downloaded to the drive, that contains vocabulary file, configuration file and checkpoint of pre-trained BERT-base model. These files are given as parameters to the BERT repository files to extract embeddings. Following is the list of all main parameters considered for configuring all 3 models:

For both the Hybrid models:

- Number of LSTM layers: 3
- Number of GRU layers: 2
- Number of dropout layers: 2
- Output function: SoftMax
- Learning rate: 0.001
- Batch size: 256
- Optimizer: Adam
- Dropout rate: 0.5

For BERT-Hybrid model:

- Maximum sequence length: Maximum sentence length
- Embedding dimension: 768
- Vocabulary size: 30522
- No. of encoder layers for embeddings extraction: 4

For GloVe-Hybrid model:

- Maximum sequence length: 500
- Embedding dimension: 256

Vocabulary size: 1 lakh approx.

C. Performance Measures

Table I describe the validation loss, validation accuracy, precision, recall and F1-score of BERT-hybrid, GloVe-hybrid and BERT-base models. From the tables, we can say that, for three epochs, BERT-base model performs better than the hybrid models with an accuracy of 96.59%. It is estimation that for larger number of epochs, hybrid model might work better than the base model. If we compare between the two hybrid models, the model that uses BERT embeddings shows a slight improvement in performance, indicating that the use of embeddings plays a major role in deciding the performance of any model.

TABLE I. LOSS, ACCURACY, PRECISION, F1-SCORE AND RECALL OF MODELS

MODEL	LOSS	ACCURACY	PRECISION	RECALL	F1-SCORE
GLOVE-HYBRID	0.156	0.953	95.74	95	95.34
BERT-HYBRID	0.1475	0.9568	95.75	95.55	95.63
BERT-BASE	0.1185	0.9659	96.93	96.85	96.88

Fig. 6 depicts the training and prediction time of implemented models. As the BERT-base model is more complex in architecture, it takes more time to train and predict. BERT-hybrid model takes least amount of time to train, with a smaller number of parameters and small vocabulary size. Also, the time-consuming task of generating embeddings is a one-time process for the BERT-hybrid model, and hence the small training time. The GloVe hybrid model takes more time to train than BERT-hybrid model due to its larger vocabulary size and parameters. With respect to the prediction time, BERT-hybrid takes less time to predict than the other models, approximately five minutes, which leads to a fact that there is a trade-off between time and accuracy when we use BERT-hybrid model. Fig. 7 and Fig. 8 below show the comparison between models on validation accuracy and validation loss in a graphical format respectively.

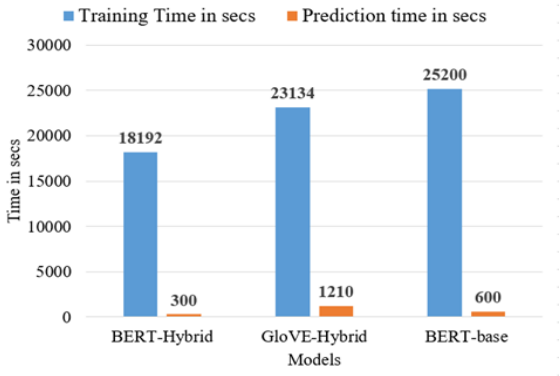

Fig. 6. Training and Prediction Time of Models.

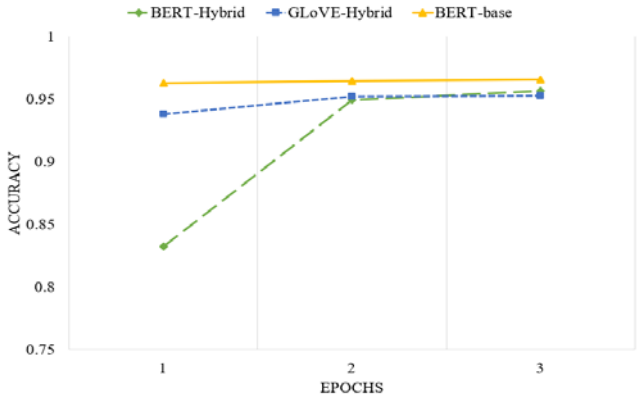

Fig. 7. Validation Accuracy of Models.

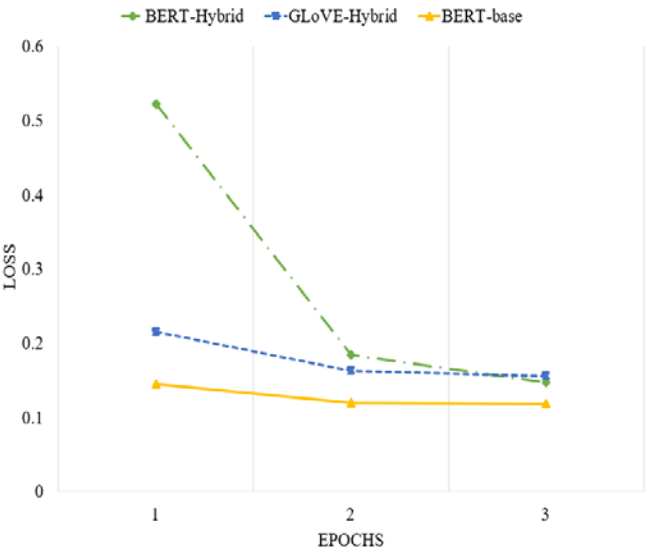

Fig. 8. Validation Loss of Models.

V. CONCLUSION AND FUTURE WORK

The research presents a novel hybrid RNN model that experiments between GloVe and BERT embeddings. In terms of accuracy, performance, and speed, this hybrid model utilizes the capabilities of both the LSTM and GRU layers to fill in the gaps. It also uses many layers of LSTM and GRU to apply the concept that deeper the model, greater the accuracy. For three epochs, it is shown that the state-of-the-art BERT-base transformer model outperforms both hybrid RNN models, with an accuracy of 96.59 %. It is expected that the hybrid RNN models will perform better over a larger number of epochs. Furthermore, the BERT-hybrid model outperforms the GloVe-hybrid model, demonstrating that contextual representation improves performance. In the future, different BERT model versions might be utilized to produce embeddings and feed them to the hybrid model for better performance.

ACKNOWLEDGMENT

This work was supported by M S Ramaiah Institute of Technology, Bangalore-560054, and Visvesvaraya Technological University, Jnana Sangama, Belagavi -590018.

REFERENCES

[1] S. Minaee, N. Kalchbrenner, E. Cambria, N. Nikzad, M. Chenaghlu, J. Gao,"Deep Learning--Based Text Classification: A Comprehensive Review", ACM Computing Surveys (CSUR), vol. 54, pp. 1-40, 2021.

[2] S. Dong, P. Wang, K. Abbas,"A Survey on Deep Learning and its Applications", Computer Science Review, vol. 40, p. 100379, 2021.

[3] K. Shah, H. Patel, D. Sanghvi, M. Shah,"A Comparative Analysis of Logistic Regression, Random Forest and KNN Models for the Text Classification", Augmented Human Research, vol. 5, pp. 1-16, 2020.

[4] A. Kumar, V. Dabas, P. Hooda,"Text Classification Algorithms for Mining unstructured data: a SWOT analysis", International Journal of Information Technology, vol. 12(4), pp. 1159-1169, 2020.

[5] P. Harjule, A. Gurjar, H. Seth, P. Thakur,"Text classification on Twitter data",in 3rd International Conference on Emerging Technologies in Computer Engineering: Machine Learning and Internet of Things (ICETCE), pp. 160-164, IEEE, 2020.

[6] X. Sun, Y. Gao, R. Sutcliffe, S.X. Guo, X. Wang, J. Feng,"Word Representation Learning Based on Bidirectional Grus with Drop Loss for Sentiment Classification", IEEE Transactions on Systems, Man, and Cybernetics: Systems, vol. 51, pp. 4532-4542, 2021.

[7] E. Giannopoulou, N. Mitrou,"An AI-Based Methodology for the Automatic Classification of a Multiclass Ebook Collection Using Information from the Tables of Contents", IEEE Access, vol. 8, pp. 218658-218675, 2020.

[8] M. Raza, F.K. Hussain, O.K. Hussain, M. Zhao, Z. Rehman,"A Comparative Analysis of Machine Learning Models for Quality Pillar Assessment of Saas Services by Multi-Class Text Classification of Users' Reviews", Future Generation Computer Systems 101,pp. 341-371, 2019.

[9] M. Roman, A. Shahid, S. Khan, A. Koubaa, L. Yu,"Citation Intent Classification Using Word Embedding", IEEE Access, vol. 9, pp. 9982-9995, 2021.

[10] Z. Huang, P. Xu, D. Liang, A. Mishra, B. Xiang,"TRANS-BLSTM: Transformer with Bidirectional LSTM for Language Understanding", arXiv preprint arXiv:2003.07000, 2020.

[11] H. Wu, Z. He, W. Zhang, Y. Hu, Y. Wu, Y. Yue, "Multi-Class Text Classification Model Based on Weighted Word Vector and Bilstm-Attention Optimization",in International Conference on Intelligent Computing, Springer, Cham. , pp. 393-400, 2021.

[12] D.A. Kumar, A. Chinnalagu,"Sentiment and Emotion in Social Media COVID-19 Conversations: SAB-LSTM Approach"in 9th International Conference System Modeling and Advancement in Research Trends (SMART), IEEE, pp. 463-467 ,2020.

[13] L. Guo, D. Zhang, L. Wang, H. Wang, B. Cui,"CRAN: A Hybrid CNN-RNN Attention-Based Model for Text Classification"in International Conference on Conceptual Modeling, pp. 571-585, Springer, Cham, 2018.

[14] M.A Al-Garadi, Y.C Yang, H. Cai, Y. Ruan, K. O'Connor, G.H Graciela, J. Perrone, A. Sarker,"Text Classification Models for the Automatic Detection of Nonmedical Prescription Medication Use from Social Media", BMC Medical Informatics and Decision Making, vol. 21, pp. 1-3, 2021.

[15] S. Shaikh, S.M. Daudpotta, A.S. Imran,"Bloom's Learning Outcomes' Automatic Classification Using Lstm and Pretrained Word Embeddings", IEEE Access, vol. 9, pp. 117887-117909, 2021.

[16] N. Aslam, W.Y. Ramay, K. Xia, N. Sarwar,"Convolutional Neural Network Based Classification of App Reviews", IEEE Access, vol. 8, pp. 185619-185628, 2020.

[17] J. Zheng, L. Zheng,"A Hybrid Bidirectional Recurrent Convolutional Neural Network Attention-Based Model for Text Classification", IEEE Access, vol. 7, pp. 106673-106685, 2019.

[18] J. John, M.S. Varkey, M. Selvi,"Multi-Class Text Classification and Publication of Crime Data from Online News Sources"in 8th International Conference on Smart Computing and Communications (ICSCC), pp. 64-63, IEEE, 2021.

[19] A.K. Gangwar, V. Ravi,"A Novel Bgcapsule Network for Text Classification", SN Computer Science, vol. 3, pp. 1-2, 2022.

[20] V.I. Ilie, C.O. Truică, E.S. Apostol, A. Paschke,"Context-Aware Misinformation Detection: a Benchmark of Deep Learning Architectures Using Word Embeddings", IEEE Access, vol. 9, pp. 162122-162146, 2021.

[21] R. Tahsin, M.H. Mozumder, S.A. Shahriyar, M.A. Mollah,"A Novel Approach for E-Mail Classification Using Fasttext"in IEEE Region 10 Symposium (TENSYMP), pp. 1392-1395, 2020.

[22] M. Aydoğan, A. Karci,"Improving the Accuracy Using Pre-Trained Word Embeddings on Deep Neural Networks for Turkish Text Classification", Physica A: Statistical Mechanics and its Applications, vol. 541, p. 123288, 2020.

[23] P. Sunagar, A. Kanavalli, V. Poornima, V.M. Hemanth, K. Sreeram, K.S. Shivakumar,"Classification of Covid-19 Tweets Using Deep Learning Techniques"in Inventive Systems and Control, pp. 123-136, Springer, Singapore, 2021.

[24] P. Sunagar, A. Kanavalli, S.S. Nayak, S.R. Mahan, S. Prasad,"News Topic Classification Using Machine Learning Techniques"in International Conference on Communication, Computing and Electronics Systems, pp. 461-474, Springer, Singapore, 2021.

[25] S. Shwetha, P. Sunagar, S. Rajarajeswari, A. Kanavalli,"Ensemble Model to Forecast the End of the COVID-19 Pandemic"in 3rd International Conference on Communication, Computing and Electronics Systems, pp. 815-829, Springer, Singapore, 2022.

[26] F. Saleem, A.S. AL-Ghamdi, M.O. Alassafi, S.A. AlGhamdi,"Machine Learning, Deep Learning, and Mathematical Models to Analyze Forecasting and Epidemiology of COVID-19: A Systematic Literature Review", International Journal of Environmental Research and Public Health, vol. 9, p. 5099, 2022.

[27] Chandrika, C. P., & Kallimani, J. S. (2022). Authorship Attribution for Kannada Text Using Profile Based Approach. In Proceedings of the 2nd International Conference on Recent Trends in Machine Learning, IoT, Smart Cities and Applications (pp. 679-688). Springer, Singapore.

[28] Chandrika, C. P., & Kallimani, J. S. (2022). Instance Based Authorship Attribution for Kannada Text Using Amalgamation of Character and Word N-grams Technique. In Distributed Computing and Optimization Techniques (pp. 547-557). Springer, Singapore.

[29] P. Sunagar, A. Kanavalli and N D Shetty, "Feature Extraction and Selection Techniques for Text Classification: A Survey", International Journal of Advanced Research in Engineering and Technology, 11(12), 2020, pp. 2871-2881. doi: 10.34218/IJARET.11.12.2020.268.

[30] P. Sunagar and A. Kanavalli, "A Hybrid RNN based Deep Learning Approach for Text Classification" International Journal of Advanced Computer Science and Applications(IJACSA), 13(6), 2022. http://dx.doi.org/10.14569/IJACSA.2022.0130636

[31] D. Chandrasekaran, V. Mago,"Comparative Analysis of Word Embeddings in Assessing Semantic Similarity of Complex Sentences", IEEE Access, vol. 9, pp. 166395-166408, 2021.

[32] T. Mikolov, K. Chen, G. Corrado, J. Dean, "Efficient Estimation of Word Representations in Vector Space", arXiv preprint arXiv:1301.3781, 2013.

[33] J. Pennington, R. Socher, C.D. Manning,"GloVe: Global Vectors for Word Representation"in Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1532-1543, 2014.

[34] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, L. Kaiser, I. Polosukhin,"Attention is All You Need", Advances in neural information processing systems, vol. 30, 2017.

[35] J. Devlin, M.W. Chang, K. Lee, K. Toutanova,"Bert: Pre-Training of Deep Bidirectional Transformers for Language Understanding", arXiv preprint arXiv:1810.04805, 2018.

[36] R.K. Kaliyar,"A Multi-Layer Bidirectional Transformer Encoder for Pre-Trained Word Embedding: A Survey of Bert"in 10th International Conference on Cloud Computing, Data Science & Engineering (Confluence), pp. 336-340, IEEE, 2020.

[37] S. Shreyashree, P. Sunagar, S. Rajarajeswari, A. Kanavalli,"A Literature Review on Bidirectional Encoder Representations from Transformers", Inventive Computation and Information Technologies, pp. 305-320, 2022.

[38] F. Zhuang, Z. Qi, K. Duan, D. Xi, Y. Zhu, H. Zhu, H. Xiong, Q. He,"A Comprehensive Survey on Transfer Learning"in Proceedings of the IEEE, vol. 109, pp. 43-76, 2020.

[39] H. Liang, W. Fu, F. Yi,"A Survey of Recent Advances in Transfer Learning"in 19th International Conference on Communication Technology (ICCT), pp. 1516-1523, IEEE, 2019.

[40] J. Peng, K. Han,"Survey of Pre-Trained Models for Natural Language Processing" in International Conference on Electronic Communications, Internet of Things and Big Data (ICEIB), pp. 277-280, IEEE, 2021.

[41] X. Qiu, T. Sun, Y. Xu, Y. Shao, N. Dai, X. Huang,"Pre-Trained Models for Natural Language Processing: A Survey", Science China Technological Sciences, vol. 63, pp. 1872-1897, 2020.

[42] R.C. Staudemeyer, E.R. Morris,"Understanding LSTM--A Tutorial into Long Short-Term Memory Recurrent Neural Networks", arXiv preprint arXiv:1909.09586, 2019.

[43] R. Dey, F.M. Salem,"Gate-Variants of Gated Recurrent Unit (GRU) Neural Networks"in 60th International Midwest Symposium on Circuits and Systems (MWSCAS), pp. 1597-1600, IEEE, 2017.

[44] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, J. Davison,"Huggingface's Transformers: State-of-the-Art Natural Language Processing", arXiv preprint arXiv:1910.03771, 2019.