Interactive Web App for Fake News Detection

Sparsh Agarwal[1,*], *Malempati* Varun[1] and *S*. Prabakeran[1]

[1]Department of Networking and Communications, SRM Institute of Science and Technology, Kattankulathur, Chennai, India

Abstract. In the contemporary era of technology, individuals who utilize mobile phones and laptops have developed a preference for accessing news through online media. News organizations disseminate news and offer confirmation sources. However, the issue at hand is how to authenticate stories and articles shared on social networks such as WhatsApp groups, Facebook pages, Twitter, and other smaller blogs and social networking sites. It is hazardous for society to accept rumours disguised as news, especially in developing nations like India, where it is crucial to prevent rumours and specialize in honest and verified information. Classifying written articles as misleading or deceptive is not easy to automate, and even experts in a specific field must evaluate several factors before rendering a judgment regarding the validity of a message. This project proposes the use of a machine learning approach to automatically classify news articles. This endeavour explores numerous text characteristics that can be employed to differentiate fabricated news content from actual news.

1. Introduction

Platforms such as Facebook and Twitter have enabled an unparalleled amount of information sharing. Social media has allowed users to generate and distribute a vast amount of information, some of which may be untrue and lacking in factual basis. The automation of the process of identifying articles containing false information or disinformation poses a considerable challenge. Even specialists in each subject must consider multiple factors before rendering a judgment regarding the accuracy of a message. In this research endeavour, we propose employing automated news article categorization methods that use machine learning techniques. Our research examines a range of textual characteristics that can be utilized to distinguish between true and fabricated information.

*Corresponding author: sa7689@srmist.edu.in

We train a range of machine learning algorithms using different ensemble methods and assess their effectiveness using real- world datasets.

Specifically, we employ SJB-LSTM, BERT, and Dense Layer Model with Word2Vec to identify and mitigate erroneous information on various social media platforms. Another approach we take is to analyse the dissemination of fake news in relation to accurate news. We categorize reactions to articles theoretically to determine whether they are genuine or not. By examining the linguistic components of an article and social reactions to it, we can employ a more sophisticated approach to determine if an article is intrinsically deceptive.

Training a universal algorithm that performs optimally across different news domains is a challenging task, as articles from various genres exhibit diverse textual patterns. In this study, we develop a machine learning aggregation method to address the problem of detecting false news. Our research explores several textual characteristics that can be employed to distinguish between authentic and fraudulent information. We use these features to train various machine learning algorithms using different synthetic methods. Learning models frequently utilize strategies such as encapsulation and reinforcement to reduce error rates. This information assists the model in classifying news from different sources as either real or fake

2. Literature review

In the paper, "Detecting and Mitigating the Dissemination of Fake News: Challenges and Future Research Opportunities"[1], the proposed methods are: Automatic Detection, Language Specific Detection, Early Detection, Stance Detection, Feature Based Detection and Ensemble Learning. These methods can classify at most 4 types of fake news. It's very challenging to find out if the content in a certain news article is fake or real as even fabricated news articles may contain certain truth points. The models must be flexible enough to detect more than 1 type of fake news even if we have 4-5 mixed in one article. In the paper, "An Integrated Multi-Task Model for Fake News Detection " [2], machine learning models used were Hybrid - CNNs, LSTM - MMFD, Memory Network and FDML. FDML was chosen as the focus of research as it produced the highest accuracy and macro among all. The accuracy and macro, both, of every model drops with an increase in the number of topics being covered in that article. The research paper, "Fake News Detection Using Machine Learning"[3], used basic algorithms and approaches with sentiment classification as the main goal: Naive Bayes, NLP and TF IDF method. Used basic algorithms and approaches with sentiment classification as main goal: Naive Bayes, NLP and TF IDF method.

The paper, "The theatre of fake news spreading, who plays which role? A study on real graphs of spreading on Twitter ",[7] focuses on analyzing the dissemination of false and genuine news on the social media platform Twitter and the impact it has on the users of the social media platform. It makes use of 2 major concepts, namely, Dataset of Diffusion and Dataset with Graphs. The Dataset of Diffusion contains data of fake news and true tweets and retweets. The Dataset of Graphs gives a query of neighbors for the spreaders. It majorly helps us understand the features to focus on when looking for fake news which can spread the most among people [10].

"Fake news detection based on news content and social contexts: a transformer-based approach",[12] makes use of tokenization and N-layered Transformer Encoder and Decoder model followed by a classification layer and finally a cross-entropy loss function. It works on supervised learning of 2 labels, namely: "true" and "false". While

the training loss is a little smaller than the loss on the validation dataset, overall, both values are seen to converge. The model accuracy is 74.89%, the precision is 72.40%, the recall value is 77.68% and the F-1 score is 74.95%. The research in, "A Comprehensive Review on Fake News Detection with Deep Learning",[13] consisted of testing the following approaches to classification of fake news: CNN, RNN, GNN, GAN and BERT. Deep Learning methods produced an accuracy of average 98%, whereas LSTM with GRU produced a very low 56%. The highest accuracy of 99% was achieved by the passive aggressive method along with TF IDF NLP method.

Kamal and colleagues utilized transformer-based models for the Hindi language to detect hostility in their study entitled "Hostility detection in the Hindi language. [14] and additionally, they assessed their models' performance on CNN, LSTM, and Attention, Multilingual pre-trained embeddings based on BERT by using translated datasets in the Hindi language.

Detecting fake news in local languages remains a significant challenge due to language barriers and the scarcity of available datasets. Consequently, there have been few studies and research projects in this area. This study addresses this challenge by utilizing the BERT model, along with existing research papers and projects in this field, to effectively detect fake news in Hindi

To collect data, various methods can be employed, including web scraping, which involves extracting information from websites containing both fake and genuine news. Additionally, classifiers like SVM and random forest, or a combination of both, as seen in, can be used to achieve high accuracy, such as with SVM-RF. While Kaggle and other websites may be used to collect datasets, this project collects data by web scraping.

Many classifiers and deep-learning models have been used in diverse projects throughout the world. Nave-Bayes is a common classifier that evaluates the likelihood of specific factors impacting the outcome. Nevertheless, as mentioned in [14], it produces inferior precision, on the other hand, claims that combining BERT with modified Nave-Bayes can result in giving better accuracy and efficient results.

Detecting fake news is a difficult task due to its deceptive presentation, which often goes unchecked by readers. This challenge is amplified when it comes to detecting fake news in local languages, as there are limited resources available compared to English. The scarcity of reliable information in local languages prompted the use of pretrained resources specific to the Hindi language in this study.[15]

In, "A Comprehensive Study on Data-Driven Fake News Detection Methods",[16] stacking approach was used consisting of 5 Machine Learning methods and 3 Deep Learning methods. Machine Learning models used: Log- Reg, Decision Tree, K-Nearest Neighbour, RF and Support Vector Machines. For Deep Learning CNN, LSTM and Principal Component Analysis was used. All of them were connected in the stack using a meta classifier. The Novel Stacking model yielded 99.87% accuracy on the ISOT dataset, while this model yielded 92.84% accuracy on the KD-Nugget dataset. Similarly, the Convolution Neural Networks-Short-Term Memory model with Principal Components Analysis model yielded 97.8% accuracy on the FNC dataset.

3. Proposed work

3.1 Model used for Hindi Language

There are various approaches that can be used for the detection of fake news. Multiple projects have been made using different machine learning and deep learning models. In

this project, the BERT model has been chosen to classify the given dataset into fake news and real news. The BERT model is essentially an open-source machine learning framework for natural language processing (NLP). BERT was created to assist computers in understanding the meaning of ambiguity by using text to establish context. The BERT framework has already been pre-trained with information from Wikipedia in the form of text and can be modified with question-and-answer datasets. BERT is an abbreviation for Bidirectional Encoder Representations from Transformers. It is a deep learning model in which every output data is connected to every input data. Old models like RNN and CNN could read the text from left to right or right to left in the past, but not both at the same time. The BERT model is unique in that sense that it is designed to read and interpret from both sides. This ability prelude to transformers is known as bidirectionality. BERT is a transformer-based model and does not use recurrent connections, but only attention and feed-forward layers. BERT is different from transformers in such a way that BERT only has an encoder while the transformer has both encoder and decoder. Fig.3.1 depicts a neural network of pre-trained model of BERT.

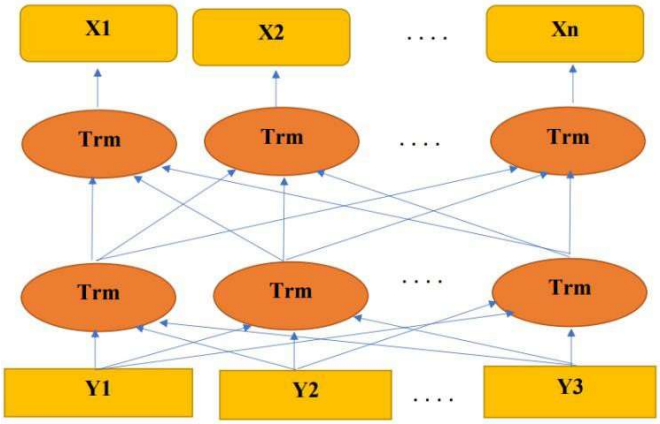

Figure 3.1 Bert Neural Network

We want the model to be able to process patterns in a language, and BERT can do so, allowing us to train the model in an efficient manner and categorize the input dataset into fake and real news. Due to bidirectionality of the BERT model, it produces a higher accuracy than the other deep learning models like convolutional neural networks (CNN) and recurrent neural networks (RNN) when it comes to a much larger and complicated datasets. BERT understands the contextual relationship between different words of the sentence. For BERT, only an encoder of the transformer is enough. The major advantage of BERT over the directional models like RNN, LSTM, CNN is that BERT is non-directional and reads the whole sentence as the input instead of sequential ordering. It has a significant benefit over the approaches discussed above. It can be quickly implemented within the parameters of already available models, providing non-specialist fake news detection systems available over the internet with understanding of the model's decision-making process. The model's classification feature, the tokenizer, and the sample are used in this project. The input for this project is a dataset which will be pre-processed, trained, and tested using TensorFlow, pytorch and transformers. This project uses a 12-layer BERT model with an uncased vocab. Fig. 3.2 depicts the block diagram of how the BERT model works

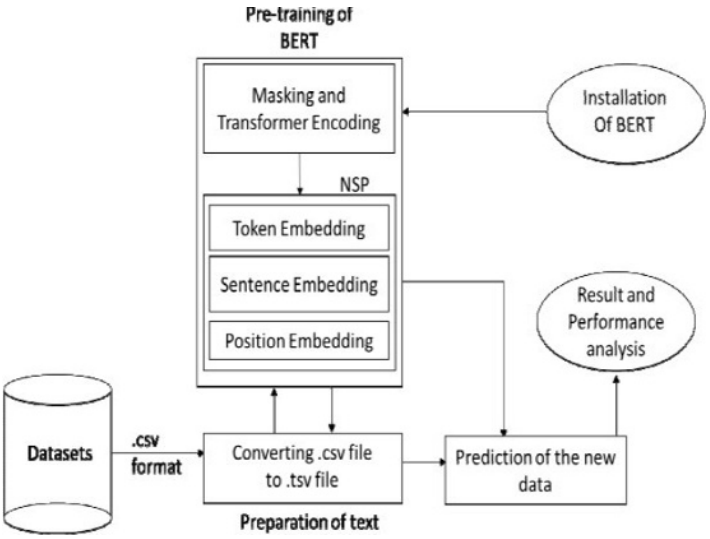

Figure 3.2 Pre-Training of BERT Model

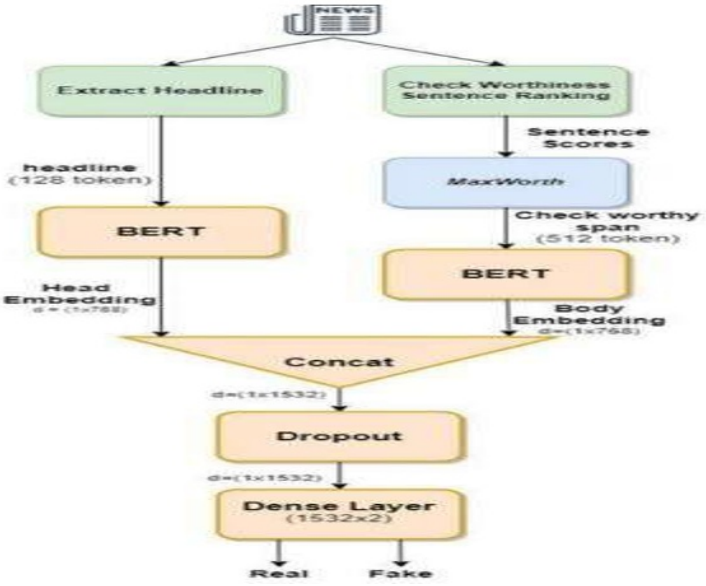

Figure 3.3 Flow Diagram

3.2 Models used for English Language

This work consists of 3 major modules: Module1 - The Training and Model building module, Module 2 - Front- end Development Module and Module 3 - Integration of Modules 1 and 2.

The Data-set used in this work has 23,480 articles on Fake News 21,416 articles on True News, both of which have been divided into columns of Title, Text, Subject and Date for better feature extraction The Word Clouds, as seen in Figures 1 and 2, are

constructed to denote the most common words in both the fake and real news, which will help us in better analysing the data-set and training the models in a more precise manner

Figure 3.4 - Most Common True News Words

Figure 3.5 - Most Common Fake News Words

We have taken inspiration from the referenced papers and implemented 3 different Machine Learning models, each with its own modification, namely - BERT, Dense Layer model along with Word2Vec encoding and SJB - LSTM model.The BERT model is a natural language processing model based on transformers which works as such: every output node is connected to every input node and the weights are dynamically calculated. The model has an architecture which is represented in Figure 3.6. The unique feature in the implemented model is that it has a total of 4 layers, with 3 hidden layers and can process a total of 7,701 parameters.

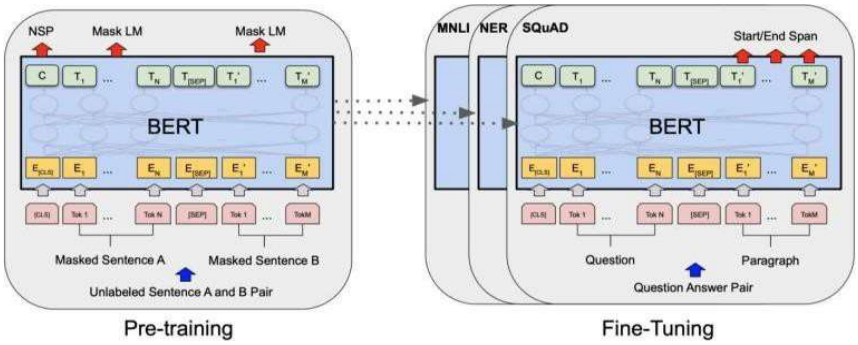

Figure 3.6 - The BERT Architecture

The Dense Layer model makes use of the natural language toolkit to identify stop-words, tokenize and lemmatize the input for the model so that it is converted into a readable format. The Word2Vec used in this works to separate the words that have the true context and from words having a false context which are obtained from the process of negative sampling. The implemented model has a total of 6 hidden layers for processing each feature of the text and is trained over 25 epochs to give it better accuracy. The workings are represented in Figure 3.7

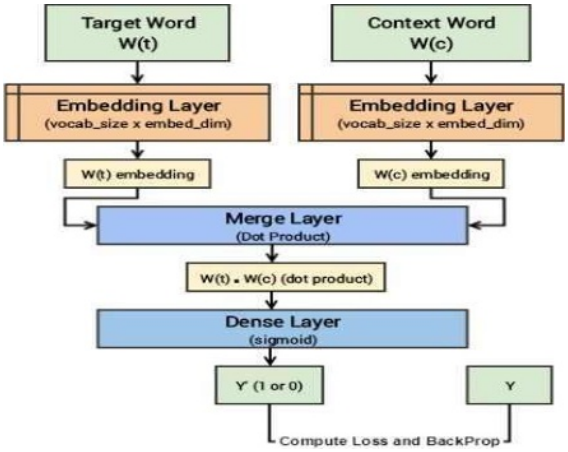

Figure 3.7 word2vec embedding

The LSTM model is very powerful as it can learn ordered dependencies in a sequence. This helps expand the reach of the work to classifying fake news from reality, not only based on words and their correlation, but also on the basis of entire sentences and human phonology. The base Long Short-Term Memory model has been upgraded and hence named after the authors, SJB-LSTM. The implementation of this in the project is such that it has 1 very powerful layer of 50 nodes followed by 3 layers of Dense each consisting of 20, 5 and 5 nodes respectively. This helps it break down the context into micro parts in the very first layer of 50 Long Short-Term Memory nodes and then rebuild it very densely with the following 3 Dense layers. This helps not only in remembering the context of short sentences or phrases but can extend up to complete sentences and paragraphs. The model is trained over 20 epochs and with the hyper parameters: Adam optimizer, learning rate: 10^{-4} , Loss: binary cross entropy loss. All this makes it an accurate and robust model. The workings are demonstrated in Figure 3.8.

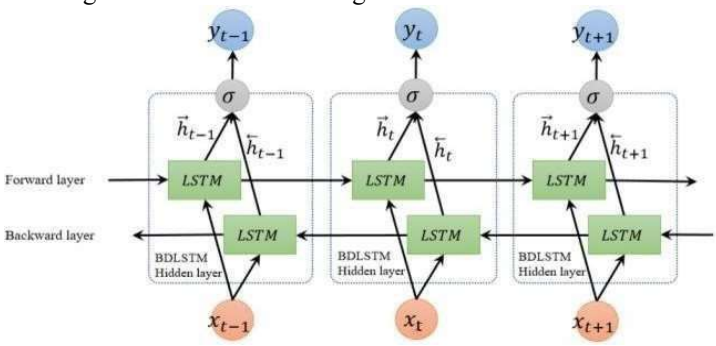

Figure 3.8 LTSM Architecture

The proposed architecture of the entire project takes in the fake and real news dataset. Data pre-processing is performed by the following methods: Normalization, trimming white spaces, removing non-words and punctuations and by Tokenization of news title and text. TF IDF Vectorization (Term frequency inverse document frequency) generates a word bag by raising the frequency with which a term appears in the text as the number of documents where it appears increases. The extracted

features are passed into the classification models which train on it and concurrently improve the hyper-parameters for better accuracy and reduced loss. The same can be seen in Figure 3.9.

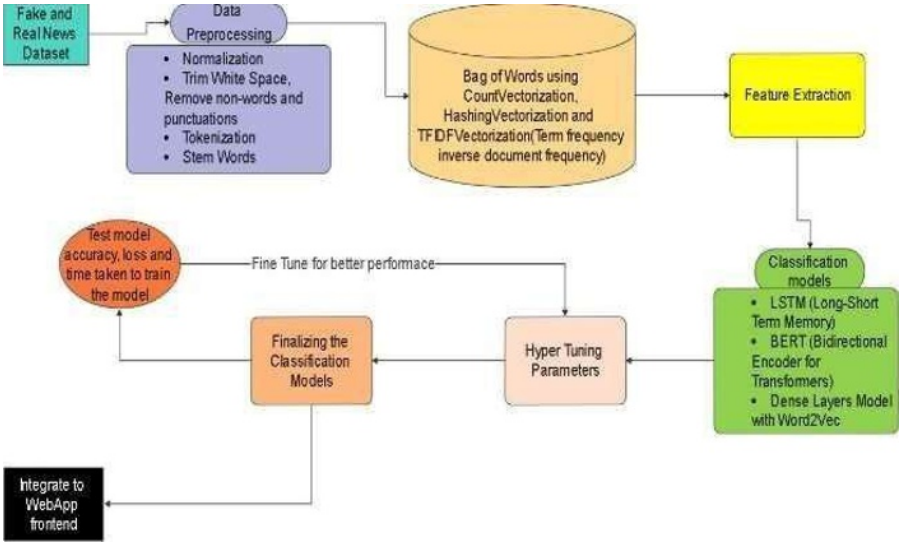

Figure 3.9 Architecture

For the front-end part, flask has been used, which has a basic template pre-built in it and gives ease in locally hosting web pages. The app routes built are hello and submit. Hello, just return the 0index page of the website. Submit route takes in the user input, calls the pickled model, and then runs the provided input through the model. The model then works on the input and returns whether the input is a fake article or a real one alongside the probability of it being fake or real.

4 Results

After all the three models had been trained on the data-set, there were a total of 23,809 parameters. The Bidirectional Encoder for Transformers model was trained over just 5 epochs and took the longest training time of 25 - 30 minutes as it was iterating over 12,319 data items. The Dense Model with Word2Vec was trained over 25 epochs but did not take as much time as the Bidirectional Encoder for transformers as it was taking into consideration individual words and not relation between them that forms the complete sentence. The Long Short Term Memory model was trained over 20 epochs taking the least amount of time. The summation of the models has been provided in the following Table 1 which captures their accuracy and loss percentages with respect to the validation data. Validation data accuracy and loss is important because it reflects how the model will work with real user input

Table 1. -Result Analysis

Model	Accuracy
SJB-LSTM Model	95.43%

BERT Model	91.13%
Dense Model(Word2Vec)	87.47%

As can be seen in Table 1. – result analysis, the most accurate model was the SJB - Long Short-Term Memory model, with an accuracy of 95.43% and a loss of just 11.69%. This model not only remembers the previous work after looking at one word but can also link them by the context that they are used in, thereby making it the most powerful model for the work and for classification in general. Even in testing, it proved to be able to classify most of the fake news presented to it correctly. Following the SJB - LSTM, the BERT model produced an accuracy of 91.13% with a loss of 22.406%. This is indicative of the fact that even after training over many parameters and batch feeding input, the transformers are feasible only for short texts and not complete sentences, titles or paragraphs. It is a good model for other applications like text generation and paraphrasing but not the best for Fake News Detection. The Dense Model with Word2Vec took a lot of Data pre-processing as it analysed and understood each element of the news title input individually. This caused it to not retain the meaning and context of the entire title and evaluate the testing data only based on particular words and stopgaps. The accuracy of this model was only 87.47% with a huge loss of 29.34%

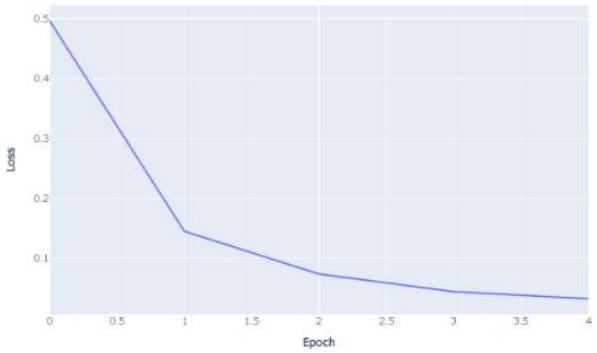

Figure 4.1 Bert model for Hindi language

As it is observed in figure 4.1, the loss of the model is reduced after each epoch and it is almost negligible when epoch=4. The scikit-learn library is used to calculate the accuracy score, precision score, recall score and f1 score.

As can be seen in Figure 4.2, the training and validation loss of the Dense with Word2Vec model reduces over time with training but eventually comes very close implying that it performs poorly to the same extent on both the training and testing data. Similarly, as it is presented in Figure 4.3, training accuracy starts off very low but later catches on but is still less than the accuracy for the training data. This can be a case of over-fitting or it can be that the model requires more training epochs.

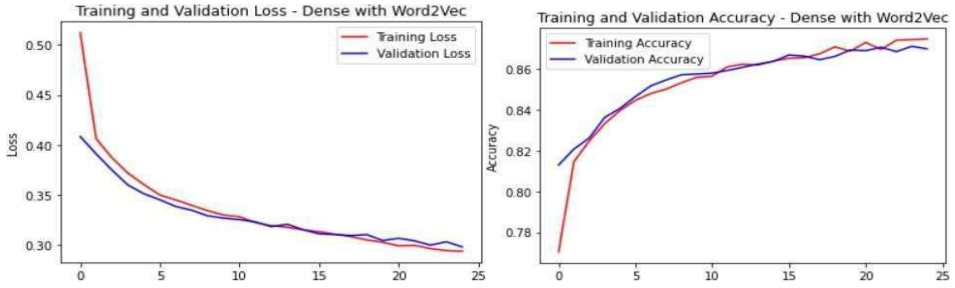

Figure 4.2 Dense Model Loss and accuracy

Figures 4.4 are representative of the training and validation loss and training and validation accuracy respectively, of SJB - Long Short-Term Memory model. As it can be seen the training loss uniformly decreases, but the validation loss is not uniform, this indicates that after a certain limit if the context is not brought together using dense layers, the model will not retain the context of the sentence and news title. The accuracy graph is also showing a similar result, indicative of over-fitting as the training accuracy is starting big and uniformly increases but, at points, the validation accuracy is decreasing. This is because of the loss in context and availability of only particular elements are those specific points. This can be improved by spreading out the Dense layers over a more periodic layer and not cut them short in a strong manner

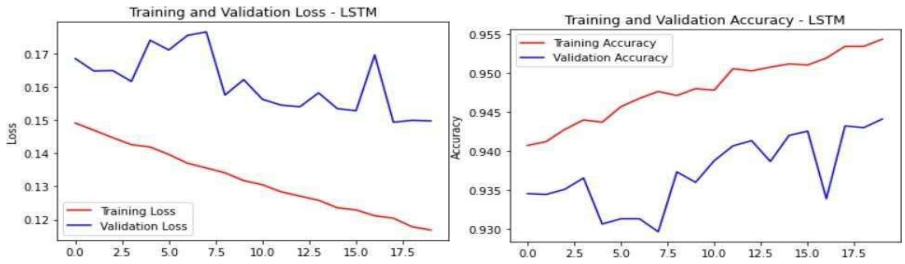

Figure 4.3 LSTM Loss and accuracy

The training loss and accuracy for the Bidirectional Encoder for transformers model, as shown in Figure 12, is an indicator that over a period of training, the loss has gradually decreased but is coming to a constant. Similarly, the accuracy also increases gradually but comes to a constant. This is owing to the smaller number of epochs and large amount of data fed into the model all at once. To improve this, we can increase the number of epochs, the number of layers and reduce how much data is fed in one iteration to the model.

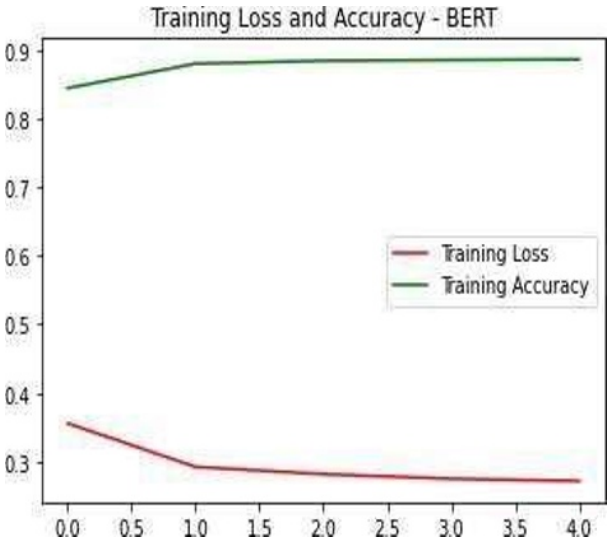

Figure 4.4 BERT Loss and Accuracy

5 Conclusion

We integrated all the 3 models into the front-end using flask. This Fake News Detection WebApp has been successfully built. The user can input the title of the news article for which they want a prediction and choose any one of the machine learning models, or even go ahead with the default if they wish to, and the output is displayed in the form of text. The first output line indicates whether the article is Fake or Real and the following line outputs the probability of it being fake or real. Further improvements can be made by hyper tuning the model parameters to get a better accuracy and reduced loss. A better front-end can be developed with cascade styling sheets and java script. The entire project can be hosted on the world wide web by using a service like ngrok, which we tried to implement, but were not very successful with

References

1. Wajiha shahid et al., *Detecting and mitigating the dissemination of fake news: Challenges and future research opportunities*, In ieee transactions on computational social systems, (2022)
2. K. Singh, a. Aditi, a. K. Jaiswal and a. Shivam, *A comprehensive study on data-driven fake news detection methods*, in ieee transactions on intelligent computing and control systems, (2022)
3. K vishnu vardhan; b.manjula josephine; k.v.s.n. Rama rao, *fake news detection in social media using supervised learning techniques*, international conference on sustainable computing and data communication system, (2022)
4. T. Diehl and s. Lee, *testing the cognitive involvement hypothesis on social media news finds me perceptions, partisanship, and fake news credibility*, Comput. Hum. Behav., (2022)

5. Gupta, h. Li, a. Farnoush, and w. Jiang, *understanding patterns of covid infodemic: A systematic and pragmatic approach to curb fake news*, J. Bus. Res, (2022)
6. Bodaghi and j. Oliveira, *the theater of fake news spreading, who plays which role? A study on-real graphs of spreading on twitter*, Expert syst. Appl, (2022)
7. S. Khan, s. Hakak, n. Deepa, b. Prabadevi, k. Dev, and s. Trelova, "*detecting covid-19-related fake news using feature extraction*, Frontiers public health, (2022)
8. L. Zheng, j. D. Elhai, m. Miao, y. Wang, y. Wang, and y. Gan, *health- related fake news during thecovid-19 pandemic: Perceived trust and information search*, Internet res, (2022)
9. P. Shailendra, R. M, R. S and R. M. R. Guddeti, *Fake News Detection in Hindi Using Embedding Techniques*, Institute of Electrical and Electronics Engineers, Mumbai, India, (2022)
10. D. K. Sharma and S. Garg, *Machine Learning Methods to identify Hindi Fake News within social-media*, 12th International Conference on Computing Communication and Networking Technologies, (2021)
11. M. F. Mridha, a. J. Keya, m. A. Hamid, m. M. Monowar and m. S. Rahman, *a comprehensive review on fake news detection with deep learning*, in Institute of Electrical and Electronics Engineers access papers, (2021)
12. O. Kamal, A. Kumar and T. Vaidhya, *Hostility detection in Hindi leveraging pre-trained language models*, International Workshop on Combating On line Hostile Posts in Regional Languages during Emergency Situation, (2021)
13. S. Singhal, R. R. Shah and P. Kumaraguru, *Factorization of fact-checks for low resource Indian languages*, arXiv preprint, (2021)
14. R. V. Gouda , t. Gauthami , karthik t., *fake news detection using machine learning*, in International Journal of Systematic and Evolutionary Microbiology volume -**3**, issue -**6**, (2020)
15. Thoudam Doren Singh, Abdullah Faiz Ur Rahman Khilji, Divyansha, Apoorva Vikram Singh, Surmila Thokchom, and Sivaji Bandyopadhyay, *Predictive approaches for the UNIX command line: curating and exploiting domain knowledge in semantics deficit data*, (2020)
16. S. Liu, H. Tao and S. Feng, *Text Classification Research Based on Bert Model and Bayesian Network*, Chinese Automation Congress, (2019).