ENHANCED NATURAL LANGUAGE PROCESSING USING TRANSFORMER ARCHITECTURE TO DETECT FAKE NEWS AND RUMORS

Dr. N. Kavitha
M.E, Ph.D, Assistant Professor (SR Grade)
Mepco Schlenk Engineering College, Sivakasi.

S. Lekasri & M. Varsha
Department of Computer Science and Engineering
Mepco Schlenk Engineering College, Sivakasi.

DOI: doi.org/10.34293/shanlax.9789361631474.ch015

Abstract

Nowadays, social networks, online articles, and blogs are the sources for spreading the fake news widely. In social media, all the posts or contents are not original; they may contain fake news or rumors. This fake news can circulate as much as fast and negatively impact the people. This is one of the reasons that we consider that the fake news and rumors detection is more important. So, the fake news detection is necessary to purify the Internet environment. So, we decided to use the transformer architecture model because it is based on neural network architecture and it is designed to handle the stream of data, particularly performing tasks like text generation. The major use of transformer architecture is the self-attention mechanism, which allows the model to focus on different parts of the input sequence when making predictions. The Transformer architecture can be extended to Multi Task Layer (MTL) by modifying the network to handle different tasks. In the MTL model, the transformer architecture learns a shared-attention mechanism across different tasks. Transformer architecture has been widely used in Natural Language Processing (NLP) because the process of NLP can be split into several steps that help to transform raw language data into a form that machines can process and understand.

***Index Terms**: Transformer Architecture, Multi Task Learning, Fake News Detection, Rumors, Natural Language Processing*

I. Introduction

IN the past few years, the usage of social media has increased, and it is one of the reasons for spreading fake news and rumors. Based on Meel and Vishwakarma [17] paper, it says that the fake news implies the false information and rumors are not properly verified pieces of information. The deep learning and machine learning approaches are used for detecting the fake news and rumors that have a difference according to extracted the texts.

The deceptive nature of this information made it challenging for individuals to discern what was real and what was fake. A method was proposed by the Boston Marathon Bombings in 2013 for social media users who struggled to differentiate between rumors and real information. For every tweet that debunked a rumor, there were approximately 44 tweets supporting it. This highlighted the difficulty users faced

in identifying fake news and rumors. Consequently, it raised concerns about the effectiveness of social media in managing such misinformation, calling into question the credibility of these platforms and the web as reliable sources of information.

Some research works have also concentrated on extracting specific features from texts, which were found to assist in the identification of fake news and rumors. For example, Kochkina et al. [4] employed features related to veracity and stance detection for rumor identification, observing a significant improvement in the performance of their classifiers compared to those trained without these features. Similarly, Ajao et al. [6] utilized the sentiment ration obtained from the text through Latent Dirichlet Allocation (LDA) and merged the extracted vector for the sentiment ratio with the feature matrix. It produces enhanced performance because of using the PHEME6 dataset. Additionally, some recent studies indicated a correlation between the authenticity of a text and its associated emotion [2].

It implies the correlation between the authenticity of a text and its inherent emotion, demonstrating that real news and fake news are often represented somewhat differently within the same emotion category. It uses a multitask approach to detect fake news and rumors that automatically learn some features, such as emotions, by using the extracted text. It tests the EFN and rumor detection as related tasks and corrects the effectiveness. The AMT dataset is used for collecting the data from various domains, as well as the Celeb and Gossipcop datasets. In a multitask learning approach, it consists of two tasks, namely the primary task and the auxiliary task. The primary task implies the relationship between the emotions and the validity of a text and also predicts the emotions that extracted from social media. The auxiliary task that is used to label the emotions and to detect the fake news and rumors.

II. Related Works

In the past few years, a lot of progress has been made in addressing the issue of identifying fake news and rumors. Experts and developers have been working on solutions to detect and prevent the spread of misleading information.

N. Colneric et al. [3] used a unison model that utilizes RNN (recurrent neural networks) for the classification of emotion, and it provides a multi-task learning approach that contains three different emotion classification tasks, namely Ekman's, Plutchik's, and POMS's emotion models. By using these models, the authors classify the emotions, and then the authors collect the data from the datasets. By using the deep learning models, such as LSTM and CNN, it will calculate the performance matrix and accuracy. According to the literature survey, the authors used a large dataset of tweets labeled with Ekman's, Plutchik's, and POMS's emotion

classifications. The data that is present in tweets is split into training, validation, and test sets. The unison model provides an accuracy of approximately 70% for POMS's emotion model, and the same accuracy is obtained for the Ekman and Plutchik models.

T. Wolf et al. [12] proposed a transformer model that contains a variety of NLP tasks by fine-tuning pre-trained models. The models such as BERT, GPT, and RoBERTa that are integrated with transformers. It uses transfer learning to handle the classification of language and generation. The dataset that is used in this paper is SST-2 for sentiment analysis, then for text selection, the authors used SWAG and ARC, and the authors used SQuAD for question answering. The overall accuracy for this transformer paper that has been achieved by using SST-2 is 86.7%.

S. Singhal et al. [11] proposed a preprocessing method. In this, the authors first removed their logos from the articles and then dropped samples that contained images. The authors have two submodules: a textual feature extractor and a visual feature extractor. The literature paper consists of two datasets from two different domains: politics and entertainment. The dataset that is used by politics domains is PolitiFact, and the dataset that is used by entertainment domains is Gossipcop. The overall accuracy varies between the two different datasets, such as PolitiFact and Gossipcop. The authors have an accuracy of 0.846 and 0.85%.

T. Saha et al. [14] proposed an approach that includes feature extraction in that it has subdivisions, such as I) textual features; it is used to extract the textual features from the particular tweet, and then II) Emoji Features: In this method, we first extract the emoji from the given tweet by using an emoji that is a Python-based library to extract the pictorial representation of the emojis extracted from the tweet. In this paper, the authors use Modality Encoders, Joint Embedding Networks (JEN), and then ensemble the network. It will integrate the outputs of multiple machine learning techniques, such as Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM), to perform some tasks like sentiment analysis (SA) and emotion recognition (ER), and the overall process is to combine the textual and emoji features using advanced deep learning techniques. The dataset used in this reference paper is the EmoTA dataset, which contains 6810 tweets, and it manually contains tweets, acts, and tags. The performance of the multitask model is better compared to unimodal and single task setups.

O. Ajao et al. [6] proposed a sentiment analysis in which it will predict whether the word is positive, negative or neural by using an Emotional Ratio Score. The authors use two models for extracting emotion scores: it will identify the most relevant word in a sentence, and the models are Latent Semantic Analysis (LSA) and Latent Dirichlet Analysis (LDA). It uses a series of machine

learning algorithms, including logistic regression (LOGIT), support vector machines (SVM), decision trees and random forests. The dataset used in this paper is PHEME-labeled Twitter dataset, and it comprises without images, and it is labeled as rumor or non-rumor. To detect fake news in the news article or social media, the sentiment-aware model is better than the traditional text-based approaches.

G. Verma et al. [15] collected COVID-19 related posts from Twitter Durant the pandemic period, the authors filtering (or) removing the inactive accounts in the Twitter that results in the reduced number of users. The misinformation affects anxiety levels using a large-scale observational study on Twitter. The authors trained a large language model called ULMFiT; it uses a two-stage fine-tuning approach to classify Twitter posts as misinformative or not. The casual inference framework is used for the relationship between misinformation sharing and increased anxiety. The overall performance by using a misinformation classifier achieved a precision of 0.90. It uses a massive social media dataset consistent with 80 million Twitter posts made by 76,985 users, and en filtrant, it contains 30 million posts from 32,290 users.

C. Guo et al. [9], this paper proposes an Emotion-based Fake News Detection framework (EFN) to provide a solution for these two challenges, such as how to capture the signals of publisher emotion and social emotion from news content, while publisher emotions means the emotion of users when posting any information on social media, and then the next challenge is how to exploit publisher and social emotions simultaneously for fake news detection, social emotion means the emotion of users when the information disseminates, the EFN uses two modules, such as the content module and the comment module. The content module learns both semantic and emotional signals from the news content, whereas the comment module captures the user emotions from comments. Both these two modules use bidirectional GRU networks to model the word sequences from both directions of words. It uses a recurrent neural network (RNN) to detect fake news on social media and to capture the hierarchical characteristics of events. The authors construct a dataset on Sina Weibo. This dataset contains 7880 pieces of fake news and 7907 pieces of real news with nearly 160k comments. The mechanisms followed by this paper are that the fake news is collected from the official rumor debunking system of Weibo, and then the real news is gathered from News Verify. The accuracy of the EFN model is 87.2 %, and then the accuracy and F1 score is checked for each module, such as for content and comment modules.

K. Starbird et al. [1], This paper illustrates an exploratory analysis of misinformation spread on Twitter after the Boston Marathon bombing. It will identify the emotion with the help of hashtags, and the data are coded into

three categories, including misinformation, correction, and others, like unrelated or unclear. The patterns of misinformation and correction were compared using temporal volume graphs to study diffusion behavior and crowd responses. The data that are collected by using the Twitter Streaming API with the relevant term (e.g., bomb, explosion).

Kochkina et al. [4], The authors used a multi-task learning approach for performing three tasks, such as i) Stance Classification, ii) Rumor detection together with Veracity Classification, and iii) Learning. The cost function in the multi-task model is a sum of losses from each of the tasks. The preprocessing work of this paper is to first remove the nonalphabetic characters, then convert all words to lowercase and tokenize the text. The authors used a NileTMRG model that uses a bag-of-words representation of the tweet concatenated with the selected features, such as the presence of a URL and a hashtag, and this model requires a stance label for each tweet in the dataset; however, these are not available in the PHEME dataset. The framework uses an LSTM-based architecture with shared layers that is used for performing a number of multiple tasks simultaneously. It compares single-task learning and multitask learning with different task combinations. The datasets that are used are PHEME and RumorEval; these datasets contain Twitter conversation threads associated with different newsworthy events. The accuracy is different for both datasets; for the RumourEval dataset, it has 0,492, and then the PHEME dataset has an accuracy of 49.2%. The multitask learning approach is better than single task models.

J. Devlin et al. [10] proposed a BERT (Bidirectional Encoder Representations from Transformers); this framework involves two steps, such as pre-training and fine-tuning. During pre-training, the model is trained on unlabeled data over different pre-training tasks. In fine tuning, the BERT model is initializing avec les pretrained parameters, and all the parameters are fine-tuned by using the labeled data from the downstream tasks. The BERT model architecture has a multi-layer bidirectional transformer encoder. The authors didn't use a traditional method of BERT models; instead, they used a pre-train BERT using two unsupervised tasks. One is Masked LM, in which the authors simply mask some percentage of the input tokens at random and then predict those masked tokens. The final mask tokens are fed into the output SoftMax layer. The second method is Next Sentence Prediction (NSP), which uses a pre-train for a binarized Next Sentence Prediction task. It uses bidirectional self-attention, allowing to use both the left and right contexts for learning. The SQuAD is used for question answering and sentence classification by using MNLI. The model is pre-trained on Books Corpus, which contains 800 million words, and English Wikipedia, which contains 2.5 billion

words. These are considered as datasets. The accuracy differs from each dataset; SQuAD has 93.2.

S. Vosoughi et al. [2] proposed an investigation of how true and false news stories spread on Twitter from 2006 to 2017. The study analyzed around 126,000 stories tweeted by approximately 3 million users. Using data verified by six independent fact-checking organizations, the research found that false news spreads significantly faster, deeper, and more broadly than true news across all topics, with political news showing the most pronounced effects. The authors suggest that the novelty and emotional impact of false news make it more likely to be shared. Additionally, the study concludes that human behavior, not bots, drives the widespread diffusion of false information.

A. Choudhry et al. [16] propose a deep multitask learning model for fake news detection by leveraging novelty detection, emotion recognition, and sentiment prediction. The authors argue that fake news often contains novel and emotionally charged content, making it more viral. The proposed model integrates these aspects, improving accuracy over existing models on benchmark datasets such as ByteDance, Fake News Challenge (FNC), and Cocid-Stance. The model outperforms single-task frameworks by simultaneously learning these tasks, resulting in state-of-the-art performance in fake news detection.

B. Bhutani et al. [7] presents a method for detecting fake news through sentiment analysis. The authors propose incorporating sentiment as a key feature to improve the accuracy of fake news detection models. The authors use various text preprocessing techniques such as TF-IDF vectorization, cosine similarity, and algorithms like Naïve Bayes and Random Forest. Experiments conducted on datasets like PolitiFact, Kaggle, and Emergent show that integrating sentiment analysis enhances the model's performance, demonstrating higher accuracy compared to models without sentiment features.

T.Shaikh et al. [8] proposed a comprehensive review of various machine learning techniques applied to the detection of fake news. The paper emphasizes the need for deep learning techniques, such as convolutional neural networks (CCNs) and deep autoencoders, to improve the accuracy of fake news detection. The study also outlines the types of fake news, such as user-based, visual-based, and style-based fabrications, and suggests that deep learning can help address evolving challenges in fake news detection on social media. Through the progress in detecting fake news, the fast-changing nature of online misinformation requires continuous development of more advanced models.

R. Kumari et al. [13] proposed a model that jointly performs novelty detection, emotion recognition, sentiment prediction, and fake news detection to

boost overall performance. The hypothesis is that fake news often exhibits novel information, evokes emotional reactions, and influences sentiment, making these factors closely related to misinformation. Developing a multitask learning framework that integrates novelty, emotion, and sentiment analysis. Achieving state-of-the-art results on three benchmark datasets: ByteDance, Fake News Challenge (FNC), and Covid-stance. It improves the accuracy of fake news detection models.

H. Guo et al. [5] proposed a model called HAS-BLSTM for detecting rumors on microblogs like Twitter and Weibo. The model combines hierarchical bi-directional LSTM (Bi-LSTM) for representation learning with social attention mechanisms that integrate social contexts, such as user behavior and post-propagation patterns. By modeling events at multiple events at multiple semantic levels—words, posts, and sub-events—the model effectively identifies crucial components in rumor detection. Experiments demonstrate that HSA-BLSTM outperforms other models in accuracy and excels in early rumor detection, making it valuable for real-time applications.

Based on the combination of Ekman and Plutchik theories, the unison model is created but it gives less accuracy and performance. So, we decided to use the transformer architecture.

III. Proposed Methodology

In this section, the proposed methodology was discussed, and its structure provides a flow of information in step-wise manner. The structure is used to detect fake news from various domains such as politics, entertainment, etc. To detect the fake news and rumors from different domains, the multitask learning approach is used. This part is split into different sections, and those sections elaborates on the methods. Section A elaborates on the used datasets and what that data set contains. Section B, the explanation of preliminary pre-processing steps is involved. Section C, elaborates the text processing and working methods and removing the un-related content from the texts. Section D, elaborates on trained classifiers, which means how we classify the data. Section E elaborates the classification of different methods with different datasets. That also elaborates the working flow of each module.

A. Understand the Data sets

The datasets like PHEME, FakeNews AMT, Celeb, and Gossip-cop were used. The PHEME dataset is used for detecting the rumors. It is a Twitter-based rumor dataset that includes labeled tweets posted on twitter that are categorized as true, false, or unverified across multiple topics. The FakeNews AMT dataset contains the fake news from the various domain like sports, politicians. The Celeb dataset focuses on celebrities related fake news which contains over 200000 images. These images are annotated with 40

features like age, gender, name and facial features. The Gossip-cop dataset contains the entertainment and also celebrities which includes the news are posted from the social media and articles. This dataset is a binary classification. These are labeled as true and false if it's true means 0 and fake means 1, it is significant for fact checking and also for rumor detection.

B. Preliminary Pre-Process

In this module, we ensure the data is clean and ready for analysis. It consists of two methods to clean the data, namely handling missing values and label encoding. Missing values in the dataset can lead to inaccuracies during the analyzing process so we use a method called handling missing values. Without this machine learning method, we can't process the further models of deep learning. This method is used to replace the missing values with mean, median and mode. The removal of the records with missing values are employed to handle these gaps in data. The method that is used to assigning a unique value integer to each category variable. For example, the category variable like true or false, the label encoding is convert these categories into integer such as 0 for false and 1 for true. The main advantage of label encoding is easy to implement and memory efficient.

C. Text Processing

In this module, we used method called NLP, it stands for Natural Language Processing, it is a part of machine learning techniques, it mainly focus on interact with computers and human language in meaningful manner. By using the NLP method, the computer has an ability to analyze, manipulate and generate the human language which includes four methods such as removing punctuation, tokenization, remove stop-words and stemming. The method removing punctuation can provide grammatical context to a sentence which supports our understanding. It doesn't add any values and remove the all special characters, numerical values and also convert all uppercase into lowercase. For example: Where we'll meet tomorrow? It will give a result as where we will meet tomorrow. The 2nd method is tokenization, it process the text and breaking down the text into a smaller parts like words or sentences and this step is essential for structuring the text for future prediction. For example: Welcome to Madurai, it will gives a result as 'welcome', 'to', 'madurai'. The 3rd method is removing stop-words, the words are connecting together in a sequence and it will take only the important words and remove the irrelevant information and also remove the duplicate words. The 4th method is stemming which is used for reduce the unwanted words from the text. This technique used to cuts off the prefix or suffix from the words to get the root of a word. For example: Singing will give an output as sing.

The another method is called as BERT stands for Bidirectional Encoder Representational from Transformer. It is

pretrained transformer-based model and used to process the NLP techniques. By using BERT model, we create an array like structure called sparse matrix of integer datatype which helps to store a string into an integer format.

D. Data splitting

The primary goal of data splitting is to create two different sets from text processing applied dataset which stored separately from that one for training the models and one for testing their performance. It helps ensure that the models can generalize well to new and unseen data. Truth label indicates the truthfulness of the news articles. If the objective of the analysis includes assessing the truthfulness of the content, this label would be used during training and testing. Emotion label categorizes the emotions tone of the article. If the focus is on understanding the emotional sentiment of the articles, this label will be used for training and testing. These two labels can be used based on the model that we used to identify the fake news or real. The models can be trained and tested separately for each label, or a combined approach can be taken where both labels are utilized.

E. Classification

This section is where the actual classification occurs. It includes three different models BERT, LSTM and CNN-LSTM. The BERT (Bidirectional Encoder Representational from Transformer) is a transformer-based model that uses self-attention mechanisms to understand the context of words in a sentence. It processes the entire sequence of words simultaneously rather than one at a time, allowing it to capture richer representations of language. BERT is pre-trained on a large corpus of text using unsupervised learning techniques. It learns to predict missing words in sentences and to determine if two sentences are related. After pre-training, it can be fine-tuned on specific tasks, like fake news detection, using labeled datasets. LSTM (Long-Short Term Memory) is a type of recurrent neural network (RNN) designed to learn sequences of data but in RNN the words are not stored in long period of time so we choose LSTM model. It consists of memory cells that can maintain information over long periods, making it suitable for tasks where the order of words matters.

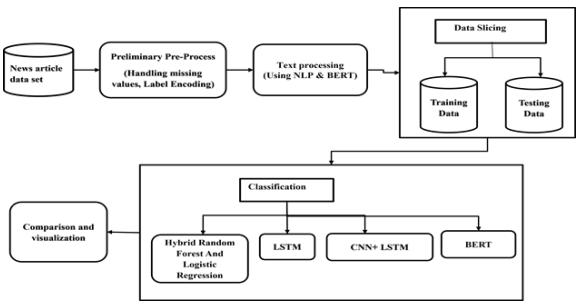

Fig: 1 Architecture diagram

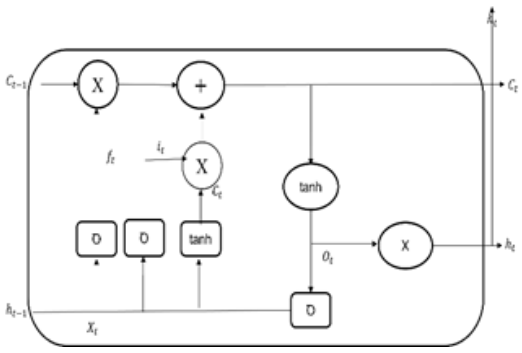

Fig: 2 LSTM architecture

LSTMs use three gates (input gate, forget gate, and output gate) to control the flow of information. This enables the network to remember relevant information while forgetting what is not useful. The LSTM model is trained on sequences of word embeddings derived from the pre-processed text data. It learns to predict the truth label and the emotion label based on the patterns it identifies in the sequences. This hybrid model combines the strengths of CNNs and LSTMs. CNNs are effective at extracting local features from the text (such as n-grams), while LSTMs capture the sequential relationships. The CNN layer processes the input text to extract features, focusing on local patterns. It applies convolutional filters over the text data, which helps in identifying important phrases or structures. After feature extraction, the output from the CNN is fed into the LSTM layer, which learns the temporal dependencies among the extracted features.

F. Comparison and Visualization

In this module we incorporate all the methods and tabulate that methods with accuracy and error rate. Additionally, a Hybrid Random Forest and Logistic Regression model is implemented, leveraging the ensemble strength of Random Forest for decision-making and Logistic Regression for classification, enhancing both the robustness and interpretability of the system. Together, these models form a comprehensive toolkit for achieving highly accurate fake news and rumour detection.

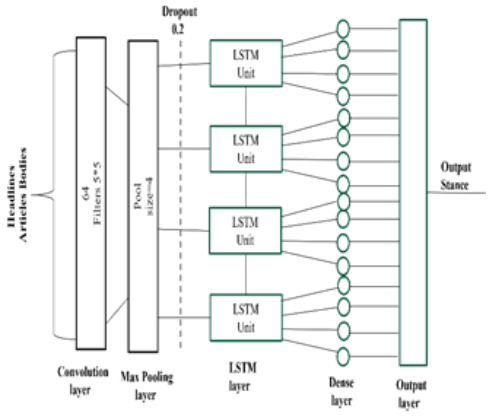

Fig: 3 CNN-LSTM Architecture

IV. Experimental Results

This section describes the experimental analysis and results. Section IV-A describes the dataset that we have used. Section IV-B describes the machine learning model (NLP) and transformer model(BERT). Section IV-C describes the various classification model such as LSTM, CNN+LSTM and Hybrid RF & LR. Section IV-D contains the performance metrics and there results.

A. Datasets

We have used PHEME 9 and FakeNews AMT dataset to evaluate our approach to detect FakeNews and Rumors. The FakeNews AMT dataset contains various domains such as sports, political, education, business and so on while the PHEME 9 dataset contains 9 different events that are posted on twitter it may be True or False, the 9 different events are 2012 US presidential election, Egyptian revolution, Boston bombing, Hurricane sandy, Germanwings crash and so on.

NLP and BERT Model for Text Processing

The text processing has 2 techniques namely NLP (Natural Language

Processing) and BERT (Bidirectional Encoder Representational from Transformer), the NLP is a way for computers to understand and work with human language and breaking down the text into smaller parts such as token or words, phrases. This technique is used to remove the irrelevant words such as 'the','a','is' and consider the meaningful or important words. It minimizes the words to their base or root form. Another technique is BERT, it is a pre-trained machine language model that can be used for various language related tasks and the important information are extracted from the news article by using this model. It has an ability to understand the content

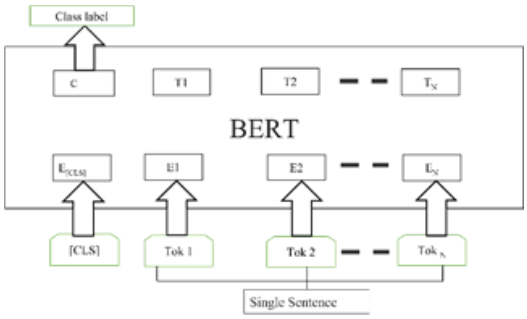

Fig: 4 BERT Architecture

of the words in a sentence that will more helpful to extract more meaningful information. The single sentence is passed into BERT model and it will split up into token such as T1, T2…Tn then it will split up into word embedding such as E1, E2 and En, in this process the words are represents as a numerical vector. The final output has a class label that consists of vectors that will give a occurrence of words.

Input: PHEME.csv, FakeNewsAMT.csv (Fake News Dataset)
Output: Fake or Real Classification, Model Comparison

```
Begin
# Load Datasets
dataframe1 ← load_csv("PHEME.csv")
dataframe2 ← load_csv("FakeNewsAMT.csv")

# Preprocessing
HandleMissingValues(dataframe1)
HandleMissingValues(dataframe2)

# Apply NLP Techniques
dataframe1["text_clean"] ← clean_text(dataframe1["text"])
dataframe2["text_clean"] ← clean_text(dataframe2["text"])

# Convert Text to Numerical Format
X1 ← CountVectorizer(dataframe1["text_clean"])
y1 ← dataframe1["is_rumor"]
X2 ← CountVectorizer(dataframe2["text_clean"])
y2 ← dataframe2["label"]

# Split Data into Train & Test
(X_train1, X_test1, y_train1, y_test1) ← train_test_split(X1, y1, 0.3)
```

```
(X_train2, X_test2, y_train2, y_test2) ← train_test_split(X2, y2, 0.3)

# Train Classifiers
## Hybrid RF + Logistic Regression
model_hybrid1 ← train_hybrid_RF_LR(X_train1, y_train1)
model_hybrid2 ← train_hybrid_RF_LR(X_train2, y_train2)

## BERT Classification
model_bert1 ← train_BERT(X_train1, y_train1)
model_bert2 ← train_BERT(X_train2, y_train2)

## LSTM Model
model_lstm1 ← train_LSTM(X_train1, y_train1)
model_lstm2 ← train_LSTM(X_train2, y_train2)

## CNN + LSTM Model
model_cnn_lstm1 ← train_CNN_LSTM(X_train1, y_train1)
model_cnn_lstm2 ← train_CNN_LSTM(X_train2, y_train2)

# Evaluate Models
acc_hybrid1 ← evaluate_model(model_hybrid1, X_test1, y_test1)
acc_bert1 ← evaluate_model(model_bert1, X_test1, y_test1)
acc_lstm1 ← evaluate_model(model_lstm1, X_test1, y_test1)
acc_cnn_lstm1 ← evaluate_model(model_cnn_lstm1, X_test1, y_test1)
acc_hybrid2 ← evaluate_model(model_hybrid2, X_test2, y_test2)
acc_bert2 ← evaluate_model(model_bert2, X_test2, y_test2)
acc_lstm2 ← evaluate_model(model_lstm2, X_test2, y_test2)
acc_cnn_lstm2 ← evaluate_model(model_cnn_lstm2, X_test2, y_test2)

# Visualization
plot_pie_chart(y1, "PHEME Dataset")
plot_pie_chart(y2, "FakeNewsAMT Dataset")

plot_comparison_bar_chart(
["Hybrid RF+LR", "BERT", "LSTM", "CNN+LSTM"],
[acc_hybrid1, acc_bert1, acc_lstm1, acc_cnn_lstm1],
[acc_hybrid2, acc_bert2, acc_lstm2, acc_cnn_lstm2]
)

# Compare Models Across Datasets
compare_models_across_datasets(
["Hybrid RF+LR", "BERT", "LSTM", "CNN+LSTM"],
[acc_hybrid1, acc_bert1, acc_lstm1, acc_cnn_lstm1],
```

[acc_hybrid2, acc_bert2, acc_lstm2, acc_cnn_lstm2]
)
End

A. *Classification*

It has three types of classification models such as LSTM, CNN+LSTM and Hybrid RF & LR. The 1st model LSTM has three layers such as Embedding layer, LSTM layer and dense layer. The Embedding layer have a input as padded sequence, like tokens and it will give a dense vectors. The LSTM layer have LSTM cell that consists of forget gate, input gate and output gate. Then it will given to the dense layer to given the output in more accurate way. The LSTM model have a accuracy for PHEM9 dataset is 96.9% and Fake-News AMT for 77.35%. The 2nd model is CNN (Convolutional Neural Network) and LSTM, both are combined to form a one model that is used for text filtering and stores the relevant information for over long period and it has a accuracy for PHEME9 dataset is 77.35% and Fake-News AMT dataset is 93.74. The 3rd model is Hybrid RF (Random Forest) & LR (Logistic Regression), the RF is a type of ensemble learning method and it constructs multiple decision trees during training and outputs the mode of their prediction. The LR is a statistical method for binary classification and it will works based on one pr more predictor variable by the binary response. It gives an accuracy for PHEME9 dataset is 96.9% and Fake-News AMT dataset is 99.2%.

B. *Performance Metrics and Results*

To compare all datasets results with modules of LSTM, CNN-LSTM and Hybrid logistic regression with random forest. To calculate these modules using performance metrics, it consists true positive, true negative, false positive and false negative. Using these we can easily find Accuracy, F1-Score, Recall and Precision.

1. Accuracy: Measures the proportion of correctly classified samples (real and fake news) to the total number of samples.
2. Precision: Measures the model's ability to correctly identify only the relevant instances of real or fake news.
3. Recall: Measures how well the model identifies actual real or fake news out of all the possible real or fake news.
4. F1-Score: The harmonic means of precision and recall. It provides a balanced measure that takes both false positives and false negatives into account.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (1)$$

$$Precision = \frac{TP}{TP+FP} \quad (2)$$

$$Recall = \frac{TP}{TP+FN} \quad (3)$$

$$F1 - Score = \frac{2TP}{2TP+FP+FN} \quad (4)$$

References

1. K. Starbird, J. Maddock, M. Orand, P. Achterman, and R. Mason, "Rumors, false flags, and digital vigilantes: Misinformation on Twitter after the 2013 Boston Marathon bombing," in Proc. IConf., 2014, pp. 1–9.
2. S. Vosoughi, D. Roy, and S. Aral, "The spread of true and false newsonline," Science, vol. 359, pp. 1146–1151, May 2018.
3. N. Colneric and J. Demsar, "Emotion recognition on Twitter: Comparative study and training a unison model," IEEE Trans. Affect. Comput., vol. 11, no. 3, pp. 433–446, Jul. 2018.
4. E. Kochkina, M. Liakata, and A. Zubiaga, "All-in-one: Multi-task learning for rumour verification," in Proc. COLING, 2018, pp. 1–12.
5. H. Guo, J. Cao, Y. Zhang, J. Guo, and J. Li," Rumor detection with hierarchical social attention network," in Proc. 27th ACM Int. Conf. Inf. Knowl. Manage., New York, NY, USA, 2018, pp. 943-951.
6. O. Ajao, D. Bhowmik, and S. Zargari, "Sentiment aware fake news detection on online social networks," in Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP), Brighton, U.K., May 2019, pp. 2507–2511.
7. B. Bhutani, N. Rastogi, P. Sehgal, and A. Purwar, "Fake news detection using sentiment analysis," in Proc. 12th Int. Conf. Contemp. Comput. (IC), Aug. 2019, pp. 1-5.
8. T. Saikh, A. De, A. Ekbal, and P. Bhattacharyya, "A deep learning approach for automatic detection of fake news," in Proc, 16th Int. Conf. Natural Lang. Process., Hyderabad, India, Dec. 2019, pp. 230-238.
9. C. Guo, J. Cao, X. Zhang, K. Shu, and M. Yu, "Exploiting emotions for fake news detection on social media," 2019, arXiv:1903.01728.
10. J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Hum. Lang. Technol., Stroudsburg, PA, USA, 2019, pp. 4171–4186.
11. S. Singhal, A. Kabra, M. Sharma, R. R. ShaT. Chakraborty, and P. Kumaraguru, "Spot Fake+: A multimodal framework for fake news detection via transfer learning (student abstract)," in Proc. AAAI Conf. Artif. Intell., vol. 34, no. 10, Apr. 2020, pp. 13915–13916.
12. T. Wolf et al., "Transformers: State-of-the-art natural language processing," in Proc. Conf. Empirical Methods Natural Lang. Process., Syst. Demonstrations, Oct. 2020, pp. 38–45.
13. R. Kumari, N. Ashok, T. Ghosal, and A. Ekbal, "A multitask learning approach for fake news detection: Novelty, emotion, and sentiment lend a helping hand," in Proc. Int. Joint Conf. Neutral Netw. (IJCNN), Jul. 2021, pp. 1-8.

14. T. Saha, A. Upadhyaya, S. Saha, and P. Bhattacharyya, "A multitask multimodal ensemble model for sentiment- and emotion-aided Tweet act classification," IEEE Trans. Computat. Social Syst., vol. 9, no. 2, pp. 508–517, Apr. 2022.
15. G. Verma, A. Bhardwaj, T. Aledavood, M. De Choudhury, and S. Kumar, "Examining the impact of sharing COVID-19 misinformation online on mental health," Sci. Rep., vol. 12, no. 1, p. 8045, May 2022.
16. A. Choudhry, I. Khatri, and M. Jain, "An emotion-based multi-task approach to fake news detection (student abstract)," in Proc. AAAI Conf. Artif. Intell., vol. 36, no. 11, Jun. 2022, pp. 12929-12930. [Online]. Available: https://ojs.aaaai.org/index.php/AAAI/article/view/21601.
17. P. Meel and D. K. Vishwakarma, "Fake news, rumor, information pollution in social media and web: A contemporary survey of stateof-the-arts, challenges and opportunities," Expert Syst. Appl., vol. 153, Sep. 2020, Art. no. 112986.