

Deep Learning for Truthfulness Assessment: Detecting Fake News in Social Media through Deep Learning

by

Mahabubul Alam

20101105

Nabil Faieaz Dipta

20201180

Md Rakibul Islam

24341314

Aunanna Binte Wahid

20201167

Tanmin Alam Rokti

21101051

A thesis submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science

Department of Computer Science and Engineering
Brac University

© 2025. Brac University
All rights reserved.

Declaration

It is hereby declared that

1. The thesis submitted is my/our own original work while completing degree at Brac University.
2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.
3. The thesis does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.
4. We have acknowledged all main sources of help.

Student's Full Name & Signature:

Mahabubul Alam

20101105

Nabil Faieaz Dipta

20201180

Md Rakibul Islam

24341314

Aunanna Binte Wahid

20201167

Tanmin Alam Rokti

21101051

Approval

Deep Learning for Truthfulness Assessment: Detecting Fake News in Social Media through Deep Learning” submitted by

1. Mahabubul Alam (20101105)
2. Nabil Faieaz Dipta (20201180)
3. Md Rakibul Islam (24341314)
4. Aunanna Binte Wahid (20201167)
5. Tanmin Alam Rokti (21101051)

Of Fall, 2024 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science on January 31, 2025.

Examining Committee:

Supervisor:
(Member)

Dr.Muhammad Iqbal Hossain

Associate Professor
Department of Computer Science and Engineering
Brac University

Co-Supervisor:
(Member)

Md Faisal Ahmed

Lecturer
Department of Computer Science and Engineering
Brac University

Thesis Coordinator:
(Thesis Coordinator)

Md. Golam Rabiul Alam, PhD

Professor
Department of Computer Science and Engineering
Brac University

Head of Department:
(Chair)

Sadia Hamid Kazi, PhD

Chairperson and Associate Professor
Department of Computer Science and Engineering
Brac University

Abstract

Misleading information is mostly intended to trick and harm people is known as fake news. It is mostly designed to harm a person or organization's reputation. In this era of information technology, the rapid dissemination of news through social media platforms such as Facebook, Instagram, Twitter and so on has become an integral part of our daily lives. It has also been seen how devastating consequences can be due to the spread of fake news. Rumors can create unimaginable havoc in real life. There are already some existing works that need more efficiency. To detect fake news from social media accurately with most accuracy, we intend to propose a deep learning-based approach. We have utilized two deep learning models LSTM and DNN, a hybrid model of DNN and LSTM, three advanced deep learning architectures such as DistilBERT+LSTM+DNN, DistilBERT+GRU+DNN, BERT+GRU+DNN and our TruthForge model to evaluate our work. Through our work, we aim to provide a robust tool for identifying and differentiating between true and false news, thus advancing the accuracy of news verification.

Keywords: Fake News, Deep Learning, Social media, technology, Rumors, LSTM, DNN, GRU, BERT, Preprocessing, Fine-tuning, Hybrid, Performance, Model

Table of Contents

Declaration	i
Approval	ii
Abstract	iv
Table of Contents	v
List of Figures	vii
List of Tables	1
1 Introduction	2
1.1 Research Problem Statement	3
1.2 Research Objective	4
2 Literature Review	5
3 Dataset Description	12
3.1 Data Collection Method	12
3.2 Data Preprocessing	12
3.3 Data Analysis	14
3.3.1 For CIC Truth Seeker Dataset 2023:	14
3.3.2 For ISOT Fake News Dataset:	24
4 Methodology	29
4.1 Proposed Approach	29
4.2 Model Analysis	31
4.2.1 LSTM	31
4.2.2 DNN	33
4.2.3 Hybrid DNN+LSTM	35
4.2.4 BERT+GRU+DNN	37
4.2.5 DistilBERT+GRU+DNN	38
4.2.6 DistilBERT+LSTM+DNN	40
4.2.7 TruthForge	41
5 Result Analysis	43
5.1 Performance Metrics	43
5.2 Performance Evaluation: Deep Learning Approach	44
5.2.1 LSTM	44

5.2.2	DNN	45
5.3	Performance Evaluation: Hybrid Deep Learning Approach	45
5.3.1	Hybrid LSTM DNN: Without Fine-Tuning	45
5.3.2	Hybrid LSTM DNN: Fine-Tuning	46
5.4	Performance Evaluation: Advanced Deep Learning Approach	46
5.4.1	BERT+GRU+DNN: Without Fine-Tuning	46
5.4.2	BERT+GRU+DNN: Fine-Tuning	47
5.4.3	DistilBERT+GRU+DNN: Without Fine-Tuning	47
5.4.4	DistilBERT+GRU+DNN: Fine-Tuning	47
5.4.5	DistilBERT+LSTM+DNN: Without Fine-Tuning	48
5.4.6	DistilBERT+LSTM+DNN: Fine-Tuning	48
5.4.7	TruthForge: Without Fine-Tuning	49
5.4.8	TruthForge: Fine-Tuning	49
5.5	Explainability Analysis	50
5.6	Best Model	51
6	Conclusion and Future Plan	54

List of Figures

3.1	Data Preprocessing Workflow	14
3.2	Distribution of 5-label majority answer	16
3.3	Percentage representation of the distribution of 5-label majority answer	16
3.4	Distribution of 3-label majority answer	17
3.5	Percentage representation of distribution of 3-label majority answer	17
3.6	Distribution of True and False Statements	18
3.7	True to False percentage ratio of the total number of data collected	19
3.8	Word Cloud of Statements	19
3.9	Correlation Matrix of data	20
3.10	Distribution of author	21
3.11	Arthur VS Target Variables	22
3.12	Distribution of Statement Length	23
3.13	Distribution of Tweet Length	24
3.14	Most Frequent Words in Titles	25
3.15	Distribution of News by Date	25
3.16	Sentiment Distribution of News Articles	26
3.17	Distribution of News Subjects	27
3.18	Distribution of News Subjects	27
3.19	Distribution of Fake and True News	28
3.20	Distribution of Title and Text Lengths	28
4.1	Workflow	31
4.2	LSTM Model	33
4.3	DNN Model	34
4.4	DNN+LSTM Model	35
4.5	BERT+GRU+DNN Model	38
4.6	DistilBERT+GRU+DNN Model	39
4.7	DistilBERT+LSTM+DNN Model	41
4.8	TruthForge Model	42
5.1	LIME: Text	50
5.2	LIME: Statement	50

List of Tables

3.1	Variable Descriptions for Truth Seeker Dataset	15
3.2	Distribution of 5-label majority answer in the Dataset	15
3.3	Distribution of 3-label majority answer in the Dataset	17
3.4	Binary Distribution and Number of True and False Statements	18
3.5	Variable Descriptions for ISOT Dataset	24
5.1	Performance Metrics of LSTM Model	44
5.2	Performance Metrics of DNN Model	45
5.3	Performance Metrics of Hybrid LSTM+DNN Model for Different Max Lengths	45
5.4	Performance Metrics of Hybrid LSTM+DNN Model for Different Max Lengths	46
5.5	Performance Metrics of BERT+GRU+DNN Model for Different Max Lengths	46
5.6	Performance Metrics of BERT+GRU+DNN Model for Different Max Lengths	47
5.7	Performance Metrics of DistilBERT+GRU+DNN Model for Different Max Lengths	47
5.8	Performance Metrics of DistilBERT+GRU+DNN Model for Different Max Lengths	48
5.9	Performance metrics of DistilBERT+LSTM+DNN Model for Different Max Lengths	48
5.10	Performance Metrics of DistilBERT+LSTM+DNN Model for Different Max Lengths	48
5.11	Performance Metrics of TruthForge Model for Different Max Lengths . .	49
5.12	Performance Metrics of TruthForge Model for Different Max Lengths . .	49
5.13	Comparison between Performance of different Models Before and After Fine-tuning	51
5.14	Summary of Research Studies on Fake News Detection: Part 1	52
5.15	Summary of Research Studies on Fake News Detection: Part 2	53

Chapter 1

Introduction

Social media has become a popular news source globally due to its low cost, rapid spread and ease of access. In this modern era, all users can access to browse news portal websites from any internet-connected device. Nowadays the majority of social media users access news online. Even though media is a valuable resource of information, it can also impact society by influencing noteworthy events negatively. The rise in popularity of these platforms has led to the dissemination of fake news over the internet.

Fake news is a widespread phenomenon in Bangladesh. Propagandists manipulated information to spread political and influential messages over the network to convince individuals to distrust society. Fake news can also harm the social and personal relationships of a person, causing political unrest and misunderstanding among communities. These devastating effects of spreading fake and misleading information have led to significant increases in fears of lynching and violence in every nation. The opinions and decisions of the public are greatly impacted by each story, article, or data published on social media platforms.

For instance, false and misleading allegations of rampant Hindu persecution in Bangladesh have been a key propaganda tool against the interim government. According to Sultana [28], it was launched by Indian news outlets and the now-disgraced Awami League after Hasina's downfall in 2024 and they are still ongoing. Moreover, several people made fake accounts during India's 2019 general election season to disseminate fake information on Twitter and Facebook and get attention from the general public [17]. Additionally, the social media platform generates a significant amount of fake and unbelievable content shared by numerous users. It posed a risk to some communities and significantly harmed people through frequent advertisements, online sales and social media advertising. Al-loctt and Gentzkow [3] cited that particularly during the 2016 U.S. presidential election, the problem of fake news on the internet has grown in prominence. Zhang and Ghorbani [20] suggest that false political statements and claims can easily influence voter behavior. Investigation reveals that rumors and fake news spread faster among people than accurate information and have far-reaching consequences.

Many scientific approaches have been put out to automatically stop online rumors by utilizing different techniques. The fake news detection method uses several variables, such as user information, news content and the spread of all social posts, to combat fake news on social media. All computational techniques for fact-checking and fake news

detection rely on prediction models that have been shown to accurately identify news items based on prior research. The findings of this study Deep Learning techniques are managed and treated which will eventually lead to a better situation on social media.

1.1 Research Problem Statement

Since the dawn of human society, there has always been fake news. However, the spread of fake news has been aided by modern technologies and the changes in the global media environment. The existence of fake news on social media platforms has the power to determine societal trust, political division, personal honor legitimacy and balance of the news system.

The problem to be solved in this study is to detect fake news in social media through Deep Learning as it is very important to know whether certain news is valid on the internet or not. Sometimes people build their views on others or situations depending on what they learn on the internet. Consequently, making our decisions can be hampered when we come across incorrect, distorted, or fake information online. Thus deep learning for truthfulness assessment in social media is tremendously necessary in this generation.

- Given that advanced frameworks are the industry leaders in false information identification, a full grasp of modern methods is vital. However, choosing an appropriate feature study that is subject to several news categories and developing a suitable model based on the predicted source is one of the biggest challenges to training such frameworks and models. This study may offer an efficient model development for fake news detection than the existing models.
- Our problem statement considers the importance of considering profiles of the users of social media in rumor detection. For example, the Credible Early Detection (CED) model detects the rumors as soon as possible before they propagate [18]. However, the absence of considerable user profiles in rumor detection could cause detection to be inaccurate in the Credible Early Detection (CED) model. Consequently, one of the goals of this problem statement is to improve this situation.
- Additionally, this problem statement also considers the significance of the accuracy of the false information. As a representation case, the hybrid LSTM-based fake news detection model is utilized to verify the credibility of news articles which includes the speaker profiles [6]. Nevertheless, the hybrid LSTM-based fake news detection model rules out the specific subjects about which the speaker is dishonest. The ultimate goal of this problem statement is to improve the accuracy of fake news identification on social media.

1.2 Research Objective

Traditional fact-checking methods are often ineffective, so advanced methods for determining truthfulness are required. With a particular focus on the identification of fake news on social media, the main objective of this research is to offer thorough attempts to create and assess Deep Learning models for fake news detection. This research seeks to clarify the fundamental ideas, working methods and recent developments in Deep Learning models that support precise identification and categorization of incorrect information and misleading content in the ever-changing and dynamic social media ecosystem. The research aims to provide the readers who have the necessary expertise and skill to effectively understand the challenges presented by the fake news detection process by delving into cutting-edge research and the implementation of different advanced models for fake news detection. The following objectives outline the scope of this study:

- **Robust Fake News Detection Models:**

This study explores different architectures and then will create and implement deep learning models that can accurately detect fake news on social media platforms. This includes experimenting with different architectures such as two deep learning models LSTM and DNN, a hybrid model of DNN and LSTM, three advanced deep learning architectures such as DistilBERT+LSTM+DNN, DistilBERT+GRU+DNN, BERT+GRU+DNN and our TruthForge model to increase detection accuracy. The goal is to understand their effectiveness in classifying fake and real news.

- **Investigate the Impact of Hybrid Deep Learning Models:**

This study aims to ascertain whether advanced hybrid techniques, as opposed to traditional models, can enhance the reliabilities of the detection of fake news by integrating models like DNN+LSTM, DistilBERT+GRU+DNN and our TruthForge model.

- **Analysis the Potential of DL Models for Automated Fake News Detection:**

This study evaluates the efficiency of several models in identifying false information, shedding light on their advantages and limitations. These results will contribute to the academic knowledge of deep learning applications in misinformation detection.

- **Differentiate between the Performance of Various Deep Learning Models:**

This study will measure important performance metrics like accuracy, precision, recall, loss and F1-score to assess each model. The goal of the study is to determine which model offers the optimal balance between computational efficiency and performance, aiming for best accuracy while avoiding overfitting.

Chapter 2

Literature Review

This paper [13] introduces DeClarE, an end-to-end neural network model for detecting fake news, which is compared with the other models of LSTM and CNN on four different datasets. DeClarE takes into account external evidence or counterevidence from the Web as well as linguistic style and the reliability of the underlying source. It also produces reasons so that it becomes easier for users to understand. At first, DeClarE searches for evidence from the Web to prove the claim and returns annotated snippets. These snippets act as proof in the form of data and assertions. For a certain allegation, DeClarE looks for proof from the online data. It takes into consideration the context of the claim and also uses word embedding and bilingual LSTM (biLSTM). Finally, using the attention mechanism, it highlights certain sections of the articles based on how appropriate they are to the claim. DeClarE analyzes the claim by listing all the information from the claimed sources. It has been found that DeClarE outperforms LSTM-text and CNN-text significantly on Snopes and PolitiFact datasets, while on Snopes, the performance of DeClarE falls short of Distant Supervision. However, it excels on the PolitiFact dataset, outperforming all baseline models. On the NewsTrust dataset, DeClarE (full) outperforms baselines by 17 percent in credibility regression. On the SemEval Twitter rumor dataset, DeClarE outperforms top-performing models, demonstrating its effectiveness in credibility classification and confidence score prediction.

This paper [26] used “CIC Truth Seeker Dataset 2023” to determine the effectiveness of detecting questionable text for the Long Short-Term Memory Model. This dataset contains over 180,000 tweets related to 700 real and 700 fake pieces of news. At first, the dataset was preprocessed by using the NLTK library. The Adam optimizer was then utilized to train both models. After training, both models were evaluated using Matplotlib. Later on, under the “Forensics” directory in Google Drive, the dataset was kept and then NLTK was employed. Using the Tokenizer function, the processed tweets were tokenized from TensorFlow’s Keras API. The Feed-Forward Neural Network (FFNN) model was compiled with the Adam optimizer. It was shown that the FFNN model achieved an accuracy of around 96 percent on the validation set, while approximately 95.8 percent accuracy was accomplished by the Long Short-Term Memory Network (LSTM) model.

From the paper [23], the hybrid CNN-RNN model is carried out in Python language. In the model, the Conv1D of the CNN layer processes the input vectors and extracts local features from the texts. This generates the feature maps that act as outputs for the RNN layer of LSTM units. The RNN layer learns the long-term dependencies of the news arti-

cles's local features that determines them as authentic or fake through utilizing the local features that CNN extracted. The models mentioned are evaluated on two publicly available datasets, which are FA-KES and ISOT. The FA-KES dataset has 804 news articles and ISOT has 45000. The datasets were evaluated with other supervised classification methods but the hybrid model stood out among them with the highest accuracy.

from this paper [21], a method proposed by which fake news can be determined by EchoFakeD, an efficient deep neural network. Both the content of the news article and social network's echo chambers are taken into account for detecting false news. In this model, a latent description of both news content, as well as social circumstance, is merged with the tensor using a coupled matrix-tensor factorization method. The model is constructed with various numbers of filters across each dense layer along with dropout. A deep neural network was employed with optimal hyperparameters to organize the content of the news and social circumstances based on facts individually as well as together. The model was validated on two real-world datasets which are BuzzFeed and Politifact. Classification results for Echo-FakeD stood out with the highest accuracy at 92.30 percent which is greater than other classification methods.

From paper [12], the research recognizes the difficulties caused by the escalation of in-authentic information on social media platforms and focuses on the crucial problem of fake news detection. This project attempts to use computational approaches for the efficient identification of fake news, which can impact public opinion due to its increase. The methodology involves a thorough investigation of multiple deep learning and machine learning models, such as Random Forest, Naive Bayes, K Nearest Neighbors, Decision Trees and Convolution Neural Networks(CNN) in conjunction with Long Short-Term (LSTM) networks. Various feature extraction techniques are investigated, including n gram and TF-IDF. The project's dataset is made up of both authentic and fictitious news stories that were gathered from websites like Kaggle. The outcomes present the performances of each model, emphasizing criteria like accuracy, precision, recall and F1-score. The CNN model, for example, demonstrates its effectiveness in differentiating between true and fake news with 98.3 percent accuracy. The precision and recall of the CNN and LSTM combo are higher. In summary, the project highlights the value of using a variety of models and features for Fake News Detection, highlighting the necessity of cutting-edge methods like deep learning. The results provide important new information to the continuing investigation into halting the dissemination of false material on social media platforms.

In this paper [18], the development of fake news detection in the social network model is the main goal of the proposed study. The SPOT methodology specifically addresses the difficulty of confirming the credibility of tweets, events and individuals on Twitter when formulating problems. First, the SPOT method uses natural language (NLP) for text preparation of social media text communications. Then the suggested method extracts pivotal features of the input text. To identify tweets as rumors or non-rumors, the suggested method feeds the GRNN model with the extracted features. The four primary stages of the SPOT method are opinion mining, data transformation, credibility analysis and fake news identification. During pre-processing, the raw tweet data is normalized and cleaned. After that, by using the continuous skip-gram model, the SPOT technique identifies the co-occurring features that are most connected. The SPOT method estimates

a user's and tweet's credibility or trustworthiness by evaluating the credibility on three different levels, including users, Twitter and event levels. The suggested SPOT approach uses summary information to obtain the word annotation with the aid of bi-directional GRNN. The experimental framework uses the test data set to compare the suggested SPOT strategy with the current DUAL approach to illustrate how well it performs. With 3127 samples in the training set, the SPOT method outperforms the DUAL technique in terms of recall rate, achieving 6.28 percent more. Due to its lack of attention to user sentiment, the DUAL strategy only achieves 85 percent of the accuracy value but the SPOT approach achieves a better accuracy score of 99.15 percent. It is noteworthy that the SPOT method achieves exceptional results in the early stages of fake news detection. As a result, the suggested method has shown better performance than the current method as it achieved 97 percent accuracy.

This paper [14] emphasizes the substantial influence that fake news has on social dynamics, politics and the economy. It focuses on applying deep learning techniques to the identification of fake news. To address the difficulties in fake news identification, the paper presents four attention-based neural network models: single attention, dual attention, word-level memory network and character-level memory network. The study makes several contributions: it examines how side information affects the main text; it finds interactions between texts and side information; it suggests memory networks for external information storage; it looks into multiple computations in stacked memory networks; it achieves state-of-the-art accuracy; and it investigates memory networks at the character level. Deep learning models, such as neural networks, squared error and cross-entropy loss functions, Convolution Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) are covered in the literature review. The paper represents Long Short-Term Memory Networks (LSTMs) as a way around the conventional Recurrent Neural Networks' (RNNs') inability to handle long dependencies. To control the notion of memory, LSTM integrates three gates- forget, input and output-into the network's cell. Mechanisms for forgetting irrelevant information, creating new candidate memory, calculating how much data to add to the updated memory and calculating how much cell memory to remove are all included in the architecture. Convolutional Neural Networks (CNNs), linguistic characteristics and hybrid characteristics are among the methods that are covered. Along with an overview of datasets and models utilized in the field, challenges in differentiating between fake news and rumor identification are emphasized. The study finds that the best method for detecting fake news is to use a word-level memory network with attention mechanisms and side information specifically, credit history and party information.

This paper [16] has used some Deep Learning techniques to detect fake news for social media content. For this classifiers like Logistic Regression, Random Forest, Naive Bayes and Deep Neural Network were used. At first, from a dataset from an internet source, information processing has been done based on the deadlines and articles. There was a lot of information so NLP has been used to classify and find out the crucial information. Seven steps have been followed to train the AI. Firstly, to obtain network yield previously processed records have been given input to execute the network forward. Then the values of errors have been calculated. The error signal vectors of both output and hidden layers have been calculated. The weights of the both output layer and hidden layer have been adjusted. Then the process started for the beginning and an experimental setup was created. From the test Logistic Regression (LR) took the least memory and Deep Neural Network

(DNN) took the most. But the DNN did the process early and LR took the highest time. The DNN algorithm required only 400ns to compete. The accuracy rate is LR 74 percent, RF 77 percent, SVM 79 percent, NB 89 percent and DNN 91 percent. From the result, the DNN is more accurate than the other classifiers and also it takes less time though it takes the highest memory space. However, the DNN is more crucial and efficient in detecting fake news.

This study [22] discusses significant studies to address the detection of fake news classification overall. The survey examined techniques for identifying false news that rely on advanced Deep Learning and NLP techniques. Data preprocessing and word embedding using the NLP technology is the first step. Pre-processing of data is used to manage obscure characteristics, binarize attributes, alter discrete attributes and express complicated structures with attributes. Preprocessing data conserves computation space and time. Text pre-processing not only minimizes the effect of artifacts during learning but also prevents every ingestion of noisy data. When Umer [19] experimented with the characteristics that excluded data cleaning or pre-processing procedures and removed extraneous data, the accuracy of their fake news detection model which had an accuracy of just 78 percent rose drastically to 93.0 percent. It has been demonstrated that combining 2-dimensional convolutional layers with pre-trained word embeddings to find patterns in text data is helpful. However, StackedCNN performs poorly when compared to state-of-the-art CNN. Authors have suggested using vanilla RNN uni-directional LSTM-RNN and RNN to forecast false news stories. Particularly effective at processing sequential data and extracting meaningful characteristics from a variety of data sources are RNNs and LSTM more specifically. A deep learning model called BERT has demonstrated state-of-the-art performance among other neural language processing applications. BERT is an advanced transformer-encoded word-embedding model that has been pre-trained. The ability of the BERT approach to recognize and record contextual information in a text or sentence sets it apart. The hybrid of BERT and 1D-CNN is capable of handling large-scale organized and unstructured text. As a result, the combination helps while handling vaguely. This review will help scholars studying the detection of fake news obtain a clearer understanding of current issues and potential solutions as well as potential paths.

This analysis explores fusion approaches for multi-model content such as text and pictures, with the purpose of recognizing fraud news. Here [15] the three subtasks that are covered are false news text detection, false news image detection and false news multi-modal. A multi-modal data repository known as MCG-FNeWS is uploaded by the authors to the Weibo social network that contains publisher profiles, raw news items and photos or videos. Subtask A's primary aim is text content efficiency; Subtask B's primary goal is visual content recognition; and Subtask C's primary goal is information from many modalities. The outcomes of LSTM, GRU, TextCNN and BERT models are compared in Subtask A; LSTM exceeds TextCNN in recall and F1-score and BERT in accuracy and precision. Pre-trained VGG19 and MVNN are compared in Subtask B, where MVNN exceeds VGG19 on all three performance metrics. The atRNN, late and early baselines are compared in Subtask C. Early fusion surpasses late fusion in terms of accuracy, recall and F1tRNA score whereas late fusion is more precise.

This research [27] delves into two hybrid deep learning models. DeepCnnLstm(CNN and LSTM) and DeepCnnBilstm (CNN and BiLSTM) by the FA-KES and ISOT datasets.FA-

KES identified 804 articles regarding the Syrian War as either fake or true. ISOT identified 52.30 percent fraudulent and 47.40 percent authentic of the 44898 articles. Assessment is done using performance metrics such as F1-score, recall, accuracy and precision. DeepCnnLstm scores 63 percent accuracy, 62 percent precision, 98 percent recall and 76 percent F1-score on FA-KES, compared to 68 percent accuracy, 64 percent precision, 88 percent recall and 74 percent F1-score for DeepCnnBilstm. The performance of other deep learning models is also compared. The models on ISOT generally, achieve figures of 99 percent to 100 percent for accuracy, precision and recall; the exception is LSTM alone, which achieves a score of 98 percent across all performance matrices.

This [5] discourse dissects an approach for tweet stance classification and rumor verification. Tweets are classified into supporting, rejecting, questionnaire and commenting categories employing a supervised classifier that uses text features from learned and pre-trained word embeddings. The model makes use of the following parameters: a dropout probability of 0.5 and a word embedding dimension of 128, 128 filters in Convolutional layers. To address the class imbalance, Adam optimization is used along with three filter area sizes (2,3,4), each having two filters and balanced mini-batching. Ascertaining whether or not a rumor is accurate is the objective of rumor verification, which does not involve the use of any external data. According to the findings of the experiments, the CNN model performs much better than baseline classifiers on a wide range of event datasets. Through the use of a variety of filter window widths, the CNN model demonstrates that optimal outcomes are achieved by using a combination of sizes (3,4,5). The CNN model that was recommended comes in second place for rumor verification, with a score of 0.536 and an RMSE of 0.763.

This research study [11] examines the building of classifiers for predicting the validity of news information. The classifiers are constructed using models such as Recurrent Neural Networks (RNNs), Gated Recurrent Unit (GRUs) and Long Short-Term Memory (LSTM) networks. Both of these types of networks are used. The benchmark for evaluation consists of 12,836 brief statements that have been meticulously categorized and organized with the help of the LIAR dataset. The performance of deep learning models, such as CNNs and RNNs, routinely outperforms that of traditional machine learning models, such as Support Vector Machine (SVMs) and Logistic Regression (LR), among CNN and RNN, CNN outperforms RNN.

Using a dataset consisting of 13,00 fake news items from Kaggle and 50,00 actual articles from the Signal Media News dataset, this work [4] outlines the construction of a classifier that can identify false news based only on the content of the articles. The three different kinds of data that were generated were the experimental data, the validation data, and the training data. Several different models were used, including LSTMs, Bidirectional LSTMs, Logistic Regression, a feedforward network, a vanilla RNN, GRUs, a CNN with max pooling, and a CNN with an attention mechanism. These models were utilized to do the job. Additionally, a variety of different models were deployed more than the ones mentioned above. Among these, the CNN with max pooling achieved the highest precision, while GRUs performed best in terms of recall and F1.

A hybrid model of Long Short-Term Memory (LSTM) and 14 layer Bidirectional Long Short-Term Memory (BLSTM) neural networks are used in [24] research to investigate

the identification of false news in Persian texts and tweets. The dataset was expanded to 7,650 healthy tweets and 6,370 false rumors from 650 healthy tweets and 645 rumor tweets for better performance. Preprocessing included cleaning the text, normalizing it, tokenizing it, removing stop words, phrase identification, stemming, lemmatization, handling terms not in the dictionary and splitting the data into training and test sets. We compared the state of the art models with the recommended models. The proposed model achieved the best levels of accuracy (91.08 percent), precision (92.05 percent), recall (91.10 percent) and F1 core (91.57 percent), all while maintaining an error function of 0.2373 throughout the process.

This [8] work makes inquiries about false news identification surface-level linguistic patterns-based methods employing a novel hybrid Convolutional Neural Network that incorporates metadata with text. This study also uses the LIAR dataset. A majority baseline, Logistic Regression (LR), support vector machines (SVM), Convolutional Neural Networks (CNNs) and Bi-directional Long Short-Term Memory (Bi-LSTMs) were the five baseline models that were used to assess performance. With a test accuracy of 0.270, CNNs outperformed SVMs by a substantial margin ($p < 0.0001$). The productiveness of the hybrid model that combined text and metadata varied. Comprehensive metadata integration improves the evaluation of fake news identification which can be proved by a test accuracy of 0.274 was attained by the margin of all metadata elements.

A model named CSI was proposed in [7] work analysis. The model is composed of Deep Neural Networks, which have the ability to select key characteristics, capture temporal relationships in user involvement with articles, and extract information from many domains. By using neural networks, CSI avoids the expense of human feature selection. Overall, they reflect the temporal behavior and textual content using characteristics that are independent of distributional assumption and data context. They developed this technique to address three primary issues with fake text: first, they assess the degree of the correspondence between an article's headline and body; second, they consider the emotional impact that an article has on readers and third, they determine the article's source by examining the URL's structure and the media outlet's credibility. The datasets that were used were from Weibo and Twitter, which are both examples of real-world social media. Across all the datasets being used, the CSI model consistently obtained the greatest accuracy and F-score across both datasets.

This [9] investigation examines deep neural network solutions for geolocation and fake news detection, using graph Convolution Neural Networks (GCNs). Datasets like GeoText, UTGeo2011 and FakeNewsNet are employed for user geolocation and news verification tasks. The GCN model surpasses previous techniques, attaining better accuracies of 0.944 on BuzzFeed and 0.895 on PolitiFact datasets for fake news classification. Additionally, the MENET with S2labels model showed outstanding performance in geological tasks, with small error values and high accuracy on both GeoText and UTGeo2011 datasets.

This discourse dissects false news transmission on social media, important for maintaining democracy. It [10] divides datasets into Type I (Short texts) and Type II (Longer texts). Deep neural networks are employed, combining techniques in information retrieval and NLP. The method concentrates on style-based and truthfulness detection submodules.

Results show accuracy scores of about 72 percent for truthfulness detection submodules. Results show an accuracy of about 72 percent for truthfulness detection, 81 percent for style-based detection and up to 82.4 percent for the combined module.

The purpose of [1] research is to examine the possible application of character-level Convolutional Networks (ConvNets) for the classification of text. Datasets utilized are DBPedia, AG's News, Sogou News, Yelp Review Full, Yelp Review Polarity, Yahoo Answers, Amazon Review Full, Amazon Review Polarity. Character-level ConvNets demonstrated competitive or superior performance compared to traditional models and word-based deep learning models across multiple datasets.

Chapter 3

Dataset Description

3.1 Data Collection Method

For the data collection, the dataset integral to this thesis "CIC Truth Seeker Dataset 2023" which was collected from the Canadian Institute for Cybersecurity, a prominent institute renowned for its rich content in technology and cybersecurity [25] and "ISOT Fake News Dataset" [2]. Both dataset has a diverse range of perspectives and information, crucial for holistic understanding of the fake news domain.

These dataset are organized in an Excel file based on their particular articles. The Politifact's Dataset includes a tweet crawl of both authentic and fraudulent news, which was utilized for both Truth Seeker and Basic ML datasets. Using these ground truth values, we manually generated keywords associated with the news in the issue to crawl for tweets relevant to these topics before final processing. This allowed us to extract over 186,000 tweets concerning 700 bogus news items. ISOT dataset has 45,000 news where both real and fake news exist.

After obtaining this raw tweet data, the researchers employed Amazon Mechanical Turk crowdsourcing to produce a majority response regarding the degree to which the tweet aligns with the Real/Fake news source statement. Subsequently, a majority agreement procedure is utilized to assign validity to the corresponding tweets in a column that classifies them as 3 and 5 labels.

Consequently, one of the biggest ground truth datasets for Twitter's false news identification has ever been produced, the dataset of TruthSeeker. Next, using the metadata of the individuals who wrote the relevant tweet as well as the tweet itself, they created a dataset of features - giving the user the choice to employ both traditional machine learning methods and deep learning modules. Same process was followed for ISOT dataset.

3.2 Data Preprocessing

There are standard preprocessing processes that are helpful regardless of the types of deep learning models we need to utilize for preparing the text input for our algorithms like two deep learning models LSTM and DNN, a hybrid model of DNN and LSTM, three advanced deep learning architectures such as DistilBERT+LSTM+DNN, Distil-

BERT+GRU+DNN, BERT+GRU+DNN, and our TruthForge model.

The steps for pre-processing and cleaning data are as follows:

- **Data Loading and Initial Cleaning:** Firstly, the datasets used for evaluation was loaded from CSV files. This dataset contained various text-based features such as tweets, statements, author details, title, text, etc. The columns with no meaningful names, such as those labeled "Unnamed: 0", are dropped from all datasets to manage and focus on relevant data only.
- **Data Shuffling:** To ensure a fair distribution of data, the dataset was randomized before processing. As data is needed in training and testing sets before splitting, the dataset is shuffled using a random seed. This shuffling is crucial for reducing bias and variance during model training and also ensuring the models do not learn the order of the data but rather the underlying patterns.
- **Text Cleaning and Normalization:** Raw text data often contains unnecessary information that does not contribute to model performance. Various regex patterns are utilized to clean the text data extensively. For example, the removal of usernames such as "@rokti", hashtags, possessive forms and single spaces on purely textual information, elimination of web links like URLs, any unnecessary textual clutter and excessive spaces are condensed to single spaces. Moreover, these cleaning operations are applied to several fields, like tweets, author names, statements and manually entered keyboards. To ensure uniform and suitable data for deep learning models, this process helped standardize the textual content and remove noise.
- **Tokenization and Vectorization:** Utilizing the BertTokenizer from the library of 'HuggingFace', the cleaned text is tokenized. This tokenizer is created for the pre-trained version of the base uncased BERT text model, which fits well with the previously normalized text. Then text is converted into numeral representation called token IDs. These token IDs are mapped to their respective index in BERT's vocabulary. Alongside tokenization, attention masks were generated to help the model distinguish between meaningful tokens and padding elements which is critical for processing sequences of varying lengths effectively in neural networks.
- **Sequence Padding:** After tokenizing, the BERT tokenizer handles sequence padding automatically up to a maximum sequence to ensure all input sequences have the same fixed length for consistency in input size. This uniform is important for batch processing in deep learning models that require uniform input shapes, like LSTM and DNN models.
- **Data Splitting:** The dataset was split into three sets in order to assess the model's performance. They are training, validation, and testing, where the training and validation sets used 80 percent of the data and the remaining 20 percent was reserved for the testing set to train the models. By dividing the dataset, it is possible to train and validate the model on a significant portion of the data and testing it on different, unknown data sets to assess its performance and generalization ability.

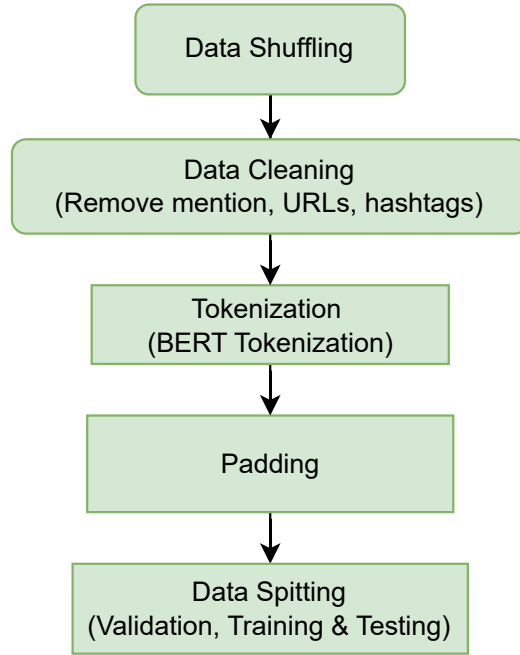


Figure 3.1: Data Preprocessing Workflow

3.3 Data Analysis

3.3.1 For CIC Truth Seeker Dataset 2023:

The data frame initially contained a mixture of fake and real news posted on Twitter with different descriptions. Each column contains the variable's name, type and description, along with the engineering and source information. Table 3.1 below refers to each column for the sake of convenience in the description. It lists the variables in the dataset and their corresponding data types that can be accessed using the provided link. Table 3.1 includes data based on text, numeric and category as variable types.

Variable	Type	Description
author	Categorical	The author of the statement
statement	Categorical	The content or title of the statement
target	Boolean	The ground-truth value of the statement
BinaryNumTarget	Boolean	Binary representation of the target value (1 = True / 0 = False)
manual_keywords	Categorical	Manual categorization, indicating the nature of the tweet

tweet	Text	The actual tweet content related to the associated manual keywords
5_label_majority_answer	Categorical	The majority answer using 5 labels and they are Agree, Mostly Agree, Disagree, Mostly Disagree, and NO MAJORITY which suggests that there was no consensus when a majority answer was generated.
3_label_majority_answer	Categorical	The majority answer using 3 labels and they are Agree, Disagree, and NO MAJORITY which indicates that there was no consensus when a majority answer was generated.

Table 3.1: Variable Descriptions for Truth Seeker Dataset

The survey findings, which reflect the distribution of 5-label majority answers to a certain question or statement, are displayed. According to the poll results, a sizable majority of participants have a favorable opinion: 40.49 percent of respondents agree with the statement or question and 40.09 percent agree with it mostly, for a total of 80.58 percent in favor, which is shown in Table 3.2, Figures 3.2 and 3.3. However, very few respondents had an unfavorable opinion, with only 0.42 percent disagreeing and 2.15 percent mostly disagreeing, for a total of only 2.57 percent against. Furthermore, 16.84 percent of the participants did not conform to any predominant viewpoint, indicating a combination of impartiality, hesitancy, or diverse viewpoints that do not fall into the agree or disagree categories.

Label	Quantity	Percentage
Agree	54,335	40.49%
Mostly agree	53,806	40.09%
Disagree	561	0.42%
Mostly disagree	2,891	2.15%
No majority	22,605	16.84%

Table 3.2: Distribution of 5-label majority answer in the Dataset

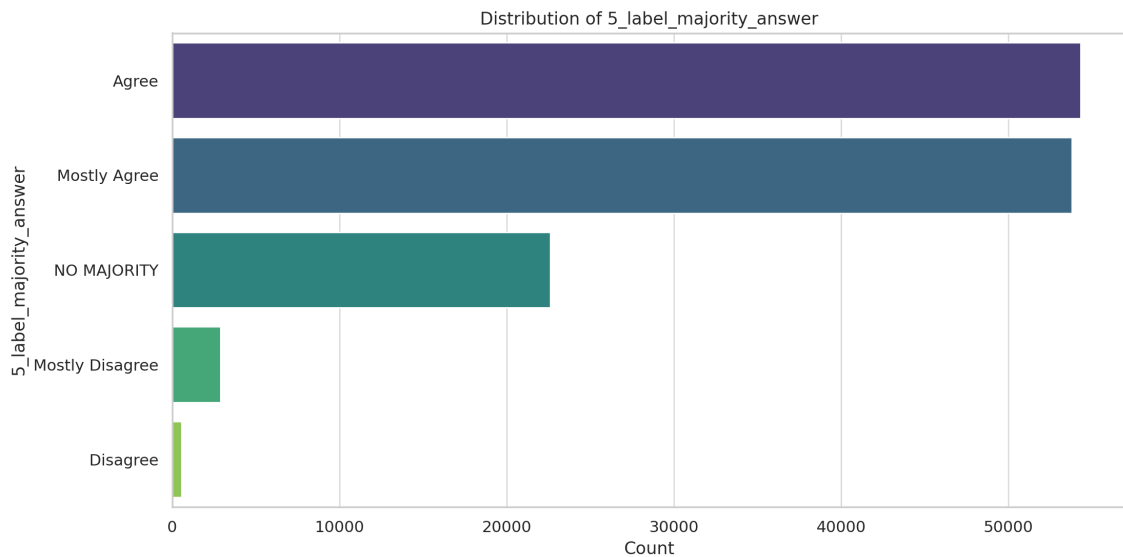


Figure 3.2: Distribution of 5-label majority answer

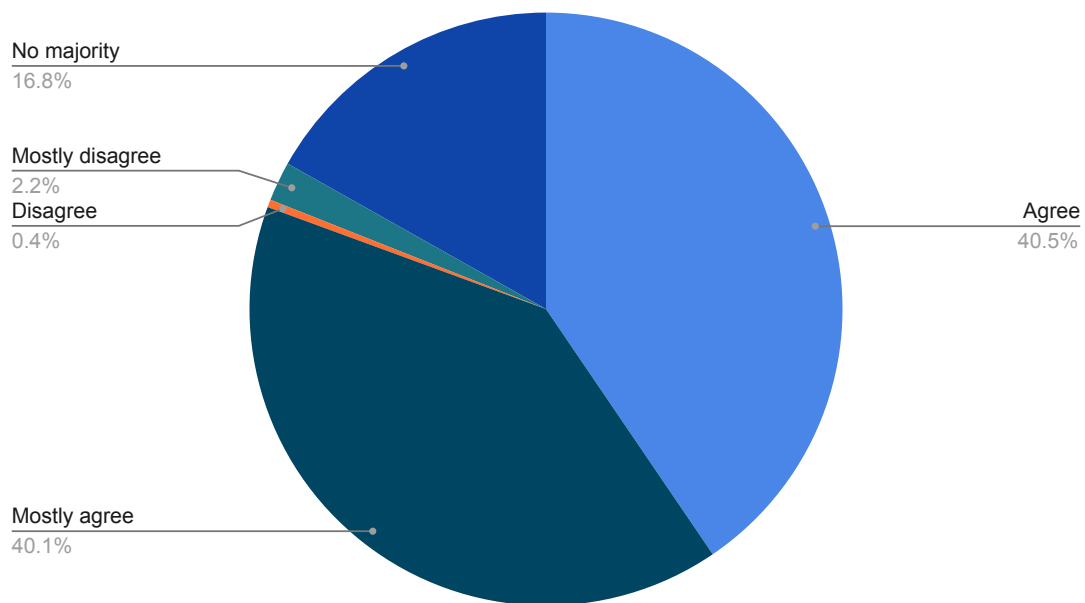


Figure 3.3: Percentage representation of the distribution of 5-label majority answer

The survey findings, which reflect the distribution of 3-label majority answers to a certain statement or question, are displayed. According to the poll results, a sizable majority of participants have a favorable opinion: 95.73 percent of respondents agree with the statement or question. Meanwhile, very few respondents had an unfavorable opinion, with only 4.27 percent disagreeing, which is shown in Table 3.3, Figures 3.4 and 3.5. However, none of the participants conform to any predominant viewpoint, indicating no hesitancy in agreeing or disagreeing with the statement.

Label	Quantity	Percentage
Agree	128,471	95.73%
Disagree	5,727	4.27%
No majority	0	0%

Table 3.3: Distribution of 3-label majority answer in the Dataset

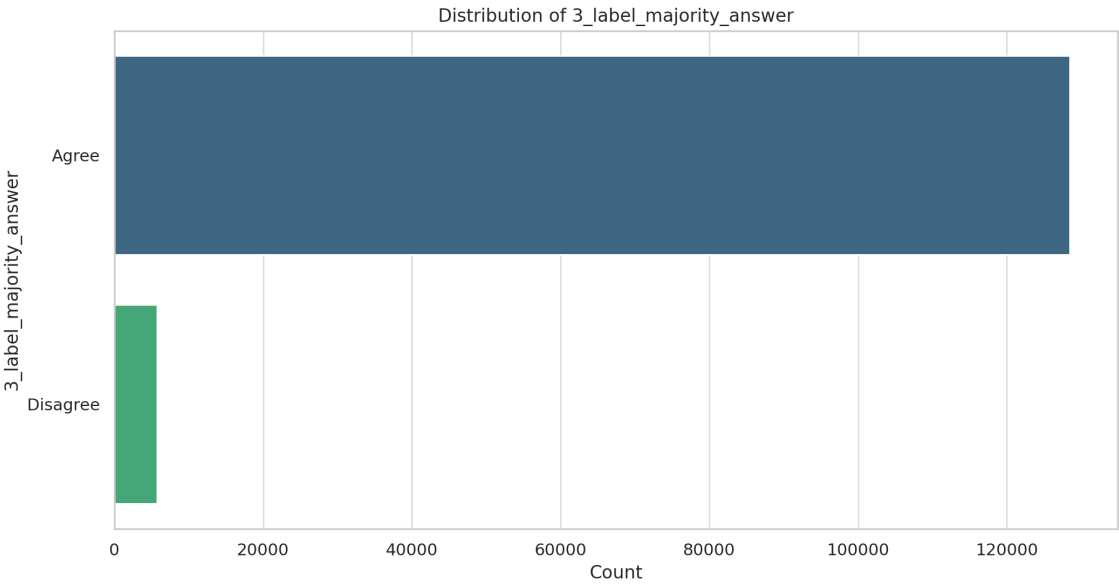


Figure 3.4: Distribution of 3-label majority answer

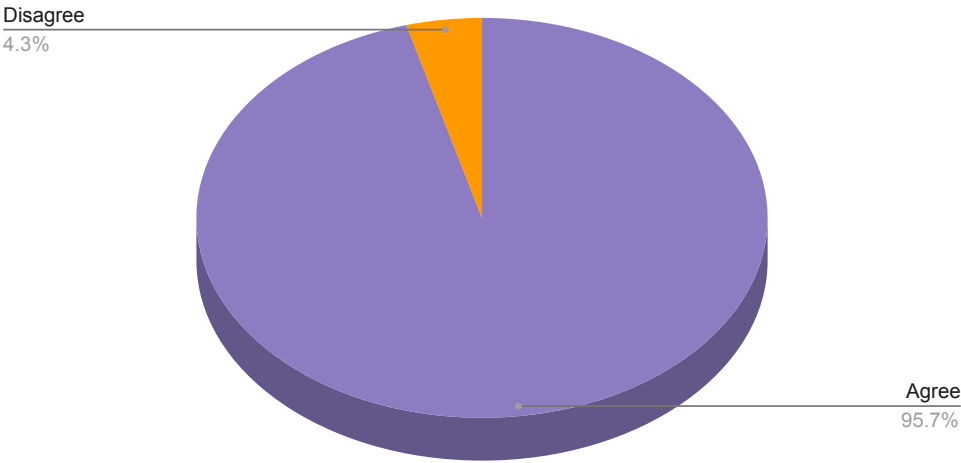


Figure 3.5: Percentage representation of distribution of 3-label majority answer

The significance of labeling statements as true (1) or false (0) lies in their utility for binary classification tasks, enabling the analysis and identification of truthfulness patterns. This binary representation helps in evaluating the accuracy of information and developing models to detect false statements.

This histogram shows the graphical representation of two possible values in the categorization scenario represented by the provided data: 'TRUE' and 'FALSE' and represents 1 and 0 of BinaryNumTarget respectively. 68,930 is the quantity of the true data which is 51.36% of the total data. The quantity of false data is 65,268 which represents 48.64% of the total data which is shown in Table 3.4, Figure 3.6 and 3.7. The dataset is fairly balanced overall, with a slight majority of occurrences classified as 'TRUE'.

Label	Quantity	Percentages
FALSE=0	65,268	48.64%
TRUE=1	68,930	51.36%

Table 3.4: Binary Distribution and Number of True and False Statements

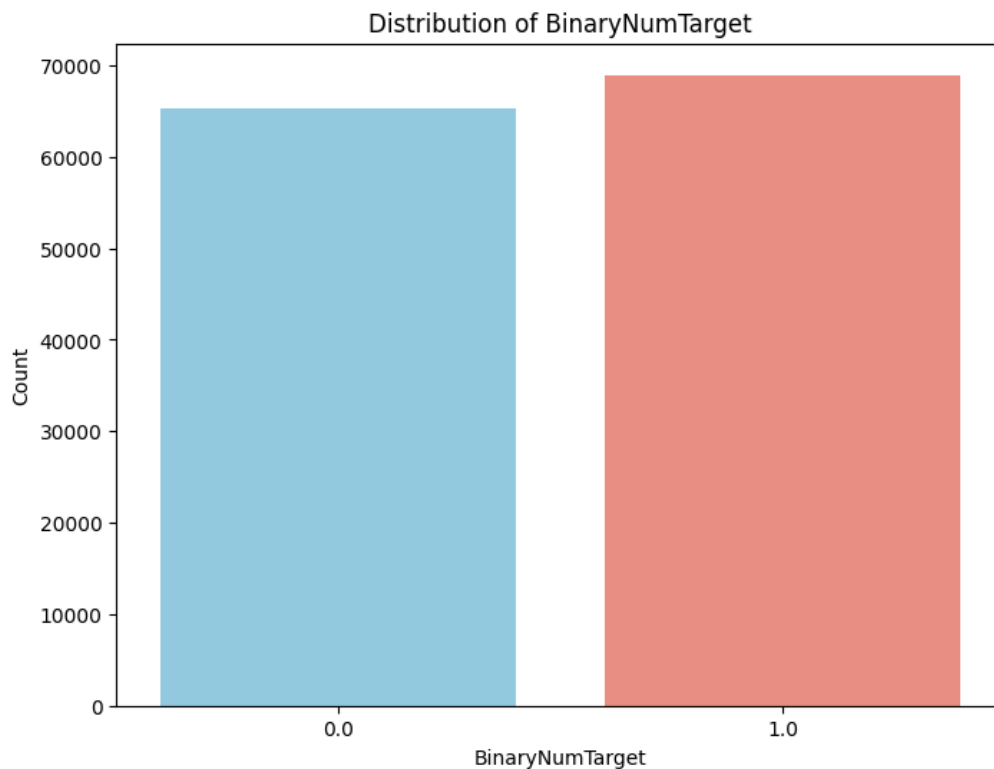


Figure 3.6: Distribution of True and False Statements

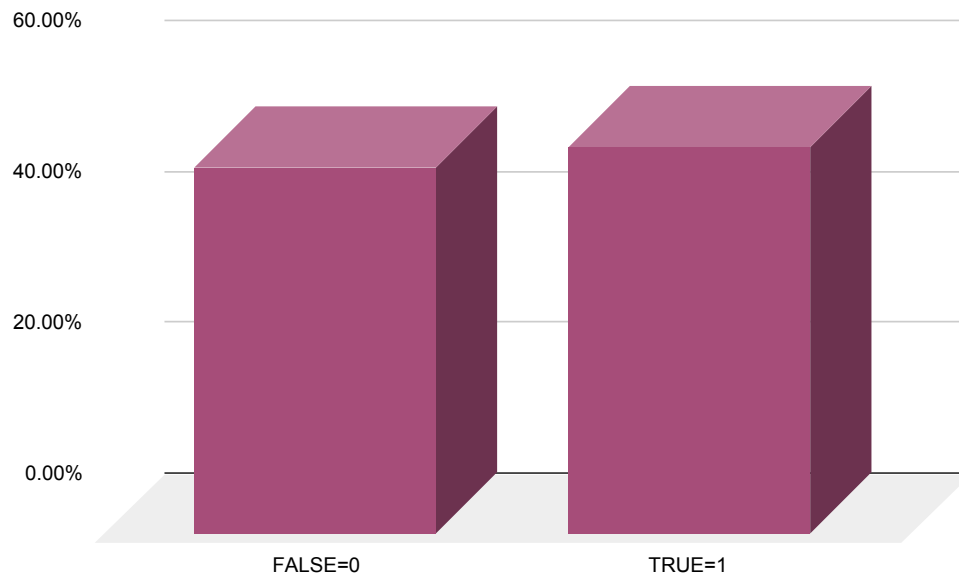


Figure 3.7: True to False percentage ratio of the total number of data collected

The word cloud of statements is significant because it provides insights into significant themes and subjects by visually representing the most frequently used words. Analyzing the word cloud can help you identify the key ideas and the ideas that the dataset stresses most quickly. Figure 3.8's word cloud illustrates conversations that are mostly focused on political personalities, national policies and significant relevant topics like the COVID-19 epidemic and economic policies.

The word cloud provides a visual representation of commonly used phrases in the text, emphasizing important themes that were the subject of fake news, like Joe Biden, Donald Trump, the COVID-19 vaccine, the minimum wages, drug overdoses, etc. Another common word, says, indicating that many statements are quotes or references to what someone has said before. Kamala Harris, Barack Obama, and Nancy Pelosi, are names of political figures, indicating that they are frequently mentioned in the statements.

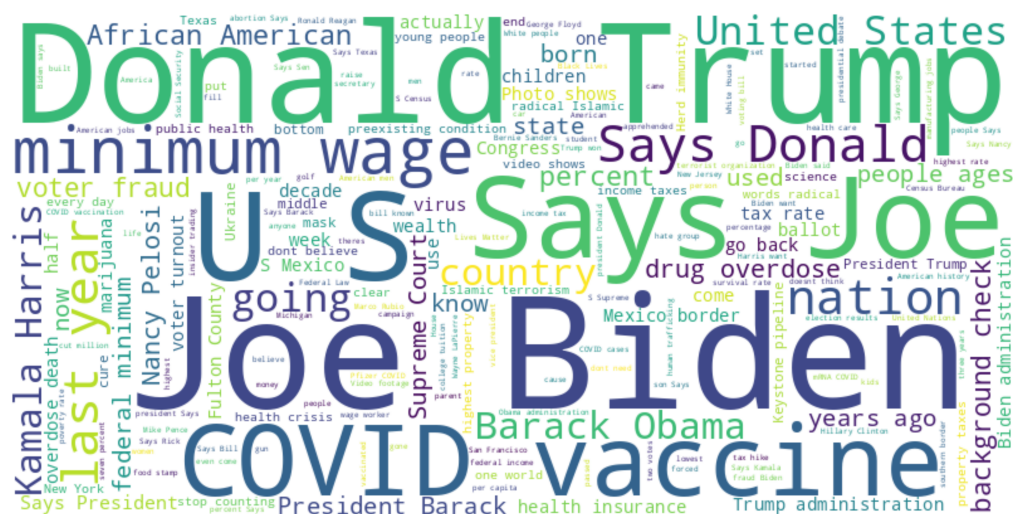


Figure 3.8: Word Cloud of Statements

A correlation matrix is a convenient statistical table that contains relevant information regarding correlation coefficients. It demonstrates the correlation between two variables and provides insights into how variables interact with one another in terms of their effect on the dependent variable. Figure 3.9 enables pattern-level visualization and data-driven decisions and improves the prediction power of our model by learning primarily from relevant attributes.

The following heat map displays a correlation matrix for several different variables. The matrix is an illustration of the correlation coefficients that exist between sets of data, with the intensity of the color representing the degree of correlation as well as the direction in which it exists. For example, red colors represent positive correlations and blue color negative represents correlations. There is a significant positive correlation of 1.0 between ‘target’ and ‘BinaryNumTarget’ and there is a moderate positive correlation of 0.24 between ‘5 label majority answer’ and ‘3 label majority answer’. Both of these correlations are noteworthy and other relationships are either not significant at all or are quite weak.

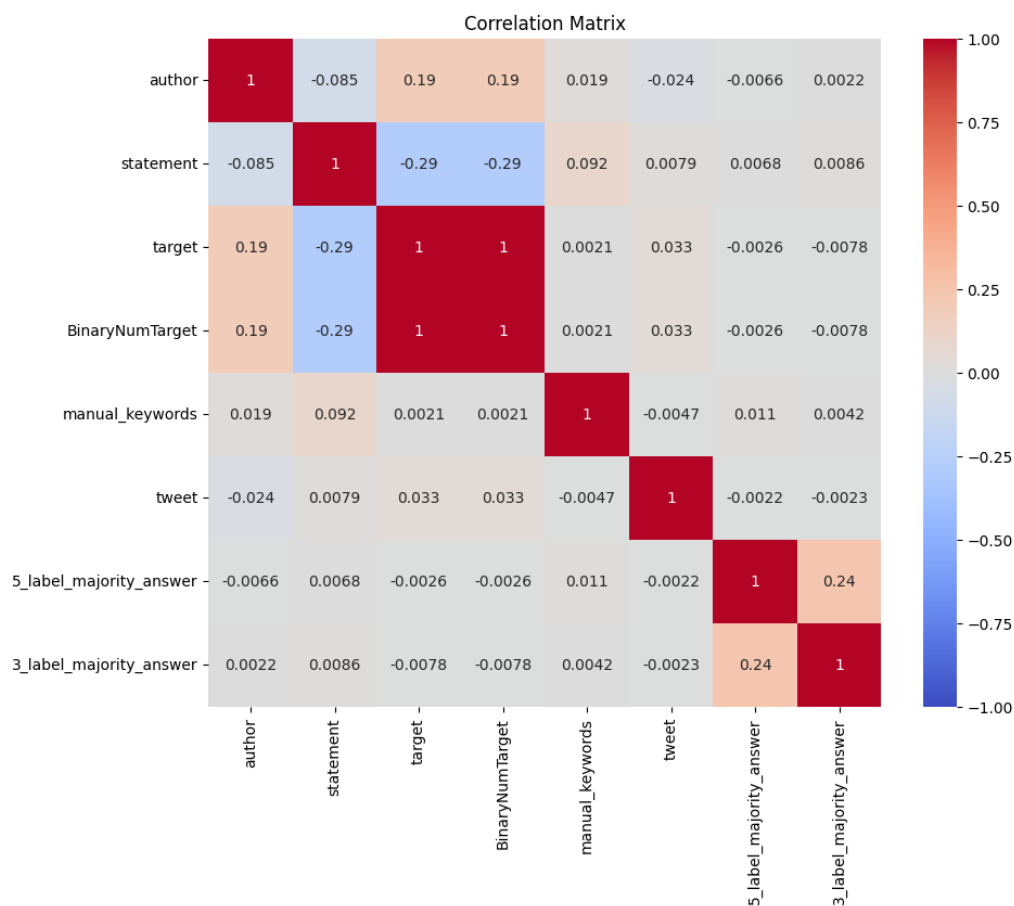


Figure 3.9: Correlation Matrix of data

The distribution of authors is necessary because it shows how different and well-represented the various viewpoints are in the data. By analyzing this distribution, one can ensure a more thorough and balanced grasp of the topic matter by identifying any potential biases or dominant vices. The horizontal chart from Figure 3.10 illustrates the distribution of contributions among the many authors.

The horizontal bar chart from Figure 3.10 demonstrates the distribution of authors according to the number of tweets or statements they have written. Here, Ciara O'Rourke leads the pack in this category with slightly more than 16,000 contributions. Louis Jacobson has made about 14,000 contributions, making him the second highest. Tom Kertscher and Jon Greenberg come next, both with tallies of about 11,000. The remaining writers have contribution counts under 10,000, with Bill McCarthy's (about 7,500) and Katie Sanders' (just over 1,000) contributions falling off over time. There is an obvious visual distinction between the authors who are more and less productive.

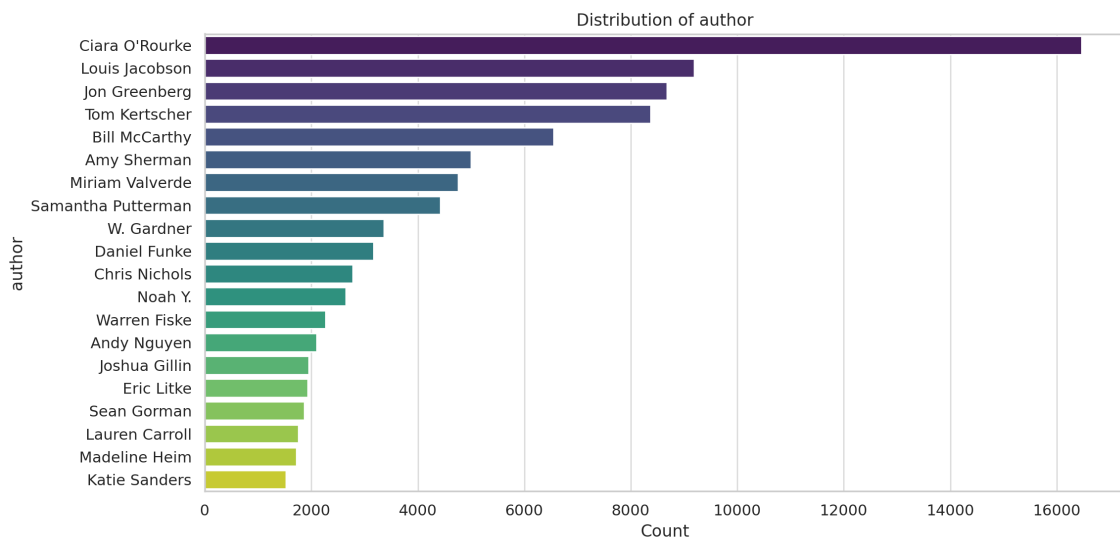


Figure 3.10: Distribution of author

The distribution of authors versus target variables is significant because it can show how the contributions of various authors correspond to particular categories or outcomes. Examining this relationship facilitates the identification of influence patterns, possible biases and the range of viewpoints related to different target variables. This graph from Figure 3.11 illustrates the variation in the number of 'Not Fake' articles written by different authors.

Figure 3.11 demonstrates the distribution of tweets authored by various authors and marked as 'Not Fake' as seen in this bar chart. Here, the number of 'Not Fake' articles is represented by a bar for each author. With more than a thousand 'Not Fake' articles, Madeline Heim holds the record. Louis Jacobson has written about 800 pieces, whereas Amy Sherman has written about 700. Significant numbers are also shown by Stephaine Stapleton, Jill Terreri and Lyle Muller, each of whom has between 600 and 700 publications. With fewer than 100 publications each, a few writers, including John Pullano, Ciara O'Rourke and Bandon Mulder, have extremely low numbers, the counts of other authors vary, usually falling between 200 and 600 articles.

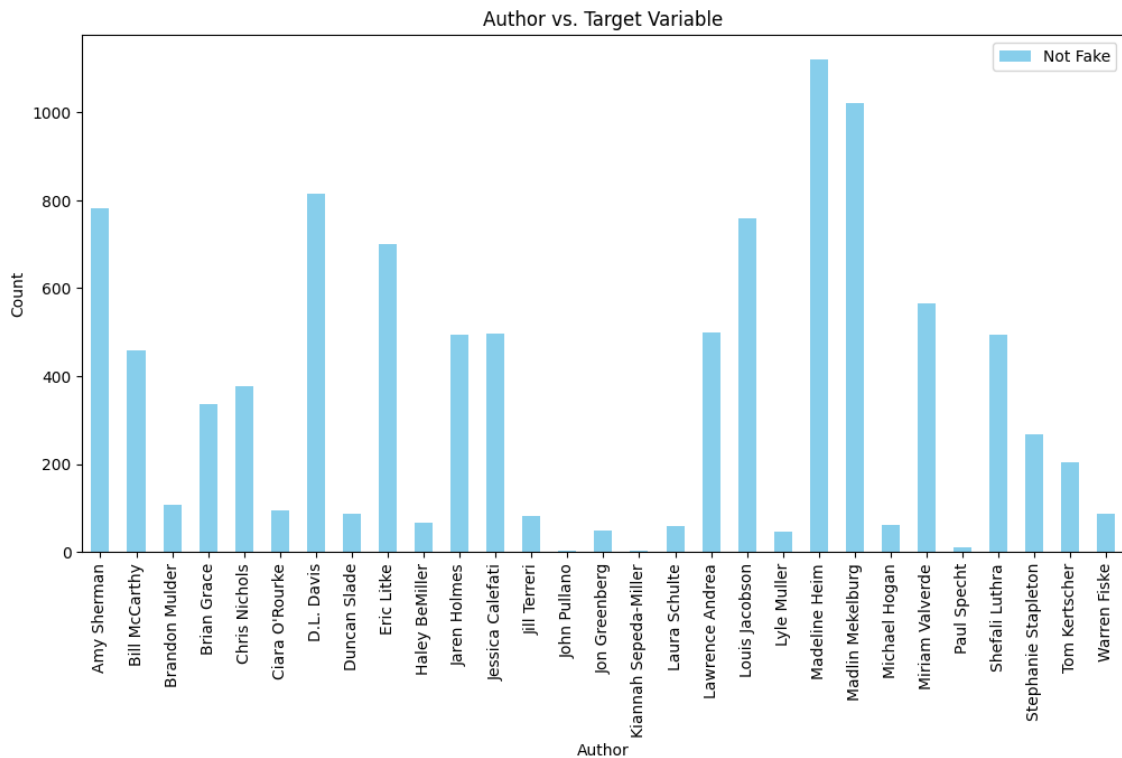


Figure 3.11: Arthur VS Target Variables

The distribution of sentence lengths can provide insights into communication patterns, including repetition of words and complexity, which can affect comprehension and readability. By inspecting this distribution, one can determine whether the statement is communicated well or whether lengthy, possibly unclear remarks are more common. The chart in Figure 3.12 shows the frequency at which utterances of various durations occur.

Now, this histogram from Figure 3.12 illustrates the statement length distribution. With a high frequency of roughly 1500 statements, the most common statement length range in this case is about 75 characters. Additionally, a discernible decline in frequency is observed for statement lengths of approximately 100 characters. At statement lengths of approximately 125 characters, another peak happens at a frequency of just over 800 statements. Additionally, statement lengths longer than 150 characters result in a considerable decrease in frequency, with smaller peaks occurring at around 150 and 200 characters. The histogram generally indicates that statements with a character count between 50 and 125 are more common than those with a character count exceeding 125.

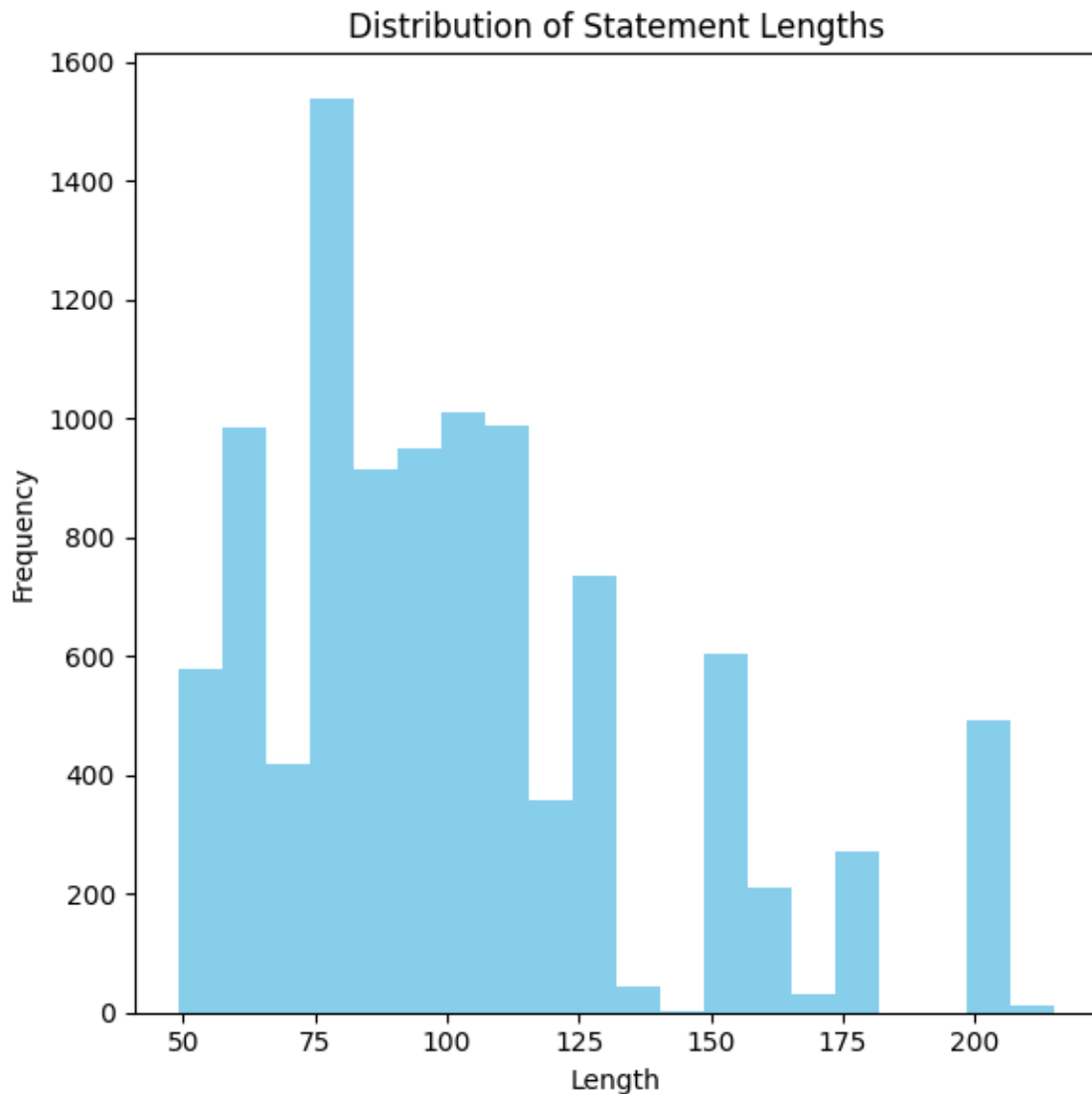


Figure 3.12: Distribution of Statement Length

The distribution of tweet length is important because it can reveal user behavior and content density, which reflects the platform’s (in this dataset scenario Twitter) emphasis on sharing information concisely which is shown in Figure 3.13. Understanding engagement levels can be gained by examining the distribution of tweet lengths; short tweets, which are easier to read, frequently promote conversation more than larger ones.

This histogram in Figure 3.13 depicts the distributions of tweet lengths. The frequency of tweets in this instance peaks at about 3500, culminating at 200 characters. The significance concentration of tweets in the 100-300 character range suggests that most tweets are written in this range. Twitter posts become far less frequent after 300 characters. Additionally, as tweets get longer, they become less frequent; very few tweets are longer than 400 characters and almost none are longer than 1000 characters. The graphic shows the frequency of tweets is less than 300 characters, which suggests that they are often short.

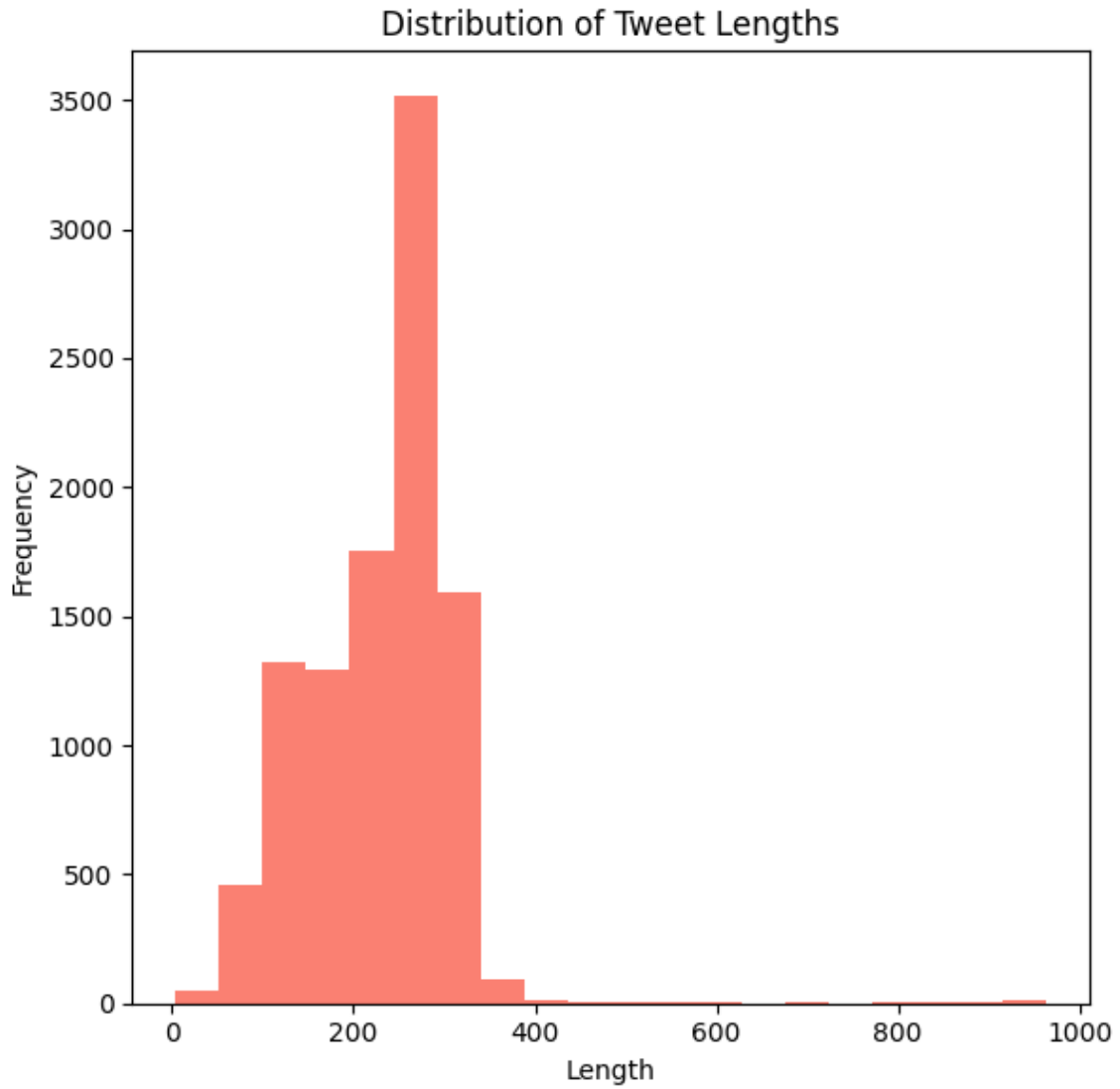


Figure 3.13: Distribution of Tweet Length

3.3.2 For ISOT Fake News Dataset:

For fine-tuning, another dataset was used, 'ISOT Fake News Dataset' [2]. ISOT dataset has 45,000 news where 23503 data for fake news and 21418 data for true news were present. Table 3.5 below lists the variables in the dataset and their corresponding data types and their descriptions.

Variable	Type	Description
title	Text	Title of the news
text	Text	News report
subject	Text	Subject of the news

Table 3.5: Variable Descriptions for ISOT Dataset

This chart in Figure 3.16 shows the sentiment classification of the articles with a vast majority labeled as having positive sentiment, followed by negative and very few neutral.

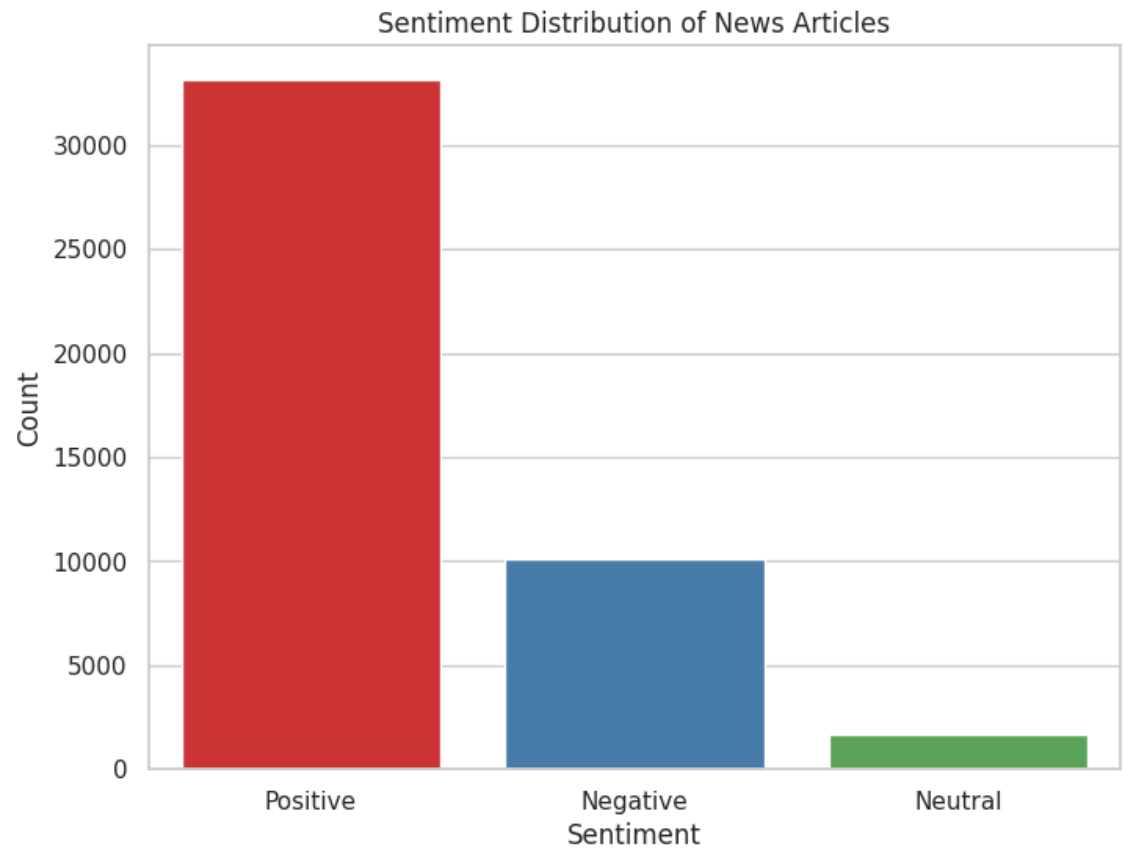


Figure 3.16: Sentiment Distribution of News Articles

A number of articles categorized by different subjects is shown by the bar chart in Figure 3.17. 'PoliticsNews' tops the dataset, depicting a worldwide focus, followed by the considerable representation of topics like 'News', 'Politics' and 'Worldnews'.

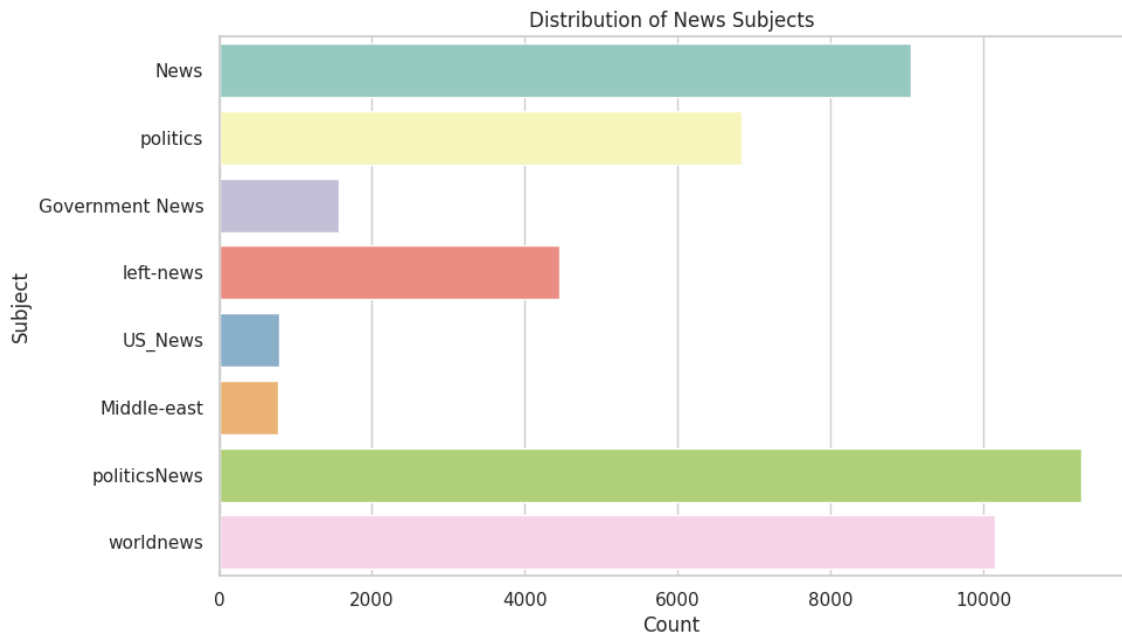


Figure 3.17: Distribution of News Subjects

The box plot diagram in Figure 3.18 shows the difference in the length of texts between fake and true news articles.



Figure 3.18: Distribution of News Subjects

The bar graph in Figure 3.19 shows a nearly equal distribution of fake and true news. False news is slightly higher in number than true news.

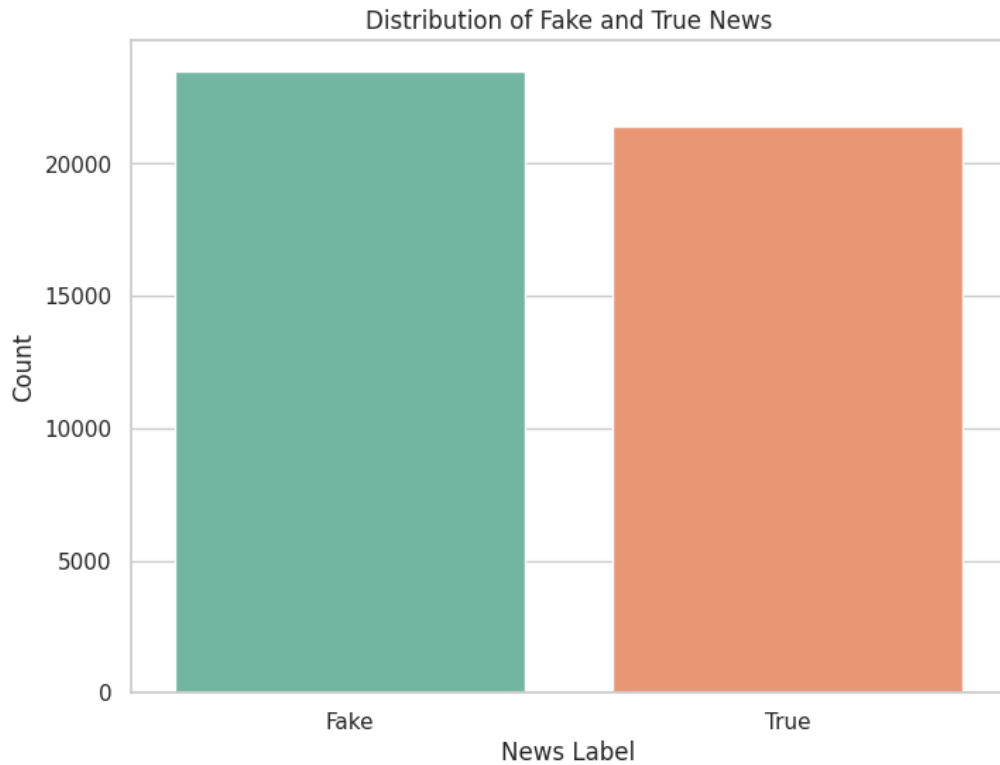


Figure 3.19: Distribution of Fake and True News

The histogram in Figure 3.20 illustrates the distribution of title and text lengths, indicating that most titles are deep (common in news headings), while the length of the article displays a wide distribution that peaks at lower word counts. This distribution helps understand the compactness and detail level expected in news articles.

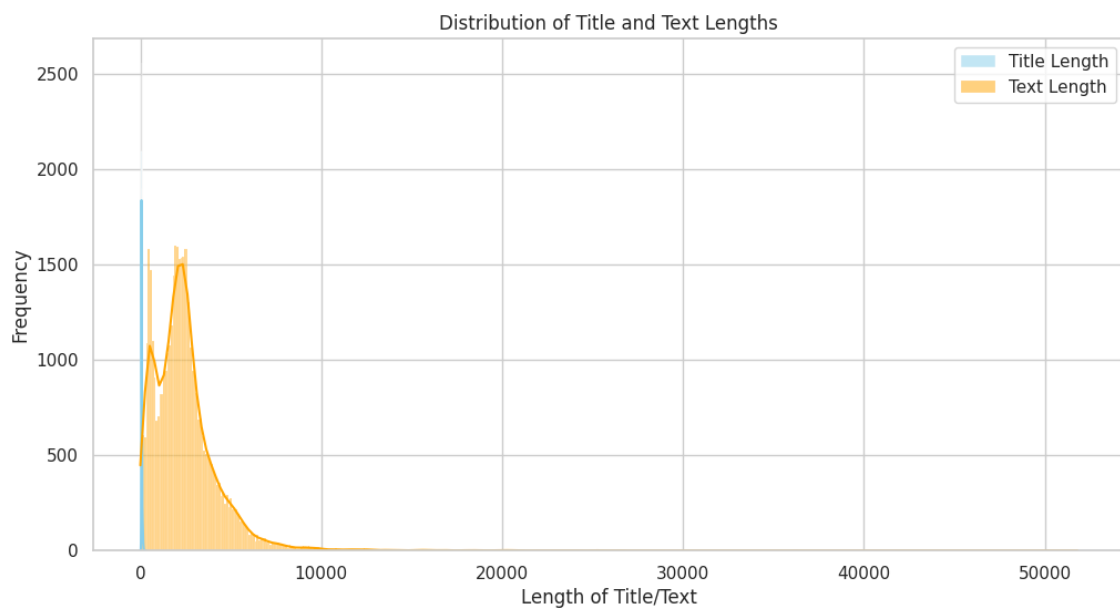


Figure 3.20: Distribution of Title and Text Lengths

Chapter 4

Methodology

4.1 Proposed Approach

The required data for our thesis was collected from an existing dataset called "CIC Truth Seeker Dataset 2023" which was published by the Canadian Institute for Cybersecurity [25] and "ISOT Fake News Dataset" [2]. The implementation of this research was guided by a structured work plan designed to measure multiple deep learning architectures to detect fake news on social networks. The plan involved different integrated steps to evaluate the result.

1. Data Collection and Pre-processing:

After collecting the dataset, several preprocessing steps were taken to ensure consistency and clearance to improve the actual analysis of the data. Tokenization, removal of irrelevant content such as URLs, hashtags, or any other special characters, padding, and data splitting.

2. Model Selection and Implementation:

Different deep learning architectures such as two deep learning models LSTM and DNN, a hybrid model of DNN and LSTM, three advanced deep learning architectures such as DistilBERT+LSTM+DNN, DistilBERT+GRU+DNN, BERT+GRU+DNN and our Truth-Forge model. These models were selected for the experiments and used to grasp the contextual understanding provided by transformer-based models. GRU and LSTM layers will be utilized to understand temporal dependencies and sequential information, while the DNN layers will concentrate on classification tasks.

3. Fine-Tuning:

To adjust to the distinct features of social media text, pre-trained language models such as DistilBERT and BERT were fine-tuned on the domain-specific dataset. To maximize model performance, fine-tuning concentrated on hyperparameter tuning to optimize sequence length, learning rate using the Adam optimizer or other optimizer.

4. Training and Validation:

The training, validation and test sets were split from the data set. To determine the model's robustness, 80 percent of dataset were used on training and validation using cross validation techniques. Accuracy and loss are used as performance metrics which were evaluated on the validation set. It is necessary to identify the best performing configurations.

5. Testing and Evaluation:

The test data set evaluated the accuracy, recall, precision, and F1 score of each model in identifying fake news and was trained on the remaining 20% of the data set. To identify the best architecture for the task, the performances of the models were compared. A modified analysis will be performed to identify the strengths and weaknesses of each model, highlighting the most effective approach.

6. Result analysis:

Now we need to interpret the results into meaningful conclusions by discussing the implications of the findings. Important findings from the study were examined to determine the advantages and disadvantages of identifying fake news in various settings.

7. Planning Future Enhancements:

Some improvements were determined by the result analysis. Nevertheless, expanding the dataset and exploring more model architectures will enhance accuracy. Then, we can implicate the findings for real-world applications, for example, proposing future research directions and social media moderation with deployment to further improve fake news detection.

8. Documentation and Reporting:

The models, outputs and conclusions drawn from the research were all highlighted in the thorough documentation of the findings. This thesis seeks to make substantial contributions to detecting fake news on social media using deep learning models.

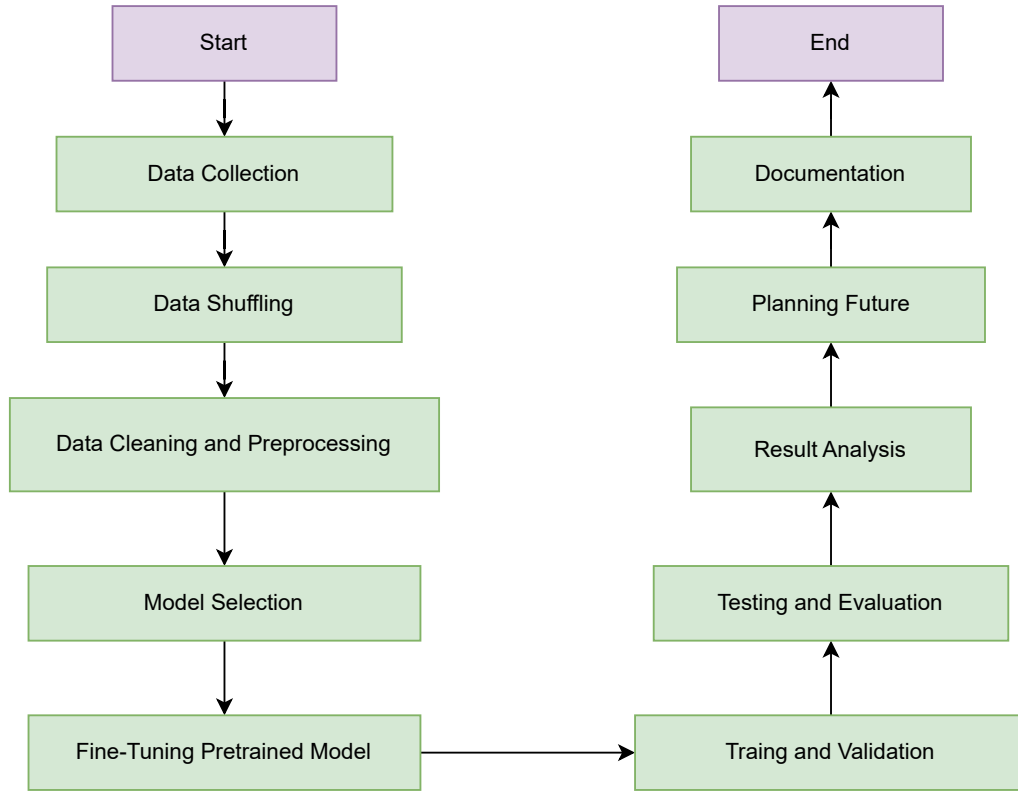


Figure 4.1: Workflow

4.2 Model Analysis

4.2.1 LSTM

Long Short-Term Memory (LSTM) is utilized to process input sequences of tokenized and padded text data with a vocabulary length determined by the tokenizer. The embedding layer transforms each word into dense vector representations to capture semantic meanings. The LSTM layer with 64 units processes these sequences, taking into account long-term dependencies. It is trained by the the Adam Optimizer and binary cross-entropy loss function which produces binary classification predictions through a dense layer with a sigmoid activation function.

Initially, the dataset is thoroughly cleaned by removing rows with null values in important columns such as ‘statement’ and ‘BinaryNumTarget’. This phase makes sure that the dataset has no instances of defective records, which might alter the model’s learning process. To ease the model training, the ‘BinaryNumTarget’ column corresponding to the labels (real or fake) is subsequently modified to a numeric form. During this conversion process, any non-numeric values are forced into NaN (Not a Number) and then removed. This comprehensive data-cleaning procedure ensures that the model generates a consistent and clean data set.

Tokenization and padding techniques have been employed for text preprocessing after data cleansing. With each integer representing a distinct word in the vocabulary, tokenization converts text into sequences of integers. Then we need to build a vocabulary by mapping words to unique integers and turning text into the word index. As it is critical for LSTM models that require fixed-length input sequences, padding sequencing is essential to make certain that all input sequences have the same length after tokenization. Usually, padding is applied to either the longest length seen in the dataset or a pre-specified length and in the LSTM model, it is 100. In this phase, the text data turns into an organized structure which is appropriate for the LSTM model.

Because of the LSTM model's capacity to preserve long-term dependencies, the model for identifying fake news is built using an LSTM network that is useful for sequential data such as text statements. Starting with an embedding layer, the model creates dense vectors of predetermined size which is 100 in the LSTM model from the integer sequences that represent words. Next, the LSTM layer processes the vectors that identify patterns within the sequences and temporal dependencies. To prevent overfitting, the LSTM is set up with dropout and recurrent dropout, which involve randomly setting a fraction of input units to zero while training.

A dense layer with a sigmoid activation function is the last of this model. It produces a probability score indicating the possibility of the news being fraudulent. To make it appropriate for binary classification problems, the Adam optimizer and binary cross-entropy loss are used to compile this architecture. The model is developed on the training set and estimated on a separate validation set to assess its performance where the performance of the model is estimated utilizing metrics like accuracy, precision and recall and modified parameters as needed during training. In the end, the test set determines its generalization proficiencies, which provides information about its usefulness in real-world circumstances.

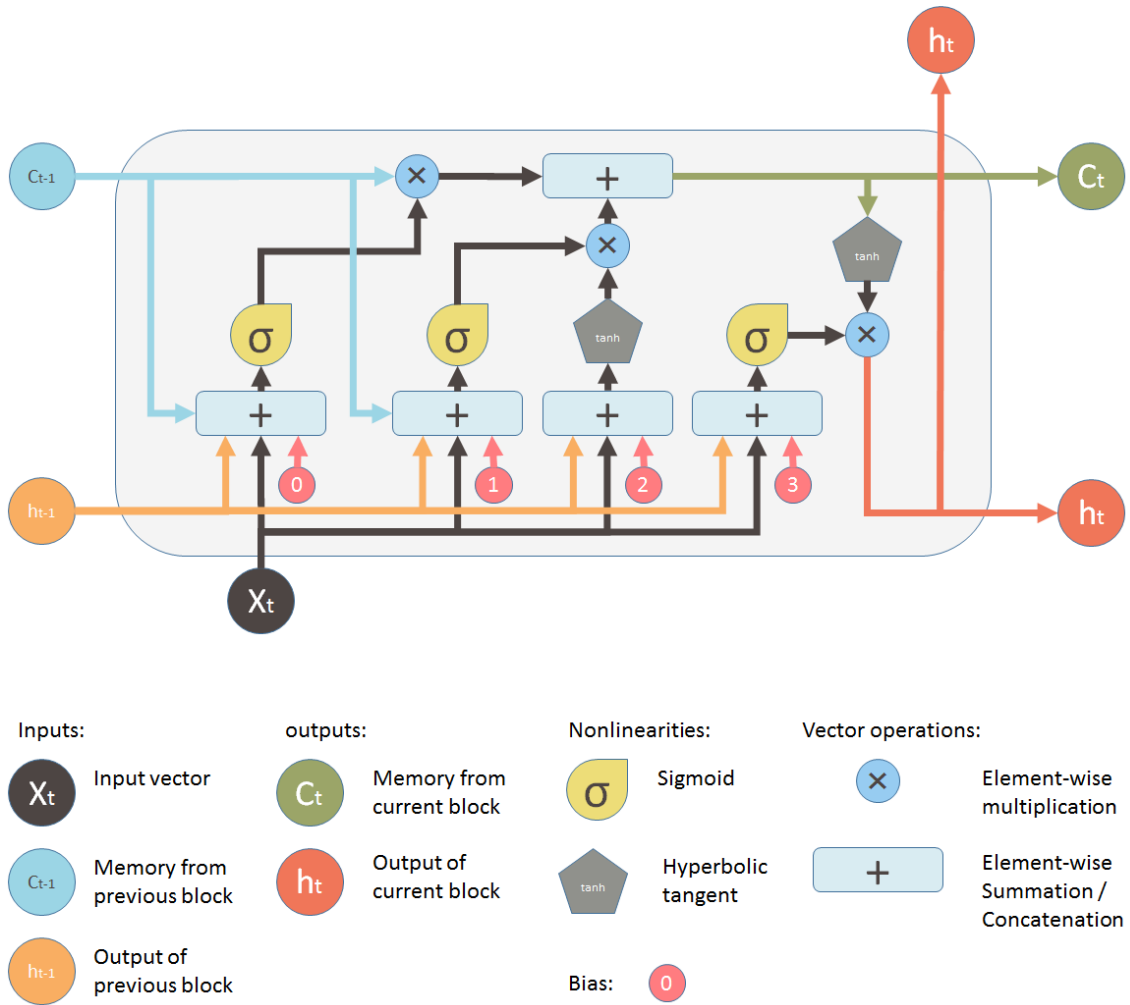


Figure 4.2: LSTM Model

Advantages:

1. LSTM effectively captures sequential dependencies, making it suitable for analyzing text context.
2. It enables the model to understand relationships across lengthy articles as LSTM handles long-term dependencies.

Limitations:

1. LSTM is computationally expensive and time-consuming due to its complex architecture.

4.2.2 DNN

The model Deep Neural Network (DNN) is particularly used to carry out binary text classification. It merges statements and tweets to anticipate a binary target. It engages an embedding layer to transfer the input text sequence into dense vectors of unchangeable length, permitting the model to learn semantic representations of the words. To stop overfitting, a SpatialDropout1D layer is utilized at each update by arbitrarily fixing a fractions of input units to 0 during training. The center of the model is an LSTM layer, capturing

long-term dependencies effectively and context in the sequential text data. The model is trained with the Adam Optimizer and binary cross-entropy loss function, producing binary classification predictions through a dense layer with a sigmoid activation function.

DNN is also one of the deep learning models that are utilized to implement NLP functions to determine whether a particular tweet is either fake or true. In the beginning, the input data is generated from the dataset that contains the combined text of both tweet and statement. Texts are converted to numerical representations when the input data is tokenized. Secondly, the tokenized data is padded to ensure that the tokenized data has the same length as deep learning models require. Thirdly, it transfers to the embedding layer that converts the tokenized data to dense vectors that display semantic relationships between different tokens. Fourthly, to make sure that there is no space for overfitting, the SpatialDropout1D layer removes an entire 1D subset of features and the model does not have to depend on any other features. Fifthly, it is transferred to the LSTM layer which is also similar to RNN, it filters out long-term dependencies and context in the sequence data which is utilized to deduce the semantics of the tweet and statements from the dataset. Finally, it is picked by the Dense Layer which has an activation function of sigmoid, which combines those features for the classification process like 1 for fake and 0 for true for the combined text.

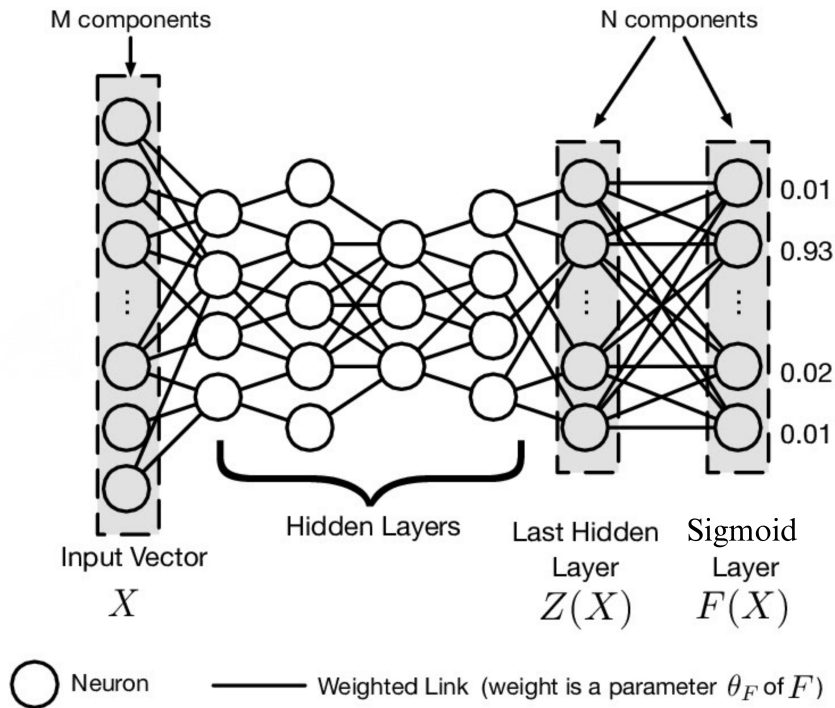


Figure 4.3: DNN Model

Advantages:

1. DNNs are effective for fake news detection as they can learn complex patterns and representations.
2. DNNs can integrate various features (Text, Metadata, etc.), making it very flexible.

Limitations:

1. DNNs are prone to overfitting.
2. DNNs require large labeled datasets for effective training.

4.2.3 Hybrid DNN+LSTM

The Hybrid model is a combination of Deep Neural Network (DNN) and Long Short-Term Memory (LSTM). It increases the ability of each architecture to manipulate varied data types and structures efficiently. By assimilating convolutional layers, they suppress in capturing spatial characteristics, whereas recurrent layers assist the modeling of sequential interdependencies. Also, dense layers volunteer to the extraction of high-level representations. This hybrid approach allows the model to master complex patterns at multiple levels of abstraction.

It becomes apparent that enhanced deep-learning strategies are needed to detect fake news. We use LSTM along with the deep neural networks by integrating them to form a hybrid model to work best for the required task. It attempts to take advantage of LSTM to handle the sequence data and make good use of long-term memory while making good use of the ability of DNN to identify all kinds of patterns and structures present in the data. Thus, the proposed DNN + LSTM hybrid model aims to address the fake news detection problem using overall textual information from different sources with high performance.

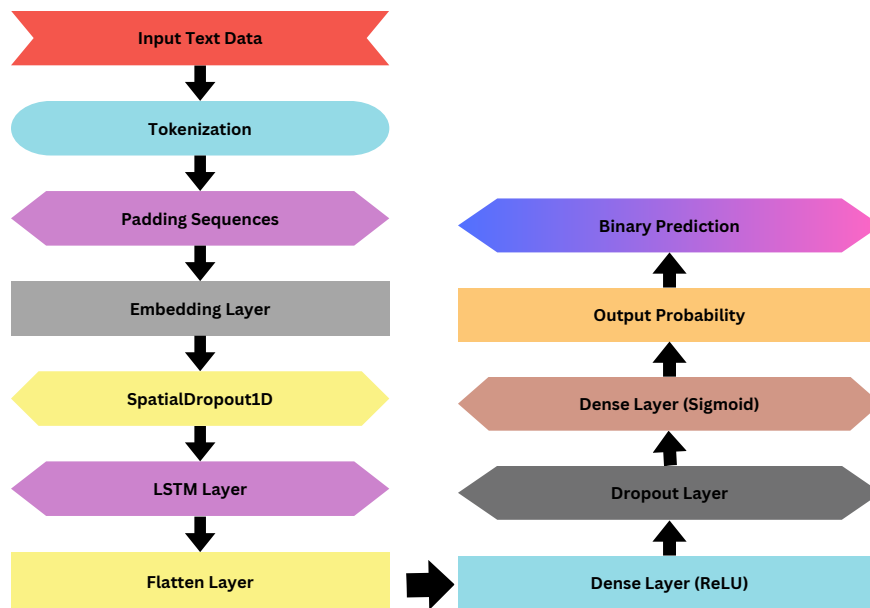


Figure 4.4: DNN+LSTM Model

Building the individual models

Two separate models are defined, model1 which consists of an LSTM architecture and model2 which consists of a DNN architecture.

1. LSTM:

Hence, the LSTM element of our issued model offers an important role when it comes to depicting the sequential nature of textual content. LSTMs are a type of Recurrent Neural Network that is expected to deal with long-term dependencies and the vanishing gradient problem, which is named because of its main characteristic - recurrence. In addition to the tokenization and conversion of text data into word vectors or embeddings, the LSTM layer utilizes the sequence data, which allows the model to correctly apprehend the temporal dynamics inherent in the data. This is because it will help the model retain the relevant and important information so that it can create a sequential output over long sequences for handling the combined tweets and statements. The particular feature that is germane to the LSTM of being able to remember and utilize information from the previous parts of the sequence proves to be advantageous in the ability of the model to uncover patterns associated with fake news, thereby contributing to the efficiency of a DNN + LSTM hybrid.

2. DNN:

The identifying feature of our proposed hybrid model is that in the Deep Neural Network component of the architecture, the model is used to detect non-linear relationships in the data. After the LSTM component, which captures temporal dependencies within the data blocks, the DNN architecture is composed of fully connected layers to further process the extracted features. The dense layers of the DNN coupled with ReLU being the activation function increase the ability of model to identify complex distinctions germane to the segregation of fake and real news. Similarly, dropout layers are also to mitigate overfitting during training by randomly dropping out some neurons and improving the generalizations of the functions in intricate features of a clearer and more reliable output that improves the ability the model to accurately detect false news.

Creating Hybrid DNN+LSTM

In developing the proposed DNN and LSTM hybrid model, it is crucial to identify the features of DNN and LSTM that will help in addressing the challenges involved with textual data analysis. The sequence starts with an Embedding layer responsible for converting the input text to a dense representation of low dimensions and capturing the word semantics. Next is a SpatialDropout1D layer to prevent overfitting of the network on the archive and its constituent features. These embeddings that are formed by RNNs go to the core LSTM layer that analyzes the temporal relations between them and contextual data from the sequences. Then transferred to the Flatten layer that transforms the output into a 2D array ahead of the DNN input. The next component of DNN is the densely connected layers with ReLU activation to better process extracted features in order to identify nonlinear relationships. This convolutional layer randomly drops neurons throughout learning to improve generalization and is named a dropout layer. For the final dense layer, ReLU activation is used and the output layer contains only one neuron, using the sigmoid activation function to predict whether the text is fake or not. This approach combines LSTM since it is good at solving sequences and DNN, which is good at pattern recognition to uncover an effective fake news detection model.

Advantages:

1. The Hybrid model improves accuracy as it uses LSTM for sequential context and DNN for learning complex patterns.
2. It provides a comprehensive analysis as it can process both textual & non-textual data.

Limitations:

1. Designing and tuning the hybrid architecture is complex and may require expensive skills.

4.2.4 BERT+GRU+DNN

To ensure that all the inputs are usable and applicable the dataset is cleaned initially by omitting any data entries with null values in essential columns. The model converts the raw text into a suitable format that can be processed by the neural network using BERT's tokenizer. This includes transforming texts into a series of token IDs, strengthened by the attention masks to inform the model which parts of the data should be focused on. In addition, these sequences are padded to a fixed length to ensure uniformity in input size. To change the input tokens to a dense vector embedding, the model's first layer works as pre-trained BERT. These embeddings are rich in contextual information, setting a solid foundation for sequence learning.

Then, GRUs are used to process the embeddings from the BERT layer. If GRUs are compared to LSTMs, GRUs are simpler and often faster to train, while still capable of capturing dependencies in sequence data effectively. They manage sequence by utilizing gating mechanisms without distinct memory cells.

The DNN, which consists of several dense layers culminating in a prediction layer with a sigmoid activation function to output probabilities indicating class membership receives the processed data from GRU layers. The model is utilized with Adam optimizer for efficient gradient descent and a binary cross-entropy loss function making it efficient for the binary classification tasks. Training incorporates checkpoints to save model weights periodically and callbacks to monitor training progress to ensure that the best model state is preserved based on validation accuracy. During training, validation checks are conducted to adjust parameters and avoid overfitting. This is crucial for tuning the GRU and DNN layers to work effectively with the BERT-generated embeddings. The model's effectiveness is measured using accuracy, along with other relevant metrics like precision and recall, depending on the specific application.

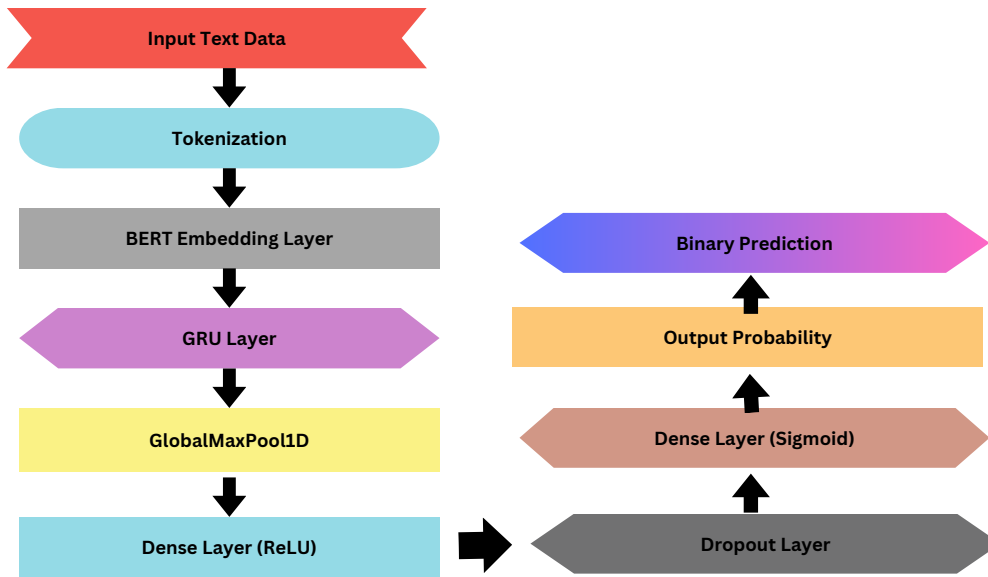


Figure 4.5: BERT+GRU+DNN Model

Advantages:

1. GRU paired with the BERT makes it suitable for applications where both performance and computational speed are priorities.
2. The GRU efficiently handles various lengths and complexities of text data, providing a balance between complexity and performance.

Limitations:

1. Even though the GRUs are less complex than LSTMs, the pairing with BERT still needs a significant amount of computational resources like memory and computing power.
2. The integration of multiple advanced components necessitates careful tuning of hyperparameters to attain optimal performance, which can be a time-consuming and skill-demanding process.

4.2.5 DistilBERT+GRU+DNN

The data preprocessing initializes with cleaning the dataset to make sure of high-quality input for the model. To avoid inaccurate or misleading information during training and predictions, rows with missing values in essential columns, such as 'text' or 'label,' are discarded. Using BERT's tokenizer, text data is converted into a sequence of integers, representing token IDS in the BERT vocabulary which is called tokenization. Then during padding, sequences are padded to a maximum length to standardize input size and guarantee consistent processing within neural networks. To capture both inherent meanings and contextual relationships, the model architecture incorporates a pre-trained BERT

layer, which extracts rich contextual representations of tokens as high-dimensional vectors.

An LSTM layer processes the vectors sequentially to identify immediate and long-term dependencies, which is important for tasks requiring a broader knowledge of sequences. The LSTMs outputs are passed into a dense neural network (DNN) which is comprised of multiple layers that enable the model to learn increasingly abstract features. Equipped with a sigmoid activation function, the last dense layer generates the likelihood of the input having a relationship to a particular class. The model is trained with the Adam optimizer for efficient gradient handling and flexible learning rate capabilities. Additionally, a binary cross-entropy loss function is used for binary classification tasks.

Later on, dropout techniques are used in the LSTM and dense layers to avoid overfitting by randomly eliminating subsets of features during training. Learning rates are carefully adjusted to balance convergence speed with the risk of overshooting minimal loss points, enabling fine-tuning of the pre-trained BERT for specific tasks without compromising its generalization. Evaluation involves validating the model iteratively during training to monitor metrics such as accuracy, precision and recall and giving feedback for parameter adjustments. Finally, the model's performance is determined on a different test set to evaluate its ability to generalize to unrevealed data, reflecting its effectiveness in real-world applications.

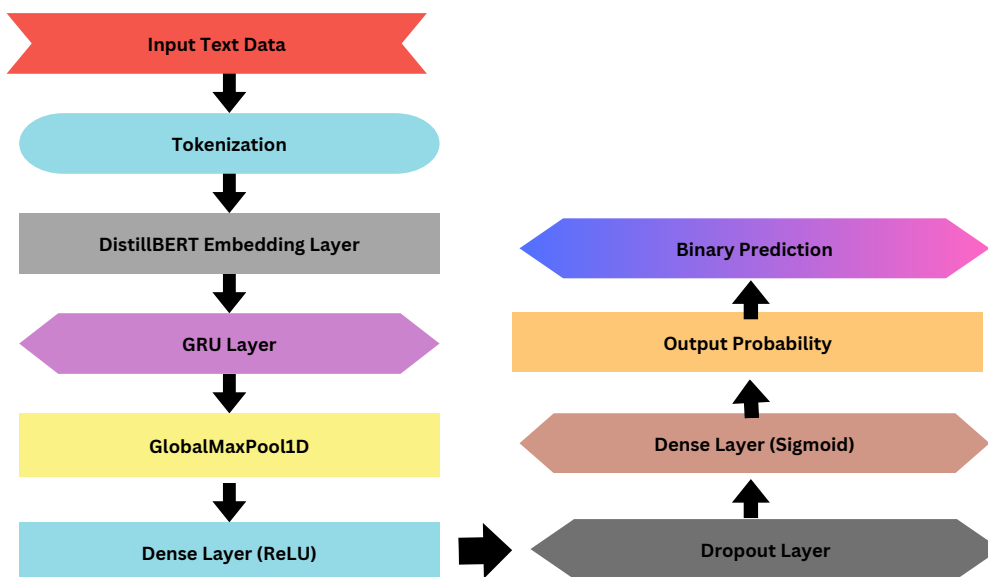


Figure 4.6: DistilBERT+GRU+DNN Model

Advantages:

1. DistilBERT is more faster and scalable than BERT since DistilBERT uses less power and computational resources.

2. GRU brings about a balance between computational efficiency and the ability to capture relevant temporal features in the text data.

Limitations:

1. Although distilBERT retains much of BERT’s effectiveness, some deeper contextual nuances of texts might be lost compared to the full model.
2. Combining various architectures necessitates meticulous tuning and integration, which makes it difficult to attain model stability and hyperparameter optimization.

4.2.6 DistilBERT+LSTM+DNN

The data cleaning process begins by removing rows with missing values in essential columns, ensuring high-quality data is utilized to train the model. Tokenization and input formatting are then established using the DistilBERT tokenizer, which converts text into token IDs and standardizes sequence lengths through padding. Moreover, this process includes the generation of the attention masks, helping the model concentrate on the relevant tokens. The lighter type of BERT, which is called the DistilBERT model, processes these tokenized inputs to extract meaningful contextual embeddings while maintaining computational efficiency.

Following this, an LSTM layer is employed to further process the embeddings from DistilBERT, capturing sequential dependencies and enhancing the model’s ability. Furthermore, LSTM outputs have been transformed into a DNN, where several dense layers are added which help the model learn higher-level abstractions from the data. Consequently, a sigmoid activation is added to the last layers to classify the inputs. To optimize learning, the Adam optimizer is used partially because of its adjustable learning rate capabilities which allow efficient fine-tuning. Later on, the binary class-entropy loss function is utilized to enhance the ability of the model to distinguish separable classes in binary classification tasks.

To enhance generalization and avoid overfitting, dropout is enabled on the LSTM layer as well as on the DNN layers to adjust the model optimally for other unseen data. This makes the model perform flawlessly. Validation sets are performed to monitor performance and adjust the model. during the training phase. Performance metrics such as precision, accuracy and recall are tracked and evaluated the model’s reliability. Lastly, the model is evaluated on a different test set to ensure the model’s effectiveness in handling new, unseen data which will confirm its ability to generalize well to real-world applications.

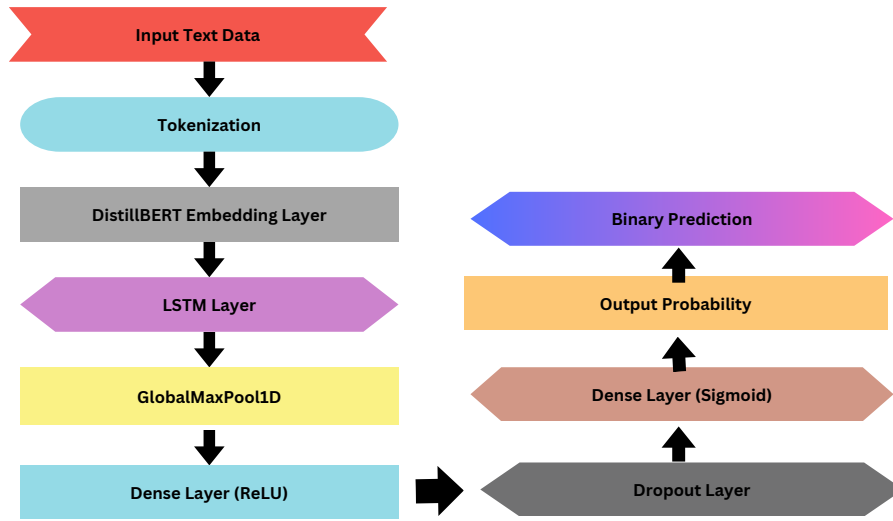


Figure 4.7: DistilBERT+LSTM+DNN Model

Advantages:

1. DistilBERT is applicable for situations with limited GPU resources because it drastically decreases the computational effort when compared to full BERT models.
2. The combination of distilBERT and LSTM aids in a quick yet accurate analysis of textual data by successfully combining performance and speed effectively.

Limitations:

1. Although distilBERT contains most of BERT's capabilities, some subtle nuances in texts may not be captured as effectively as with the full BERT model.
2. The integration of distilBERT with LSTM and DNN layers can introduce complexities which makes it more difficult for model tuning, especially when trying to develop each component's parameters for specific tasks.

4.2.7 TruthForge

The initial step in data preprocessing is cleaning the dataset to make sure that high quality data is fed into the model. This involves discarding any rows with missing values in crucial columns such as "text" or "label". Moreover, text data is converted into a sequence of integers using BERT's tokenizer where each integer corresponds to a token from BERT's vocabulary. This tokenization aids in preserving the contextual use of words. Subsequently, the sequences are padded to a specific maximum length to meet the requirements of the neural networks for a standard input size. The hybrid model utilizes a pre-trained BERT layer to extract contextual representations of each token in the input sequences. This layer generates high dimensional vectors for all tokens that obtain both the token's internal meaning and its context within the sentence.

Following this, the LSTM layer processes these token vectors sequentially, capturing not only the immediate context but also long-term dependencies. It is critical for tasks where knowledge over longer sequences affects the prediction. Then the outputs from the LSTM are fed into DNN dense layers so that higher abstraction can be learned from it. The last dense layer of DNN employs a sigmoid activation function to output a probability by the final dense layer, suggesting the likelihood of the input indicating relationship to a particular class. The model utilizes the Adam optimizer, which is efficient in handling sparse gradients and adaptive learning rate capabilities, while utilizing a binary cross-entropy loss function, ideal for binary classification tasks.

To prevent overfitting, dropout techniques are used in both the LSTM and subsequent DNN dense layers, randomly eliminating a subset of features in each training phase. During training, learning rates are carefully adjusted to balance the speed of convergence with the risk of skipping minimal loss points. This fine-tuning is important when adapting the pre-trained Bert to a specific task without losing the generalization it brings. While observing the training progress, performance metrics such as accuracy, precision, and recall are calculated to validate the model. The model is then assess on a separate test set.

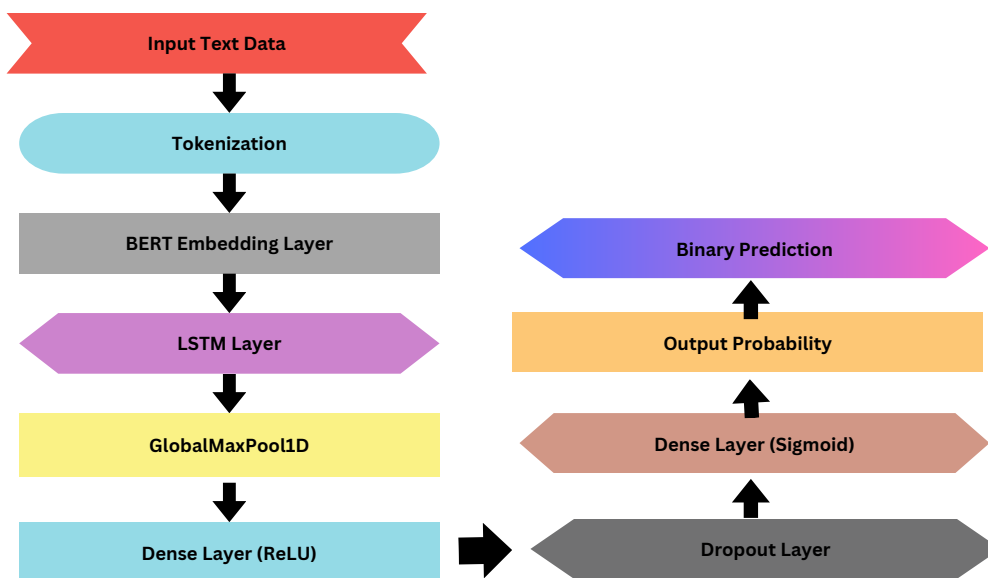


Figure 4.8: TruthForge Model

Advantages:

1. The combination of BERT and LSTM allows the model to comprehend both the order of the sequences and the context in the text, providing a deep semantic and temporal understanding.
2. The model has a layered architecture which enables the model to be flexible to various text-based tasks and scalable when additional data becomes available.

Chapter 5

Result Analysis

5.1 Performance Metrics

We have used five metrics to evaluate performance and make comparisons between different deep learning algorithms. These matrices are accuracy, precision, recall, loss and F1-Score.

1. Accuracy:

Accuracy is the proportion of correctly classified instances out of total instances. It is calculated as follows:

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

Where:

TP = True Positives (Number of Correctly Predicted Positives)

TN = True Negatives (Number of Correctly Predicted Negatives)

FP = False Positives (Number of Incorrectly Predicted Positives)

FN = False Negatives (Number of Incorrectly Predicted Negatives)

2. Precision

How many of the anticipated positive instances turn out to be positive is measured by precision. Here, high precision means a low number of false positives.

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

Where:

TP = True Positives (Number of Correctly Predicted Positives)

FP = False Positives (Number of Incorrectly Predicted Positives)

3. Recall

The number of true positive cases that were accurately detected is measured by recall. When false negatives are expensive, it is helpful. High recall means a low number of false negatives.

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

Where:

TP = True Positives (Number of Correctly Predicted Positives)

FN = False Negatives (Number of Incorrectly Predicted Negatives)

4. Loss

Loss is Binary Cross-Entropy which measures the difference between actual labels and predicted probabilities for classification tasks. Here, a lower loss value indicates better model performances.

$$\text{Loss} = -\frac{1}{N} \sum_{i=1}^N [y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i)]$$

Where:

N = Total number of samples

y_i = Actual label which is 0 or 1

$\hat{y}_i$ = Predicted Probability

5. F1-Score:

The F1-Score is the precision and recall harmonic mean. Particularly, when there is an imbalance between positive and negative classes which balances both measurements. Here, a high F1-Score indicates a good balance between precision and recall.

$$\text{F1-Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

5.2 Performance Evaluation: Deep Learning Approach

5.2.1 LSTM

	Precision	Recall	F1-score
Weighted Avg	85.32%	87.22%	86.26%
Accuracy	85.75%		

Table 5.1: Performance Metrics of LSTM Model

From Table 5.1, we can see that the LSTM model shows a precision (Weighted Average) of 85.32 percent, recall (weighted average) of 87.22 percent, F1 score (Weighted Average) of 86.26 percent and accuracy of 85.75 percent. The precision demonstrates the accuracy of the model in forecasting positive instances. The recall rate shows the model's moderate ability to correctly identify all applicable occurrences. The F1 score points to a high degree of balance performance by merging both precision and recall. Finally, the accuracy shows the model's overall performance.

5.2.2 DNN

	Precision	Recall	F1-score
Weighted Avg	85.23%	86.82%	86.02%
Accuracy	85.52%		

Table 5.2: Performance Metrics of DNN Model

From Table 5.2, we can see that the DNN model shows a precision (Weighted Average) of 85.23 percent, recall (weighted average) of 86.82 percent, F1 score (Weighted Average) of 86.02 percent and accuracy of 85.52 percent. The precision of DNN is almost the same as the precision of LSTM which suggests that the accuracy of both models is similar to that of forecasting positive instances. The recall rate is slightly lower in DNN than in LSTM, suggesting that DNN is slightly lacking in forecasting positive instances. Also, DNN has a lower F1 score than LSTM, suggesting a slight degree of balance and performance. Finally, the accuracy shows the model's overall performance.

5.3 Performance Evaluation: Hybrid Deep Learning Approach

5.3.1 Hybrid LSTM DNN: Without Fine-Tuning

Max_ Length	Epoch	Train_ Loss	Val_ Loss	Acc- uracy	Test_ Accu- racy	Test_ F1- score	Test_ Peci- sion	Test_ Recall
20	10	0.2722	0.2546	88.13%	89.66%	90.26%	87.08%	93.68%
25	10	0.2372	0.2214	91.15%	91.62%	92.06%	89.31%	94.99%
50	10	0.1241	0.1036	95.38%	96.71%	96.78%	96.91%	96.65%

Table 5.3: Performance Metrics of Hybrid LSTM+DNN Model for Different Max Lengths

From Table 5.3, we can see that the LSTM+DNN model was evaluated with varying Max_Length, while keeping the number of epochs constant at 10 for the CIC truth seeker dataset. We achieve the best test result when the Max_Length is 50. The best result shows an accuracy of 96.71 percent, precision of 96.91 percent, F1 score of 96.78 percent and recall of 96.65 percent. These metrics indicate the model performs optimally with a longer input sequence length.

5.3.2 Hybrid LSTM DNN: Fine-Tuning

Max_ Length	Epoch	Train_ Loss	Val_ Loss	Acc- uracy	Test_ Accu- racy	Test_ F1- score	Test_ Peci- sion	Test_ Recall
20	10	0.6296	0.5211	71.37%	93.32%	92.95%	93.38%	92.52%
25	10	0.7025	0.6206	70.55%	90.93%	90.29%	91.96%	88.68%
50	10	0.8097	0.7093	58.30%	71.36%	73.68%	65.44%	84.30%

Table 5.4: Performance Metrics of Hybrid LSTM+DNN Model for Different Max Lengths

From Table 5.4, we can see that the LSTM+DNN model was evaluated with varying Max_Length, while keeping the number of epochs constant at 10 for the ISOT dataset. The best test results were achieved with a Max_Length of 20. This configuration led to an accuracy of 93.32 percent, a precision of 93.38 percent, an F1 score of 92.95 percent, and a recall of 92.52 percent. These metrics indicate the model performs optimally with a shorter input sequence length.

5.4 Performance Evaluation: Advanced Deep Learning Approach

5.4.1 BERT+GRU+DNN: Without Fine-Tuning

Max_ Length	Epoch	Train_ Loss	Val_ Loss	Acc- uracy	Test_ Accu- racy	Test_ F1- score	Test_ Peci- sion	Test_ Recall
20	8	0.3369	0.2802	85.43%	88.73%	88.84%	90.11%	87.60%
24	8	0.3031	0.2390	87.71%	90.06%	89.85%	91.57%	86.02%
28	8	0.2587	0.2110	89.47%	92.44%	92.62%	92.49%	92.75%

Table 5.5: Performance Metrics of BERT+GRU+DNN Model for Different Max Lengths

From Table 5.5, we can see that the BERT+GRU+DNN model was evaluated on the Truth Seeker dataset, with the number of epochs fixed at 8. The best performance was achieved when the Max_length is set to 28. This setup yielded the highest accuracy of 92.44 percent, precision of 92.49 percent, an F1 score of 92.62 percent and a recall of 92.75 percent. The model benefits from a slightly longer input sequence.

5.4.2 BERT+GRU+DNN: Fine-Tuning

Max_ Length	Epoch	Train_ Loss	Val_ Loss	Acc- uracy	Test_ Accu- racy	Test_ F1- score	Test_ Peci- sion	Test_ Recall
20	8	0.3323	0.2499	86.24%	95.96%	96.76%	95.48%	96.04%
24	8	0.3320	0.2382	87.01%	97.99%	97.89%	97.55%	98.24%
28	8	0.3211	0.2579	86.67%	97.07%	96.97%	95.17%	98.75%

Table 5.6: Performance Metrics of BERT+GRU+DNN Model for Different Max Lengths

From Table 5.6, we can see that the BERT+GRU+DNN model was evaluated with varying Max_Length, while keeping the number of epochs constant at 8 for the ISOT dataset. The best test results were achieved with a Max_Length of 24. This configuration led to an accuracy of 97.99 percent, a precision of 97.55 percent, an F1 score of 97.89 percent and a recall of 98.24 percent. These metrics indicate the model performs optimally with a moderate input sequence length.

5.4.3 DistilBERT+GRU+DNN: Without Fine-Tuning

Max_ Length	Epoch	Train_ Loss	Val_ Loss	Acc- uracy	Test_ Accu- racy	Test_ F1- score	Test_ Peci- sion	Test_ Recall
20	8	0.3004	0.2669	86.94%	88.96%	88.81%	92.25%	85.61%
24	8	0.2690	0.2299	89.00%	90.68%	90.48%	94.75%	86.58%
28	8	0.2442	0.2014	90.38%	92.40%	92.81%	89.97%	95.82%

Table 5.7: Performance Metrics of DistilBERT+GRU+DNN Model for Different Max Lengths

From Table 5.7, we can see that the DistilBERT+GRU+DNN model was evaluated on the CIC Truth Seeker dataset, with the number of epochs fixed at 8. The best performance was achieved when the Max_length is set to 28. This setup yielded the highest accuracy of 92.40 percent, a precision of 89.97 percent, an F1 score of 92.81 percent and a recall of 95.82 percent. These metrics indicate that the model performs optimally with a longer input sequence length.

5.4.4 DistilBERT+GRU+DNN: Fine-Tuning

From Table 5.8, we can see that the distilBERT+GRU+DNN model was evaluated with varying Max.length, while keeping the number of epochs constant at 8 for the ISOT dataset. The best test results were achieved with a Max.length of 24. This configuration led to an accuracy of 97.40 percent, a precision of 97.35 percent, an F1 score of 97.32 percent and a recall of 97.29 percent. These metrics indicate that the model performs optimally with a moderate input sequence length.

Max_ Length	Epoch	Train_ Loss	Val_ Loss	Acc- uracy	Test_ Accu- racy	Test_ F1- score	Test_ Peci- sion	Test_ Recall
20	8	0.2971	0.2273	88.27%	96.12%	95.92%	95.99%	95.85%
24	8	0.3464	0.2447	89.31%	97.40%	97.32%	97.35%	97.29%
28	8	0.3578	0.2832	87.54%	97.34%	97.26%	96.72%	97.81%

Table 5.8: Performance Metrics of DistilBERT+GRU+DNN Model for Different Max Lengths

5.4.5 DistilBERT+LSTM+DNN: Without Fine-Tuning

Max_ Length	Epoch	Train_ Loss	Val_ Loss	Acc- uracy	Test_ Accu- racy	Test_ F1- score	Test_ Peci- sion	Test_ Recall
20	8	0.2885	0.2583	87.77%	89.53%	89.87%	89.17%	90.59%
24	8	0.2555	0.2199	90.17%	90.88%	90.82%	93.84%	87.99%
28	8	0.2210	0.1931	91.23%	93.04%	93.34%	91.30%	95.47%

Table 5.9: Performance metrics of DistilBERT+LSTM+DNN Model for Different Max Lengths

From Table 5.9, we can see that the DistilBERT+LSTM+DNN model was evaluated on the CIC Truth Seeker dataset, with the number of epochs fixed at 8. The best performance was achieved when the Max_length is set to 28. This setup yielded the highest accuracy of 93.04 percent, a precision of 91.30 percent, an F1 score of 93.34 percent and a recall of 95.47 percent. These metrics indicate that the model performs optimally with a moderate input sequence length.

5.4.6 DistilBERT+LSTM+DNN: Fine-Tuning

Max_ Length	Epoch	Train_ Loss	Val_ Loss	Acc- uracy	Test_ Accu- racy	Test_ F1- score	Test_ Peci- sion	Test_ Recall
20	8	0.2891	0.2238	88.16%	96.22%	96.04%	95.76%	96.32%
24	8	0.3448	0.2758	88.30%	97.27%	97.18%	97.33%	97.03%
28	8	0.3555	0.2712	89.91%	97.63%	97.56%	97.41%	97.71%

Table 5.10: Performance Metrics of DistilBERT+LSTM+DNN Model for Different Max Lengths

From Table 5.10, we can see that the DistilBERT+LSTM+DNN model was evaluated on the ISOT dataset, with the number of epochs fixed at 8. The best performance was achieved when the Max_length is set to 28. This setup yielded the highest accuracy of

97.63 percent, a precision of 97.41 percent, an F1 score of 97.56 percent and a recall of 97.71 percent. These metrics indicate that the model performs optimally with a moderate input sequence length.

5.4.7 TruthForge: Without Fine-Tuning

Max_ Length	Epoch	Train_ Loss	Val_ Loss	Acc- uracy	Test_ Accu- racy	Test_ F1- score	Test_ Peci- sion	Test_ Recall
20	8	0.3131	0.2707	86.51%	88.90%	88.89%	91.06%	86.83%
24	8	0.2714	0.2325	88.45%	91.05%	91.23%	91.57%	90.87%
28	8	0.2426	0.2015	90.22%	92.81%	93.01%	92.50%	93.53%

Table 5.11: Performance Metrics of TruthForge Model for Different Max Lengths

After the preprocessing, Bert was used for embedding. Then the data was passed through LSTM and DNN layers to train the model. We have trained the model on 80% of the data from the Truth Seeker dataset. We have used different Max_Length to evaluate the result whereas we found the best performance for Max_Length 28. We have used hold-out validation and ran 8 epochs to achieve that result. Then we implemented this state on our test data to evaluate the final performance. The results in our test data were considered as our main result. If the test data has the best results in certain parameters that means the model was well-trained because of that value.

From Table 5.11, we can see that the highest accuracy of 92.81 percent, precision of 92.50 percent, an F1 score of 93.01 percent and a recall of 93.53 percent were achieved at Max_Length 28. Since the lowest average val_loss is 0.2015 at Max_Length 28. We took it as our best performance. The model benefits from a slightly longer input sequence as suggested by the results, successfully utilizing the additional context to enhance overall prediction metrics.

5.4.8 TruthForge: Fine-Tuning

Max_ Length	Epoch	Train_ Loss	Val_ Loss	Acc- uracy	Test_ Accu- racy	Test_ F1- score	Test_ Peci- sion	Test_ Recall
20	8	0.3204	0.2446	86.45%	96.06%	95.88%	95.45%	96.32%
24	8	0.2326	0.1418	91.28%	98.07%	97.97%	97.82%	98.12%
28	8	0.3124	0.2258	87.06%	98.20%	98.13%	97.46%	98.81%

Table 5.12: Performance Metrics of TruthForge Model for Different Max Lengths

We ran our model in the Truth Seeker dataset and with the value that we got we anticipated that our model could perform better than that. So we have used another dataset

“ISOT” which contains news articles and their title and subject of that news. Which contains different contents than our social media dataset. The model was then fine-tuned with a smaller learning rate (2e-5) to better capture domain-specific nuances and improve performance.

The model was then evaluated on the ISOT dataset, with the number of epochs fixed at 8. From Table 5.12, we can see that the best performance was achieved when the Max_Length is set to 28. This setup yielded the highest accuracy of 98.20 percent, a precision of 97.46 percent, an F1 score of 98.13 percent and a recall of 98.81 percent. The model benefits from a slightly longer input sequence as suggested by the results, successfully utilizing the additional context to enhance overall prediction metrics.

5.5 Explainability Analysis

The model receives two inputs, one is text or tweet and the second is a statement. Here, it checks the words of the tweet or text and for words like “right” and “american” it predicts True and for “Biden”, “come” and “out” it predicts False. The word “Biden” has 52 percent influence, “american” has 20 percent influence, “right” has 30 percent influence, “come” has 15 percent influence, “out” has 12 percent influence, “of” has 4 percent influence, “I” has 4 percent influence, “being” has 3 percent influence, “if ” has 2 percent influence and “images” has 2 percent influence.

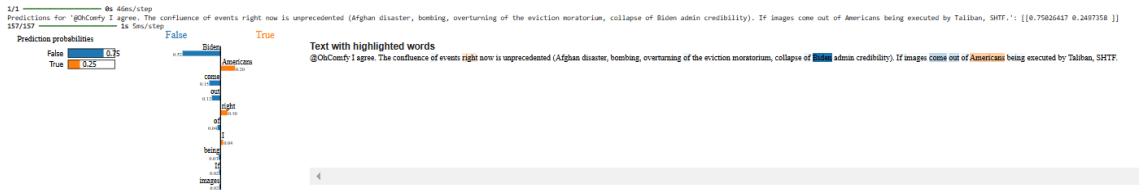


Figure 5.1: LIME: Text

Here, it matches a tweet and a statement. It is showing true for 'American', 'in', 'pandemic', 'millions', 'moratorium', 'means', 'could' and showing false for 'middle', 'end', 'their'. The word “American” has 10 percent influence, “in” has 5 percent influence, “pandemic” has 5 percent influence, “millions” has 39 percent influence, “moratorium” has 4 percent influence, “means” has 4 percent influence, “could” has 4 percent influence, “middle” has 19 percent influence, “End” has 11 percent influence and “their” has 7 percent influence.

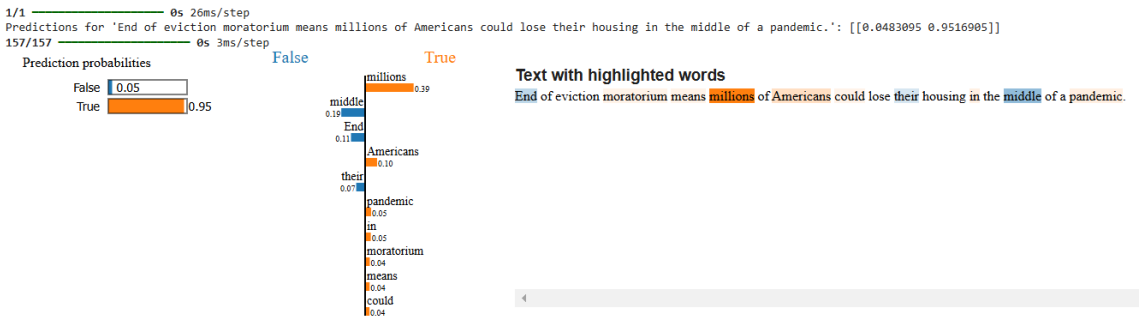


Figure 5.2: LIME: Statement

5.6 Best Model

Model	Accuracy	Precision	Recall	F1-score
LSTM	85.75%	85.32%	87.22%	86.26%
DNN	85.52%	85.23%	86.82%	86.02%
Hybrid LSTM+DNN	96.71%	96.91%	96.65%	96.78%
Hybrid LSTM+DNN Fine tuned	93.32%	93.38%	92.52%	92.95%
BERT+GRU +DNN	92.44%	92.49%	92.75%	92.62%
BERT+GRU+DNN Fine tuned	97.99%	97.55%	98.24%	97.89%
DistilBERT+GRU +DNN	92.40%	89.97%	95.82%	92.81%
DistilBERT+GRU +DNN Fine tuned	97.40%	97.35%	97.29%	97.32%
DistilBERT+LSTM +DNN	93.04%	91.30%	95.47%	93.34%
DistilBERT+ LSTM+DNN Fine tuned	97.63%	97.41%	97.71%	97.56%
TruthForge	92.81%	92.50%	93.53%	93.01%
TruthForge Fine tuned	98.20%	97.46%	98.81%	98.13%

Table 5.13: Comparison between Performance of different Models Before and After Fine-tuning

From Table 5.13, we can see that the "Truth Forge" (Bert+LSTM+DNN) fine-tuned has an accuracy of 98.20% and was regarded as the best-performing model among the other models. The reason was that Bert Tokenizer has a subword unit mechanism that deals with rare words that are not part of the vocabulary something the normal Tensorflow tokenizer fails at. Bert Tokenizer also captures the contextual elements that TensorFlow tokenizer cannot. Furthermore, Bert embeddings have richer semantic information compared to normal Tensorflow embeddings. Also, LSTM is a better component of RNN compared to GRU as it has 3 information gates compared to 2 gates in GRU. LSTM can store information efficiently and also capture long-term dependencies whereas GRU is good with shorter sequences. Hence, the combination of BERT and LSTM outperforms all the other model combinations and shows greater results.

Pa per	Dataset Type	Model Name	Evaluation Metrics	Deployment Platform	Performance
[26]	CIC Truth Seeker Dataset (Text-based)	LSTM	Precision, Recall, F1- score	Python-based framework with TensorFlow or Keras with LSTM	Acc : 96%, Pre: 93%, Rec: 96%, F1 : 94%
[22]	Fake News Dataset, Twitter15, Twitter16, Weibo (Text and Image)	Deep Learning- based models	Comparison of Accuracy	Python-based deep learning platforms with GAN and BERT support	Fake news- Acc:98.90%, LIAR- Acc:99%, Twitter16- Acc:96.77%
[18]	PHEME Dataset (Text- based)	SPOT	Accuracy, F1-score, Recall, Error Rate, Log Loss, G- mean, ROC	Python-based frameworks with SentiWordNet	Acc:99.15%, Pre:97.26%, Rec:97.27%
[12]	Kaggle Dataset (Text-based)	DNN	Accuracy, Precision, Recall, F1- score	Python-based frameworks with Nvidia DGX	Acc:98.3%, Pre:90.1935%, Rec:90%, F1:90%
[9]	BuzzFeed and Politi- Fact,GeoText Dataset, UT- Geo2011 Dataset(Text)	GCN, MENET	Accuracy, Mean and Median Dis- tance Error	Python-based graph platforms	BuzzFeed accuracy- GCN :87.8%, Politi- Fact accuracy- GCN: 89.5%
[23]	ISO and FA- KES datasets (Text-based)	Hybrid CNN- RNN Based DL	Accuracy, Precision, Recall, F1- score, Loss Graphs	Python-based DL framework with hybrid support	Acc:99% ,Pre:99%, Rec:99%, F1:99%
[21]	BuzzFeed and PolitiFact (Text-based)	Echo- FakeD	Precision, Recall, F1-score, Accuracy	Python frame- work for tensor operations	BuuzFeed- Acc:91.80%, Pre:90.47%, Rec:86.37%, F1:88.37%, PolitiFact- Acc:99%, Pre:99%, Rec:99%, F1:99%

Table 5.14: Summary of Research Studies on Fake News Detection: Part 1

Pa per	Dataset Type	Model Name	Evaluation Metrics	Deployment Platform	Performance
[16]	Custom News Dataset(Text based)	Deep learning based models	Accuracy	Python based ML platforms, deployed on standard systems	Accuracy: 91%
[13]	Snopes, PolitiFact, NewsTrust, SemEval-2017 (Text based)	DeClarE	Accuracy, F1-score, AUC	Python-based framework with TensorFlow and Keras	DeClarE(Full) on Politifact True claim Acc:67.32%, False claim Acc:69.62%, Macro F1 score:68%, AUC:75%
[11]	LIAR dataset (Text based)	Deep learning based models	Accuracy, F1-score	Python-based framework with TensorFlow	GRU- Acc: 21.7%, LSTM- Acc: 21.66%, Vanilla- Acc: 21.5%
[4]	Kaggle Fake News dataset, Signal Media Dataset (Text-based data)	Deep Learning based models	Precision, Recall, F1-Score	Python-based framework with TensorFlow	CNN with Max Pool and Att, Rec: 0.03%, Pre: 0.97%, F1: 0.06%
[24]	DataHeart dataset (Text-based)	LSTM, 14-Layer BLSTM	Accuracy, F1-score, Precision, Recall, Error function	Custom frame-works for Persian data	Acc: 91.08%, Rec: 91.10%, F1 : 91.57%
[7]	Twitter and Weibo (Text based)	CSI	Accuracy, F1-score	Scalable deep learning plat-forms	Twitter- Acc: 89.2%, F1: 89.4%, Weibo- Acc: 95.3%, F1: 95.4%
Proposed Model	CIC Truth-seeker and ISOT	Truth-Forge	Accuracy, Precision, Recall, F1 score	Python-based framework with TensorFlow and Keras with Bert support	Accuracy: 98.2%, Precision: 97.46%, Recall: 98.81%, F1: 98.13%

Table 5.15: Summary of Research Studies on Fake News Detection: Part 2

Chapter 6

Conclusion and Future Plan

As the internet has become a free and open space for users to access information, express their opinions and for readers to form their judgments around the news, social media have emerged as valuable platforms for spreading real news while also being vulnerable to the escalation of fake news. In our thesis, we used two standalone, a hybrid model and four advanced hybrid deep learning models to differentiate fake and real information from the huge amount of news circulating on various social media. The traditional deep learning models we have utilized for this work are two deep learning models Deep Neural Network (DNN) and Long Short-term Memory (LSTM), a hybrid model of DNN and LSTM, three advanced deep learning architectures such as DistilBERT+LSTM+DNN, DistilBERT+GRU+DNN, BERT+GRU+DNN and our TruthForge model to determine the most accurate and efficient approach. These hybrid models were fine-tuned using transformer-based architectures like BERT and DistilBERT. Through extensive evaluation, the TruthForge model emerged as the best-performing model, achieving 98.2% accuracy and surpassing all other models. We utilized two different datasets to provide more valuable insights. Our further research could focus on incorporating multimodal data, such as images and videos and developing an automated fake news detection application to enhance real-world solutions.

Future Plan

- **Multimodal Data Integration and Analysis:**

We can investigate the utilization of several data sources, such as photos, videos and user interactions, to enhance the detection skills of deep learning models. Using multimodal models like CLIP will extend our research of this thesis. This would create a comprehensive fake news detection system across different content formats more effectively.

- **Real-Time Application Development:**

We can implicate the findings for real-world applications. For example, deploying the best model with APIs and integrating it with social media monitoring tools will be a great step for our research. Developing solutions for the detection of fake news soon after it appears on social media platforms can help to reduce the escalating of misinformation.

- **Addressing Adversarial Harms and User-Friendly:**

We can evaluate the resilience of deep learning models to attacks from adversaries and manipulations. To withstand attempts to deceive or bypass the detection algorithms, develop strategies to improve the robustness and design an intuitive interface for false news detection systems to facilitate easy navigation.

Bibliography

- [1] X. Zhang, J. Zhao, and Y. LeCun, “Character-level convolutional networks for text classification,” *Advances in neural information processing systems*, vol. 28, 2015.
- [2] H. Ahmed, I. Traore, and S. Saad, “Detecting opinion spams and fake news using text classification,” *Security and Privacy*, vol. 1, e9, Dec. 2017. DOI: 10.1002/spy2.9.
- [3] H. Allcott and M. Gentzkow, “Social media and fake news in the 2016 election,” *Journal of Economic Perspectives*, vol. 31, no. 2, pp. 211–36, May 2017. DOI: 10.1257/jep.31.2.211. [Online]. Available: <https://www.aeaweb.org/articles?id=10.1257/jep.31.2.211>.
- [4] S. Bajaj, “The pope has a new baby!” *Fake news detection using deep learning*, pp. 1–8, 2017.
- [5] Y.-C. Chen, Z.-Y. Liu, and H.-Y. Kao, “Ikm at semeval-2017 task 8: Convolutional neural networks for stance detection and rumor verification,” in *Proceedings of the 11th international workshop on semantic evaluation (SemEval-2017)*, 2017, pp. 465–469.
- [6] Y. Long, Q. Lu, R. Xiang, M. Li, and C.-R. Huang, “Fake news detection through multi-perspective speaker profiles,” in *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, G. Kondrak and T. Watanabe, Eds., Taipei, Taiwan: Asian Federation of Natural Language Processing, Nov. 2017, pp. 252–256. [Online]. Available: <https://aclanthology.org/I17-2043>.
- [7] N. Ruchansky, S. Seo, and Y. Liu, “Csi: A hybrid deep model for fake news detection,” in *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, 2017, pp. 797–806.
- [8] W. Y. Wang, ““liar, liar pants on fire”: A new benchmark dataset for fake news detection,” *arXiv preprint arXiv:1705.00648*, 2017.
- [9] N. Deligiannis, T. Huu, D. M. Nguyen, and X. Luo, “Deep learning for geolocating social media users and detecting fake news,” in *NATO Workshop*, 2018.
- [10] S. Ghosh and C. Shah, “Towards automatic fake news classification,” *Proceedings of the Association for Information Science and Technology*, vol. 55, no. 1, pp. 805–807, 2018.
- [11] S. Girgis, E. Amer, and M. Gadallah, “Deep learning algorithms for detecting fake news in online text,” in *2018 13th International Conference on Computer Engineering and Systems (ICCES)*, 2018, pp. 93–97. DOI: 10.1109/ICCES.2018.8639198.

- [12] R. K. Kaliyar, "Fake news detection using a deep neural network," in *2018 4th international conference on computing communication and automation (ICCCA)*, IEEE, 2018, pp. 1–7.
- [13] K. Popat, S. Mukherjee, A. Yates, and G. Weikum, "Declare: Debunking fake news and false claims using evidence-aware deep learning," *arXiv preprint arXiv:1809.06416*, 2018.
- [14] P. T. Tin, "A study on deep learning for fake news detection," *JAIST: Japan Advanced Institute of Science and Technology*, pp. 1–49, 2018.
- [15] J. Cao, Q. Sheng, P. Qi, L. Zhong, Y. Wang, and X. Zhang, "False news detection on social media," *arXiv preprint arXiv:1908.10818*, 2019.
- [16] C. K. Hiramath and G. Deshpande, "Fake news detection using deep learning techniques," in *2019 1st International Conference on Advances in Information Technology (ICAIT)*, IEEE, 2019, pp. 411–415.
- [17] S. Kaushik and C. Gandhi, "Ensure hierarchal identity based data security in cloud environment," *International Journal of Cloud Applications and Computing (IJ-CAC)*, vol. 9, no. 4, pp. 21–36, 2019.
- [18] V. Sabeeh, M. Zohdy, A. Mollah, and R. Al Bashaireh, "Fake news detection on social media using deep learning and semantic knowledge sources," *International Journal of Computer Science and Information Security (IJCSIS)*, vol. 18, no. 2, pp. 45–68, 2020.
- [19] M. Umer, Z. Imtiaz, S. Ullah, A. Mehmood, G. S. Choi, and B.-W. On, "Fake news stance detection using deep learning architecture (cnn-lstm)," *IEEE Access*, vol. 8, pp. 156 695–156 706, 2020. DOI: 10.1109/ACCESS.2020.3019735.
- [20] X. Zhang and A. A. Ghorbani, "An overview of online fake news: Characterization, detection, and discussion," *Information Processing Management*, vol. 57, no. 2, p. 102 025, 2020, ISSN: 0306-4573. DOI: <https://doi.org/10.1016/j.ipm.2019.03.004>. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0306457318306794>.
- [21] R. K. Kaliyar, A. Goswami, and P. Narang, "Echofaked: Improving fake news detection in social media with an efficient deep neural network," *Neural computing and applications*, vol. 33, pp. 8597–8613, 2021.
- [22] 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," *IEEE Access*, vol. 9, pp. 156 151–156 170, 2021.
- [23] J. A. Nasir, O. S. Khan, and I. Varlamis, "Fake news detection: A hybrid cnn-rnn based deep learning approach," *International Journal of Information Management Data Insights*, vol. 1, no. 1, p. 100 007, 2021.
- [24] M. M. Sadr *et al.*, "The use of lstm neural network to detect fake news on persian twitter," *Turkish Journal of Computer and Mathematics Education (TURCOMAT)*, vol. 12, no. 11, pp. 6658–6668, 2021.
- [25] S. Dadkhah, X. Zhang, A. G. Weismann, A. Firouzi, and A. A. Ghorbani, "The largest social media ground-truth dataset for real/fake content: Truthseeker," *IEEE Transactions on Computational Social Systems*, pp. 1–15, 2023. DOI: 10.1109/TCSS.2023.3322303.

- [26] Y. Sharrah, D. Al-Fraihat, and M. Alsmirat, “Deep neural networks in social media forensics: Unveiling suspicious patterns and advancing investigations on twitter,” in *2023 3rd Intelligent Cybersecurity Conference (ICSC)*, IEEE, 2023, pp. 95–102.
- [27] F. W. R. Tokpa, B. H. Kamagaté, V. Monsan, and S. Oumtanaga, “Fake news detection in social media: Hybrid deep learning approaches,” *J. Adv. Inf. Technol*, vol. 14, no. 3, pp. 606–615, 2023.
- [28] T. Diplomat, “Bangladesh’s new democracy under threat from flood of misinformation,” *The Diplomat*, Aug. 2024, Accessed: 2025-01-29. [Online]. Available: <https://thediplomat.com/2024/08/bangladeshs-new-democracy-under-threat-from-flood-of-misinformation/>.