**TRIBHUVAN UNIVERSITY**

**INSTITUTE OF ENGINEERING**

**PULCHOWK CAMPUS**

**THESIS NO: 075MSICE014**

# HYBRID NEURAL NETWORK FOR FAKE NEWS STANCE DETECTION


**BY**

**NIROJ GHIMIRE**

**075MSICE014**


**A**

**THESIS REPORT**

**SUBMITTED TO THE DEPARTMENT OF ELECTRONICS AND COMPUTER ENGINEERING IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF MASTER OF SCIENCE IN INFORMATION AND COMMUNICATION ENGINEERING**

**DEPARTMENT OF ELECTRONICS AND COMPUTER ENGINEERING**

**LALITPUR, NEPAL**

**AUGUST, 2021**

# HYBRID NEURAL NETWORK FOR FAKE NEWS STANCE DETECTION

BY

NIROJ GHIMIRE

075MSICE014

Thesis Supervisor

Dr. Surendra Shrestha

A thesis submitted in partial fulfillment of the requirements for the degree of Masters of Science in Information and Communication Engineering

Department of Electronics and Computer Engineering

Institute of Engineering, Pulchowk Campus

Tribhuvan University

Lalitpur, Nepal

August, 2021

## DECLARATION

I declare that the work hereby submitted for Master of Science in Information and Communication Engineering (MSICE) at IOE, Pulchowk Campus entitled **"HYBRID NEURAL NETWORK FOR FAKE NEWS STANCE DETECTION"** is my own work and has not been previously submitted by me at any university for any academic award.

I authorize IOE, Pulchowk Campus to lend this thesis to other institution or individuals for the purpose of scholarly research.

NIROJ GHIMIRE
075MSICE014
Date: AUGUST, 2021

## RECOMMENDATION

The undersigned certify that they have read and recommended to the Department of Electronics and Computer Engineering for acceptance, a thesis entitled **"HYBRID NEURAL NETWORK FOR FAKE NEWS STANCE DETECTION"**, submitted by **Niroj Ghimire** in partial fulfillment of the requirement for the award of the degree of **"Master of Science in Information and Communication Engineering"**.

...........................................................................

**Supervisor: Surendra Shrestha, PhD**
**Associate Professor, Department of Electronics and Computer Engineering**
**Institute of Engineering, Tribhuvan University**

...........................................................................

**External Examiner: Adesh Khadka**
**Under Secretary (IT)**
**Ministry of Education, Science and Technology**
**Government of Nepal**

.........................................................................................

**Committee Chairperson: Basanta Joshi, PhD**
**Program Coordinator, Msc in Information and Communication Engineering**
**Department of Electronics and Computer Engineering,**
**Institute of Engineering, Tribhuvan University**

**Date: August, 2021**

## DEPARTMENTAL ACCEPTANCE

The thesis entitled **“HYBRID NEURAL NETWORK FOR FAKE NEWS STANCE DETECTION"**, submitted by **Niroj Ghimire** in partial fulfillment of the requirement for the award of the degree of **“Master of Science in Information and Communication Engineering"** has been accepted as a bonafide record of work independently carried out by him in the department.

..............................................

**Prof. Dr. Ram Krishna Maharjan**
Head
Department of Electronics and Computer Engineering
Institute of Engineering, Pulchowk Campus
Pulchowk, Lalitpur, Nepal

## ACKNOWLEDGEMENT

It has been a real joy for me to prepare my dissertation, and I would like to take this opportunity to thank everyone who has made it possible. I would like to express my heartfelt gratitude and heartfelt thanks to my Supervisor Dr. Surendra Shrestha, Associate Professor, Institute of Engineering, Tribhuvan University, who has provided guidance and constant encouragement throughout the course of my study and in the preparation of this dissertation with his extensive knowledge and professional expertise. Working under him during my entire post graduation has been an honour and a truly inspirational experience to cherish.

I am very grateful to Prof. Dr. Ram Krishna Maharjan, Head of the Department of Electronics and Computer Engineering at the Pulchowk Campus, for his precious assistance. I am thankful to our program coordinator Dr. Basanta Joshi for providing suitable platform to complete this task. I extend my thank from the bottom of my heart for his constant help, encouragement, and valuable suggestions.

I am also thankful to Prof. Dr. Shashidhar Ram Joshi, Prof. Dr. Subarna Shakya, Dr. Sanjeeb Prasad Panday and other faculties for their encouragement and precious guidance. Their wisdom and inside has helped me in completing this research smoothly. I am indebted to the Department of Electronics and Computer Engineering for incorporating this work as a part of our syllabus. It makes us the realization of our knowledge in the real-world applications and provides a benchmark to proceed for the thesis and research.

Last but not least, I would like to acknowledge with gratitude, the support of my parents, my wife Mrs. Muna Subedi, my brother Mr. Nishes Ghimire and my friends who have always been my pillars of strength and have shown enormous faith and confidence in me, which has kept me moving ahead.

Sincerely,
Niroj Ghimire
(075MSICE014)

## ABSTRACT

Fake news is becoming more readily available as technology advances, which sometime mislead the readers and leads to inaccurate social opinions. Fake news may be found on the Internet, news sources and social media platforms. The spread of low-quality news has harmed both individuals and society. In this thesis work, we analyze three hybrid models, CNN+simple RNN, CNN+GRU and CNN+BiLSTM in encoder decoder architecture to predict the stance between headline and article of the news. Pre-trained GloVe word embedding is used for word to vector representation as it can capture the inter-word semantic information. The CNN-RNN combination had been shown efficient in deep learning applications because they can capture sequential and local features of input data. The models were successfully trained and tested on both binary (ISOT) and multiclass (FNC-1) fake news datasets. It is found that the CNN+ BiLSTM model had better results than other two hybrid models in both binary and multiclass classification task for the fake news stance detection system.

***Keywords:*** Stance detection; Fake news; NLP;Hybrid Neural Network; Encoder decoder Architecture

## TABLE OF CONTENTS

## LIST OF FIGURES

## LIST OF TABLES

## LIST OF ABBREVIATIONS

| | |
|---|---|
| **AUC** | Area Under the Curve |
| **BiLSTM** | Bidirectional Long Short-Term Memory |
| **CNN** | Convolutional Neural Network |
| **Conv1D** | One Dimensional Convolution |
| **CSV** | Comma Separated Values |
| **FN** | False Negative |
| **FP** | False Positive |
| **FNC** | Fake News Challenge |
| **GloVe** | Global Vectors for word representation |
| **GRU** | Gated Recurrent Unit |
| **ID** | Identity |
| **ISOT** | Information Security and Object Technology |
| **LSTM** | Long Short-Term Memory |
| **LSVM** | Linear Support Vector Machine |
| **MLP** | Multilayer Perceptron |
| **NLP** | Natural Language Processing |
| **RNN** | Recurrent Neural Network |
| **ROC** | Receiver Operating Characteristics |
| **SVM** | Support Vector Machine |
| **TF-IDF** | Term Frequency Inverse Document Frequency |
| **TN** | True Negative |
| **TP** | True Positive |

CHAPTER 1

INTRODUCTION

## 1.1 Background

News is a very effective technique for disseminating information. It is an excellent source of information. It is also one of the most effective ways for individuals to communicate with one other and with the rest of the world. In the past, the common individual would wait until the next day to discover what had occurred in the world the day before. This is not the case in today's society, when news moves nearly at the speed of light. Data, increasing to become the richest asset that anyone can own, needs to be transmitted and exchanged and becomes much more important as becomes information. News and posts, both in physical and digital form, are one of the most common methods of knowledge sharing. With genuine knowledge helping to make human beings a more evolved species, the entire meaning will destroyed by circulations of fake news.

Fake News represents false news or propaganda comprising disinformation transmitted via classical media outlets like newspapers and TV in addition to modern media sources such social media [1]. In this day of modern technology, there is competition among individuals as who can create much more news, leading to the emergence of falsified news, also known as Fake News. There are two aspects to the idea of fake news: authenticity and intent. Authenticity refers to the fact that fake news contains incorrect material that can be recognized. Conspiracy theories, for example, are not included in fake news since they are difficult to establish true or false in most situations. The second, intent, refers to the fake content was written with the intention of deceiving the reader. Now, it is essential to know that what features may be utilized to categorize them.

Digital news is becoming more readily available to the people around the world, which leads to the spreading of hoaxes and misinformation online. Because access to the media has become so simple, and newsworthy events can occur anywhere on the universe, different news providers strive to make their stories appealing to as many people as possible via the Internet.As a result, news is rapidly transmitted to many people through a range of news sources, including as news channels, magazines, websites, and social networking sites. Fake news may be available on prominent platforms such as social media and the Internet, and it has the potential to confuse users. The word of the year was also dubbed "Fake news" by the Macquarie dictionary in 2016 [2] and Collin's dictionary word of the year 2017 (*www.collinsdictionary.com, cited on 2021 July 1*), taking into account the existence of this phenomena. Fake news, on the other hand, aims to persuade readers to believe incorrect information, making these articles complex to comprehend.

Digital news is created at a high and rapid rate,which is faster than traditional news. Machine learning has a great difficulty detecting fake news since it is updated every second. There have been a variety of solutions and attempts to spot false news, including neural network methods and natural language processing.
The processing area of natural language is changing from mathematical approaches to neural network methods. The language model demand to deal with the nuances tangled in conveying messages through the text in order to better categorization of the fake news. Detecting the fake news manually is also a very subjective and tedious task. Evaluating the veracity of a news article, also for professional experts, is a complicated. News, no longer only circulated via traditional media channels, but also through the new platforms of social media. So, the automated approach requires an understanding of the complicated and dynamic nature of natural language processing and generation. These makes the classification of text as fake news a challenging task. So, to overcome the automated fake news classification, it needs the task of stance detection. Based on multiple neural network experimental approaches, this study presents a neural network model that can detect false news by correctly predicting the relation between the title and the news content.

## 1.2 Motivation

Fake news has become a social issue, with people using it to spread false or rumor information in order to influence their behavior. It has been shown that the spread of false news had a significant impact on the 2016 US presidential election [3]. Fake news may be defined as entirely false or made-up material that is disseminated under the pretence of being real facts. Although the detection of misinformation in our daily lives is closely related to deception detection, in fact, it is far more difficult and complex.
Many civilizations across the world trust fake news and act on their beliefs without question. Content is readily created and rapidly transmitted through the social media, bringing in a significant number of data to analyze. The volume of online material, which covers a wide range of topics, increases the task complexity. Because computers alone cannot always determine the veracity or purpose of a statement, attempts must focus on collaboration between humans and technology. To address this, we believed that neural network model and natural language processing tools might be useful in resolving the issue.

## 1.3 Problem Statement

The introduction of the Internet opened the door for unprecedented levels of content transmission in human history. Consumers are producing and sharing more information than ever before because of social media platforms, some of it is false and has no basis in fact. The primary

goal of disseminating such information is to mislead or misinform the general public, to harm a person, firm, or nation's reputation, to cause confusion, and to profit financially or politically from sensationalism. The growing rise in the production and circulation of false news requires an urgent need for such news stories to be tagged and identified automatically. Automated identification of fake news, however, is a difficult task to achieve as it allows the model to comprehend nuances in natural language. It needs the ability of the model to consider how the published news is related or unrelated to the real news.
Fake news, now covers a wide area of cyberspace around the world. Identifying the vocabulary that is used to mislead readers is the essential task of identifying fake news. A difficult challenge is the concept of classifying fake news through learning word-level context. As a result, the purpose of this thesis study is to use stance detection approach to detect fake news. Given a set of news body and headline pair, stance detection is the task of automatic detection of relationship among pieces of text.

## 1.4 Objective

The objectives of this research work are:

- To analyze the performance of various hybrid neural networks in encoder-decoder architecture in predicting the stance of a news article and news headline pair.
- To present a hybrid neural network model for predicting the relationship between a news article and news headline pair.

## 1.5 Thesis Contribution

In this study, we present three distinct hybrid neural network language models that use Convolutional Neural Network and Recurrent Neural Network in an encoder decoder architecture to execute fake news classification tasks over pre-trained word vectors GloVe. The major contribution of this thesis is to obtain the suitable deep learning-based approach for detecting fake news by studying and experimenting with these three different hybrid models. We employed three different RNN cells (Simple RNN, GRU, and LSTM) as a follower of the pooling layer in CNN to reduce the loss of detailed local information and capture long-term relationships in sequences of phrases. We developed a network design that focuses on reducing network parameters while simultaneously capturing long-term relationships more accurately. Pre-trained word vectors [4] are used to initialize word embeddings, which are trained on a larger unsupervised collection of words. Initially, the proper dataset for training the three algorithms is chosen. The analysis are based on a thorough evaluation of the best model. The models are validated using an examination of the evaluation metrics on the testing dataset.

CHAPTER 2

LITERATURE REVIEW

## 2.1 Text Classification Methods

For the purpose of text classification, various kinds of algorithms have been developed over the time [5]. Classical machine learning technique and neural network model are the two primary categories of algorithms. Traditional text classification techniques such as Support Vector Machines (SVM), Naive Bayes (NB), and k-Nearest Neighbors (KNN) have been extensively explored and are still frequently used in the scientific community. However, deep learning architectures such as the Convolutional Neural Network (CNN), Recurrent Neural Network (RNN) and Hierarchical Attention Network (HAN) are becoming more attractive for text classification. Deep learning architectures, while their capacity to make excellent results in some cases, have some limitations also. So, before deciding whether to use deep learning or classical architecture for text classification, each specific case must be analyzed. The volume of data and the requirement for model interpretability are two significant factors in this selection. Deep learning methods typically require a lot more data than conventional machine learning algorithms.

## 2.2 Methods for Natural Language Processing

Text classification is a fundamental work in natural language processing that has a wide range of applications. Words are treated as discrete symbols in classical NLP systems, with the model based on a little amount of information on the relationship between the individual symbols. A fundamental baseline method is to describe a sentence structure as a bag-of-words and then build a linear classifier. The bag-of-words method, on the other hand, neglects all information on the ordering and semantics of the words [6].

N-grams models are another method for representing sentences. An N-Gram is a sentence made up of N-words. The number of words in the sequence is represented by the integer N. In the bag of words, only individual words are identified, and the order in which they appear in a document is not taken into account. The sequence of the words may be essential in NLP applications. The word embedding approach involves projecting words into a high-dimensional space, then combining the embeddings to create a fixed-size representation of the input phrase, which is then fed into the classifier. Words are treated as atomic symbols by the majority of rule-based language processing systems. This results in an extremely sparse vector representation of the vocabulary's size, with a single 1 at the current word's index point.

## 2.3 Fake News Detection System

In Natural Language Processing, stance detection is a well-researched topic. Most previous stance detection research based on text entity stance detection with respect to a subject or name. In the SemEval-2016 Task 6, [7], the purpose is to provide a stance label to a post that is pointing at a certain target using stance detection method. It presents a collaborative task for determining position in tweets: given a tweet and a target entity, automated natural language systems must determine whether the tweeter is in favor of, against or neutral. In another research on classifying online debate entries according to the speaker's perspective or opinion on a topic [8]. Many activities, such as misinformation detection [9], validation of claim [10], depend on stance detection. Another comparable study topic is Natural Language Inference (NLI), which is defined as the problem of evaluating if a natural language assumption can be derived from a natural language assumption [11].

The fake news stance identification challenge was released the dataset on February, 2017 as a public competition (*fakenewschallenge.org, cited on 2021 July 1*). The Emergent dataset served as the basis for the FNC-1 challenge dataset [12]. The stance is de-emphasized in FNC-1. In FNC-1, the stance is identified at the document level which classifies the whole news item in relation to a headline, . 'SOLAT in the SWEN' [13], which was built by the Talos Research Intelligence team, is the top-performing team in FNC-1, with an accuracy rate of 89.08 percent. A weighted average of CNN model and a gradient-boosted decision tree model were used to build their model. For classification, it employs pre-trained word2vec embeddings that are processed via CNN layers and softmax layer. The team that came in second place, 'Athene' employs a MLP algorith, which is made up of six hidden layers with crafted features.

Model that incorporates statistical, neural and external characteristics to provide the effective solution for this issue [14], use feature engineering methods to construct the neural embedding from the recurrent model.They use handcrafted external characteristics from the recurrent model and statistical features from the n-gram bag of words model. Finally, a deep neural network is used to combine all of the features, resulting in a categorization of the headline-body news combination as unrelated, discuss ,agree or disagree. With cosine similarity input into a deep neural network, Tf-Idf on unigrams and bigrams obtained an accuracy of 94.31% [1]. Previous research on fake news identification has concentrated on target-specific stance prediction, which determines the stance of a text entity that is connected to a subject or a named entity.

A news article is made up of a collection of words. As a result, several authors have proposed using text mining and machine learning methods to evaluate news textual data in order to forecast news authenticity in the past. In classical news sources, fake news detection depends mostly on news material, although additional social background auxiliary data can be used in social

media with additional information for assist in fake news detection. We would then present the details of how important qualities can be extracted and viewed from the content and social value of news [15]. Many of the current fake news detection methods are highly relied on feature extraction. Authors also suggested methods in [16],[17] and [18] which are feature extraction machine learning based model. The changeover of original data into a data set containing the most relevant information with a limited number of variables is the extraction of features[19].
CNNs have excelled in speech recognition, computer vision and NLP applications because they use convolutional filters based on the input phrase feature matrix to extract n-gram features at sequence locations, output a high-level word feature to the next layer, and train short range connections using pooling layers. In NLP, CNNs have had a lot of success with sentiment classification [20]. RNN stands for "recurrent neural network," which is made up of multiple cells with the same layout that are controlled by individual weights to utilize the "input gate" and "output gate". To avoid the problem of gradient inflating or disappearing in normal RNNs, LSTM are commonly used.
CNN-LSTM model that is divided into three parts: Encode, Attention Rearrange, and Decode, with a focus on the impact of word order on sentiment classification [21]. The CNN and LSTM are two typical models for natural language classification tasks, and they frequently employ word embedding as a model input. For fake news detection, CNN-RNN hybrid model was successfully developed and verified, resulting in detection results that were considerably superior than non-hybrid baseline systems [22].
Detecting fake news has been the subject of numerous investigations. The ISOT Fake News Dataset is another dataset, that comprises both real and fake news, published by ISOT research lab, University of Victoria. On the ISOT dataset, n-gram models and six machine learning approaches to detect false news is utilized [23]. LSVM as a classifier and for TF-IDF as a feature extraction approach produce the best results, with an accuracy of 92%. Another study using capsule network of parallel convolutional layers found that accuracy was 99.8% when utilizing non-static word embeddings and varying levels of n-grams [24].

# CHAPTER 3

# METHODOLOGY

The methodology for the fake news stance detection consists of hybrid neural network in encoder decoder architecture. The methodology for the study is shown in Figure 5.16. For analysis of binary and multiclass classification in the fake news detection, two datasets were chosen. First dataset is FNC-1 dataset, which consists of four labels as the relationship between headline and article in the news and another is ISOT dataset, with the two labels as fake and real news. The study is based on the text features contain in the headline and article, to find the stance between them. The headline and news content, need a data preprocessing steps to remove unnecessary words and symbols. The preprocessed words are applied to GloVe word embedding to represent word in numerical vector with capturing their semantic and syntactic information. I create three distinct hybrid neural network models for the thesis research, each with a separate encoder and decoder network that employs a combination of CNN and RNN. The initial CNN network with convolutional, max pool layer, and afterwards RNN layer are used in both the encoder and decoder.

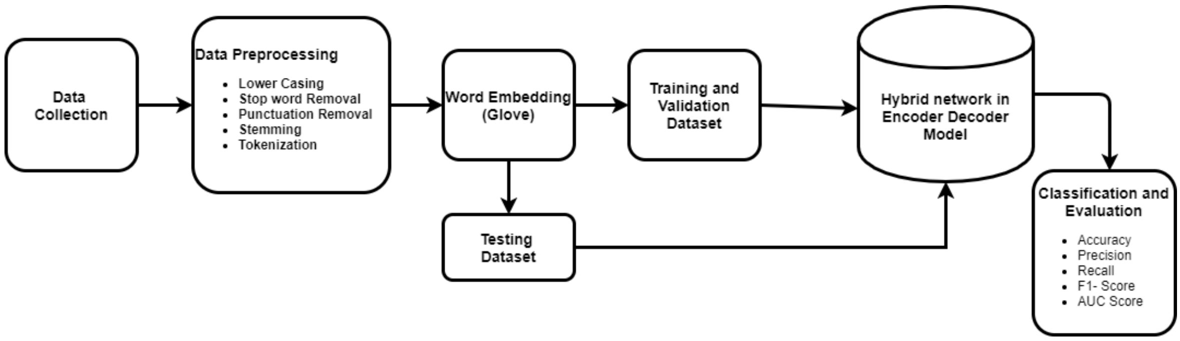



Figure 3.1: Hybrid network in Encoder Decoder for Fake news Stance Classification

To comprehend the meaning of a phrase in natural language, we must first learn how we compose a sentence and how we convey our thoughts using various words. Word vector representation is a method of generating a numerical vector for words that represent the meaning in the words. The output from embedding layer is fed to the convolutional layers which read the inputs as a one-dimensional sequential data. The role of convolutional layer is to automatic feature extract from the input fed into the network; for example, if the network is given an embedding vector of texts, the convolution layer identifies the association between neighboring texts and stores the values in vectors. The outcomes of reading are projected onto a filter map that captures the input interpretation. The pooling layers employ maximum pooling operations to derive the most important features from the generated feature maps. The pooling layer's output is sent to the

RNN layer, which learns long-term dependence and sequential information. RNNs have a unique characteristic that sets them apart from other deep neural networks that they include feedback mechanisms. Three distinct models used as hybrid network in both encoder and decoder are, CNN+SimpleRNN, CNN+GRU and CNN+BiLSTM.

## 3.1 Dataset Collection

### 3.1.1 FNC-1 Dataset

Fake News Challenge publishes the FNC-1 dataset, as the initial step for Fake News Detection task for the public competition. Emergent Dataset provided the data for this competition. The dataset contains the news story article, the news headline, and the stance label for the relationship between the news body and the headline pair. The data include 75,385 different pairs of news articles and headlines. There are 4 different category of stance: Unrelated, Discuss, Agree and Disagree. The FNC-1 dataset contains higher number of pair of headline and news article in unrelated stance label and lower number of pairs in disagree stance label. So, it is an unbalanced dataset.

Table 3.1: Stance category in FNC-1 dataset

| **Stance** | **Description** |
|---|---|
| Agree | Headline agrees with the news article. |
| Disagree | Headline disagrees with the news article. |
| Discuss | Headline addresses the same subject as the news, but does not agree completely. |
| Unrelated | Headline does not discuss the topic as news |

### 3.1.2 ISOT Dataset

The ISOT Fake News Dataset is another dataset, that comprises both real and fake contents, published by ISOT research lab, University of Victoria [25] . The real articles were taken from reuters.com, a well-known news website, whereas the false stories came from a variety of sources, primarily from websites identified by politifact.com. The dataset comprises a variety of articles on various themes, however the majority of the articles are about politics and world events. The data include 44,898 different pairs article title, article text, date and article label(real or fake).

Table 3.2: An Example of FNC-1 Dataset

| Duncan Hunter makes unconfirmed claim Border Patrol caught at least 10 ISIS fighters | |
|---|---|
| Agree | Federal officials have declared a sitting US Representative a liar after he claimed on national television that ISIS terrorists were crossing the border from Mexico. Rep Duncan Hunter (R-CA), who told Fox News on Tuesday that insurgents were crossing the Texas border, was spreading a lie, according to the Department of Homeland Security.[..] |
| Disagree | There are contradicting reports of ISIS being near the border, but there are reasons not to believe the reports.[..] |
| Discuss | Border Patrol officers have apprehended at least ten ISIS militants attempting to enter the United States from Mexico, according to Rep. Duncan Hunter.[..] |
| Unrelated | According to The Wall Street Journal, Apple is changing its retail approach for the Apple Watch to provide a more customized purchasing experience.[..] |

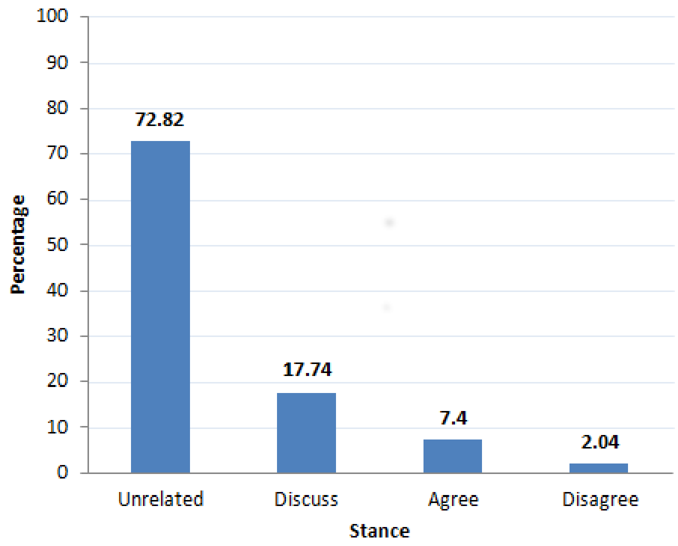



Figure 3.2: Labels distribution in FNC-1 Dataset

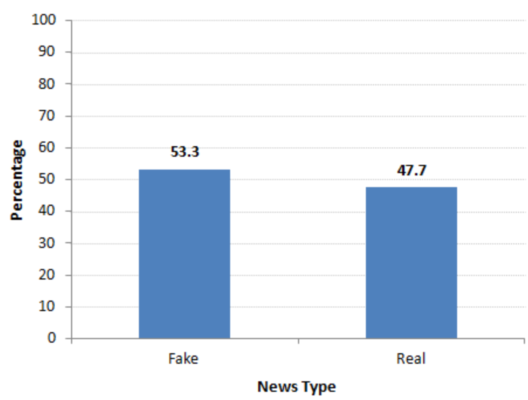



Figure 3.3: Labels distribution in ISOT Dataset

## 3.2 Data Preprocessing

Data preprocessing is commonly the basic step in a Natural Language Processing (NLP) system, based on key for the system's ultimate performance [26]. In order to apply deep learning methods on them, text data requires preprocessing. When dealing with text in Natural Language Processing, text cleaning or text pre-processing is an important initial step. Different approaches are used to convert text data into a form that is appropriate for modeling. For dimensionality reduction, text preprocessing steps will be use. Both the headlines and the news stories have to apply the data preprocessing measures.

### 3.2.1 Tokenization

Tokenization is the step which take the sentence and breaking it down into words. Using white spaces and punctuation symbols as delimiters, the text is broken down into single words or tokens. We must first define the words that make up a string of characters before process the natural languages. As a result, tokenization is the most fundamental step in the NLP process (text data). This is significant since the text's meaning can be easily deduced by examining the words in the text.

### 3.2.2 Lower Casing

Lower casing is the step of changing to the lower case of a word in English language. Words like BOOK and book have the same meaning, but in the vector space model, they are represented as two separate words if they are not transformed to lower case, which results in more dimensions. Lowercasing is one of the most basic and effective text preprocessing technique, which also greatly improves desired output consistency.

### 3.2.3 Stopword, Punctuation Removal and Stemming

The common words in any natural language are stopwords. These stopwords can not add much value to the context of the document when interpreting text data and constructing NLP models. Stop words include articles, prepositions, conjunctions, and certain pronouns. Some common words in a document are "the", "is","in","about","where","at","to","at" and so on. The concept is to simply eliminating the terms that appear in all of the corpus documents. The punctuation in the text adds very less or no value to the information. When punctuation is added to any word, it becomes difficult to distinguish it from other words. So, in text preprocessing, we also eliminate the punctuations.

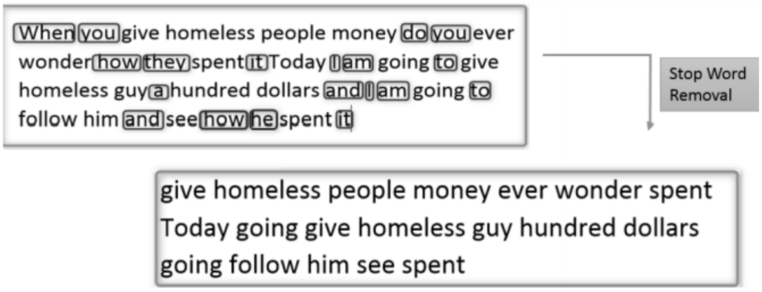



Figure 3.4: An example for Stopword Removal in data preprocessing

Stemming is the act of eliminating prefixes and suffixes from text so that words may be reduced to their stem base or root. For example, the terms "operated," "operating," and "operates" all have the same stem base "operat."

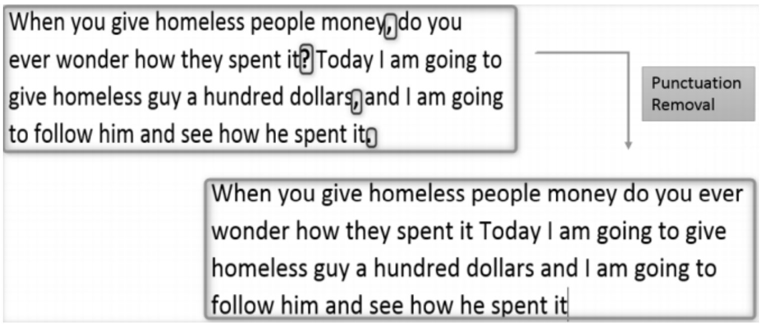



Figure 3.5: An example for Punctuation Removal in data preprocessing

## 3.3 Word Embedding

Word vector representation is use to give the numerical vector for words that can represent meaning of word. It is very difficult to use the text to model from the body and title of the news story. So, this includes translating raw text into computational functions in order to achieve text analytics. Text vectors or word embedding representations are sometimes referred to as the method of describing word as vectors. The embedding matrix is a technique to express the embeddings for each of the vocabulary terms. The Euclidean distance between two word vectors is a useful tool for determining semantic closeness between the words. This measure can sometimes discover unique but relevant words that are outside the vocabulary of the ordinary human. The dimensions of the word embedding space are represented by rows, while the words in the vocabulary are represented by columns. The dataset provided for the challenge is extremely limited when compared to the size of article posted every day. As a result, a model trained on such a limited vocabulary could underperform on a dataset that has never been seen before. So, it has to use pre-trained word embedding techniqes over broad datasets to allow our model to account for new vocabulary in test data.

GloVe is an unsupervised learning algorithm that generates word vector representations. This is

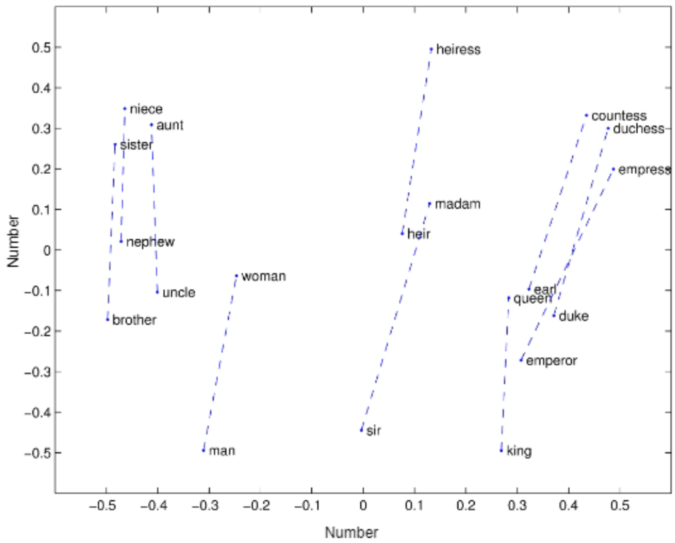



Figure 3.6: An example for linear visualization of GLoVe Embedding

achieved by mapping phrases into a meaningful space in which the semantic similarity of words equals the distance between them. The objective of using word embeddings are;

- To reduce dimensionality.
- To use a word to predict the words around it.
- To capture the inter-word semantics.

## 3.4 Encoder Decoder Architecture

Encoder-decoder models have significantly pushed the state of art performance on a range of natural language related tasks, such as text summarization [27] and machine translation [28]. The Encoder Decoder architecture is the most popular architecture used to create sequence to sequence(seq2seq) models. A seq2seq model, initially introduced by Google, seeks to map input with output when the lengths of the input and output may change [29]. The most common method for training such a model is to use all of the samples in the training data to update all of the model parameters (across many epochs). The encoder decoder model has the potential to train a single end-to-end model directly on input and target phrases, as well as the capacity to handle variable length input and output text sequences. The encoder, context vector and decoder are the three components of the model.

- Encoder and decoder, both consists of the first CNN network with convolution and max pool layer and then RNN layer.
- The encoder takes the input sequence and summarizes it in internal state vectors. We preserve the internal states of the encoder.

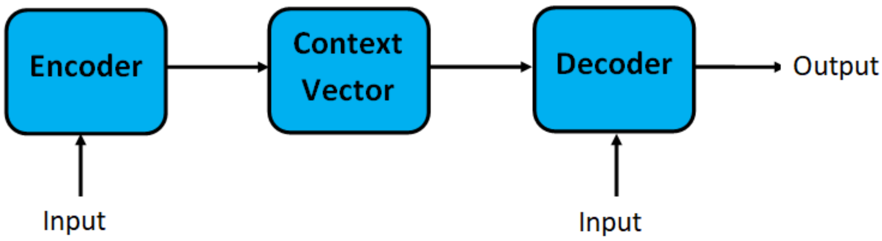



Figure 3.7: Encoder decoder architecture for sequence to sequence model

- The starting states of the decoder module are set to the encoder module final states. The decoder begins producing the output sequence using these initial states and input sequence to the decoder.
- So, the encoder converts the input sequence into context vectors (also known as thought vectors), which are then fed to the decoder, which begins producing the output sequence based on the context vectors.

**Encoder**

A stack of neural units, each of which receives a single input sequence element and propagates information about that element.

**Context Vectors**

Context vector is the final hidden state, generated by the encoder. This vector tries to capture all input element information in order to support the decoder in making correct predictions. Context vector functions as the initial hidden state for the decoder.

**Decoder**

Each recurrent unit receives a hidden state from the preceding unit and generates both an output and its own hidden state.

## 3.5 Baseline Models

### 3.5.1 Convolutional Neural Network (CNN)

The major benefit of CNN is that it incorporates automatic feature extraction and weight computation throughout the training process. Instead of manually implementing the feature extractor, CNN employs it in the training phase. The CNN feature extractor is made up of special types of neural networks that determine the weight of connections between the nodes throughout the training phase. CNN models can be use to extract features in text classification, with filters of various lengths that are used to convolve the text matrices. The extracted vectors of each filter are then operated using max pooling layer. CNN uses filters of a certain length to convolve and abstract word vectors from the original document, resulting in convolved abstract sequences from original word vectors.

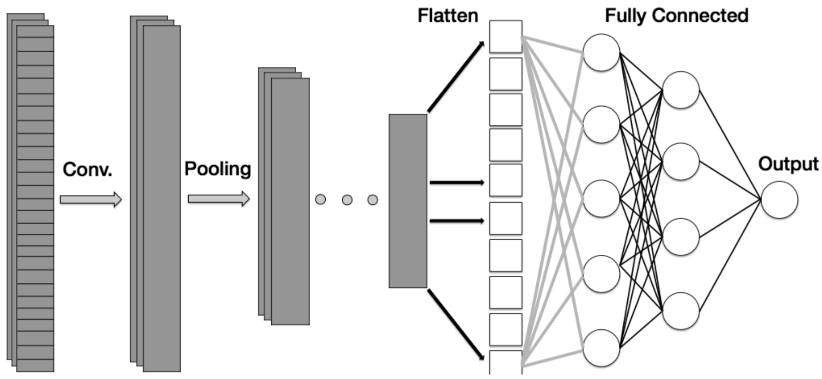



Figure 3.8: Different Layers in Convolution Neural Network

Conv1D neural network is commonly used in document classification or NLP. In Conv1D, word vectors are represented by one-dimensional arrays. In a CNN, a fixed-size window filter iterates over the training results, multiplying the input with the filter weights at each stage and producing an output that is stored in an output sequence. The feature map or output filter of the data is represented by this output array. The number of filters determines number of feature maps will be employed, while the kernel size determines the size of the filter. CNN can be used in this context to learn local characteristics that are taken from the training samples.

Given a list of words $w_{1:n} = w_1,...,w_n$, each of which are represented with a d-dimensional

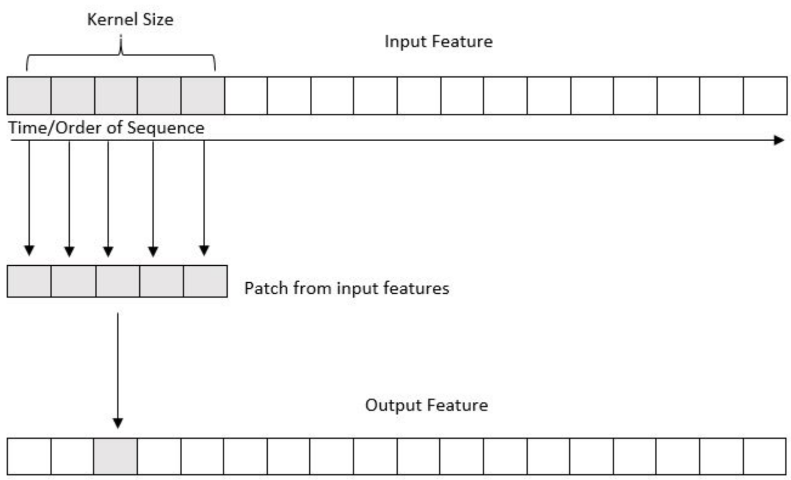



Figure 3.9: 1D Convolution

embedding vector. The output of moving a sliding-window of size k across the phrase and applying the same convolution filter or kernel to each window in the sequence is a 1D convolution of width-k. When considering a window of words $w_i$,.....,$w_{i+k}$, the $i^{th}$ window's concatenated vector is:

$$x_i = [w_i, w_{i+1}, \ldots.., w_{i+k}] \in R^{k \times d} \tag{3.1}$$

Each window is subjected to the convolution filter, which yields scalar values $r_i$, each for the $i^{th}$ window:

$$r_i = g(x_i \cdot u) \in R \tag{3.2}$$

In general, multiple filters are generally used, $u_1$,...,$u_l$ which may then be represented as a vector multiplied by a matrix U and a bias term b:

$$r_i = g(x_i \cdot U + b) \tag{3.3}$$

with $r_i \in R^l$, $x_i \in R^{k \times d}$, U$\in R^{k\dot{d} \times l}$ and b$\in R^l$.
The pooling function joins the vectors from several convolution windows into a single 1-dimensional vector. The pooling operation is done once again by taking the maximum or average value seen in the convolution result vector. This vector encapsulate the most important features of the document and is then sent down in the network, giving rise to the concept that convolution network is a good feature extractor.

### 3.5.2 Recurrent Neural Network (RNN)

Recurrent Neural Network (RNN) is a type of neural network that focuses on utilising sequential information. It processes input through a feedback loop, allowing the networks to remember what they have already done. RNN, in comparison to other neural networks, execute the same function for each element of a sequence, but the network is recurrent, meaning the output data is sent back into the network. Recurrent Neural Networks are particularly well suited to situations in which the sequence is more essential than the individual elements. The output of a recurrent neural network is determined not only by the current inputs, but also by the neuron state of the preceding phase. For sequential data, recurrent neural networks are common because each unit can remember the state of the previous unit. This is especially useful in natural language processing because it aids in language comprehension.

Deep learning is concerned with the development of algorithms that refine themselves over time. The learning algorithm's task is to construct a feature from the training data. RNNs are complex systems with an internal state at each classification level. RNN, on the other hand, has trouble retrieving information from the distant past due to the long-term dependence problem. For each timestamp, it employs a backpropagation technique known as back-propagation over time (BTT). The vanishing gradient problem and the expanding gradient problem are problems with RNN.

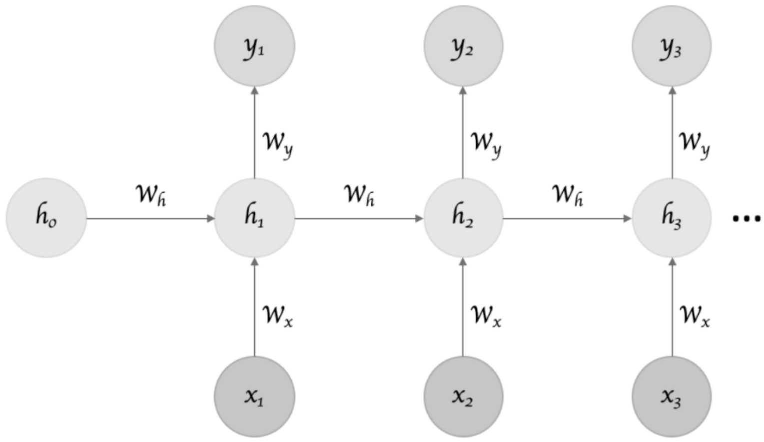



Figure 3.10: Structure of Recurrent Neural Network

**Long Short-Term Memory (LSTM)**

LSTM networks are a revised form of RNN. LSTM use the combine method of real time recurrent learning and backpropagation through time. LSTMs are explicitly designed to avoid the vanishing gradient problem and the expanding gradient problem in previous recurrent neural network. LSTMs have chain-like structure like in classical RNN, but the repeated module is distinct.

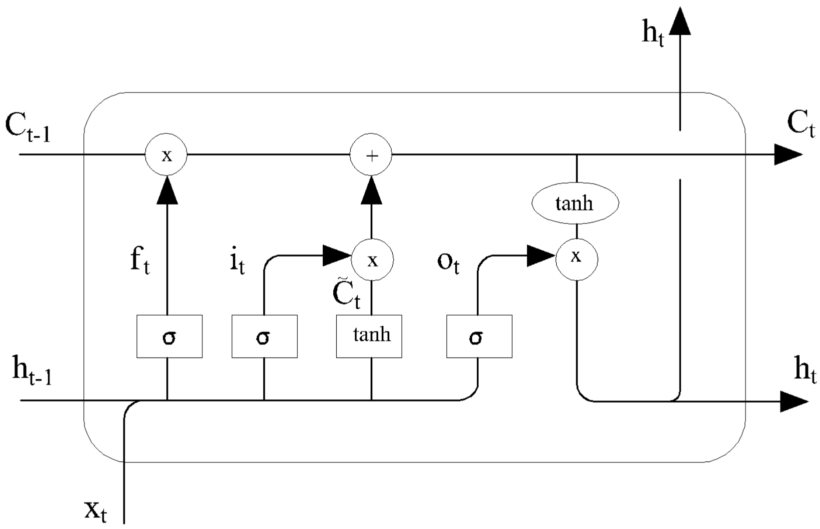



Figure 3.11: Repeating module in LSTM

There are three different gates in an LSTM cell:

a) **Forget Gate**: The first stage in the LSTM algorithm is to determine which data should be erased from the memory cell. This choice is made by the forget gate sigmoidal layer. This layer computes a number between 0 and 1 for each number in the memory cell $C_{t-1}$ based on the previous state $h_{t-1}$ and the current input $x_t$.

$$f_t = \alpha(w_f \cdot [h_{t-1}, x_t] + b_f) \tag{3.4}$$

b) **Input Gate**: The input layer determines the amount of data kept in memory. There are two steps to this technique.

   i) The sigmoidal function determines which values are permitted to pass.

$$i_t = \alpha(w_i \cdot [h_{t-1}, x_t] + b_i) \tag{3.5}$$

   ii) A tanh function creates a vector of new candidate values $C'_t$ and weights the values supplied to it.

$$C_t^{'} = tanh(w_c \cdot [h_{t-1}, x_t] + b_c) \tag{3.6}$$

The previous cell state, $C_{t-1}$, has been replaced by the new cell state, $C_t$.The updated equation can be given as,

$$C_t = f_t \times C_{t-1} + i_t \times C_t^{'} \tag{3.7}$$

c) **Output Gate**: The sigmoidal function is used to concatenate information once more. Then the tanh function is used to transmit information from the memory cell. The point-by-point operation is used to acquire the output.

$$o_t = \alpha(w_o \cdot [h_{t-1}, x_t] + b_o) \tag{3.8}$$

$$h_t = o_t \times tanh(C_t) \tag{3.9}$$

**Bidirectional LSTM**

Using the unidirectional sequence model, even though we already know the whole text, it only can capture the information from beginning to end direction of words. But it would be better if the model could also predict based on the future terms, allowing it to solve the problem more efficiently. So, here comes the concept of bidirectional LSTM (BiLSTM).

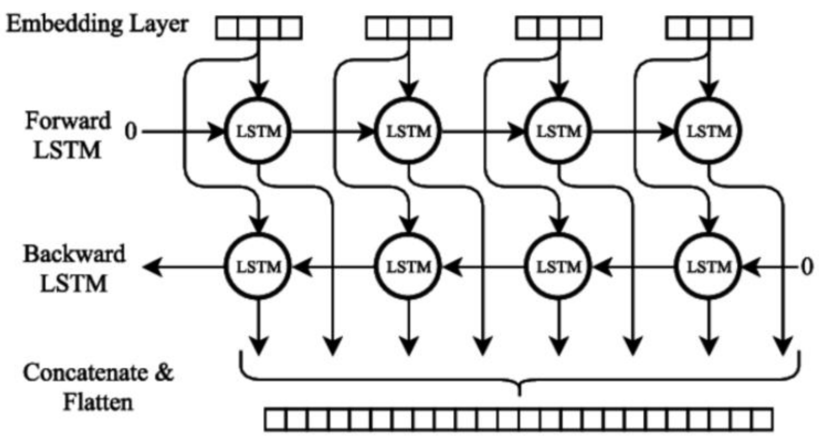


Figure 3.12: Structure of Bidirectional LSTM (BiLSTM)

There are two types of connections in a BiLSTM, one moving forward in time to learn from previous representations and the other flowing backwards in time to learn from future representations. As a result, the first LSTM layer's input is fed in normal time order, whereas the second LSTM layer's input is a reversed replica of the input sequences. At each time step, the outputs of the two networks are normally concatenated. This gives the network more context, resulting in faster and more complete learning.

**Gated Recurrent Unit (GRU)**

GRUs are also a more advanced version of a recurrent neural network. GRU employs the update gate and reset gate to address the vanishing gradient problem of a conventional RNN.

a) **Update Gate**: The update gate aids the model in determining how much historical data (from earlier time steps) should be passed on to the future. This is highly useful since the model may choose to replicate all of the data from the past, eliminating the possibility of vanishing gradients.

$$z_t = \alpha(w_z \cdot [h_{t-1}, x_t] + b_z) \tag{3.10}$$

b) **Reset Gate**: Reset gate is utilized by the model to determine how much knowledge from the past should be forgotten. We use the formula given in equation 3.11 to find the result of reset gate:

$$r_t = \alpha(w_r \cdot [h_{t-1}, x_t] + b_r) \tag{3.11}$$

## 3.6 Hybrid Neural Network

Three different hybrid neural network models that uses combination of CNN and RNN in a separate encoder and decoder networks is implemented in Python and their performance is trained

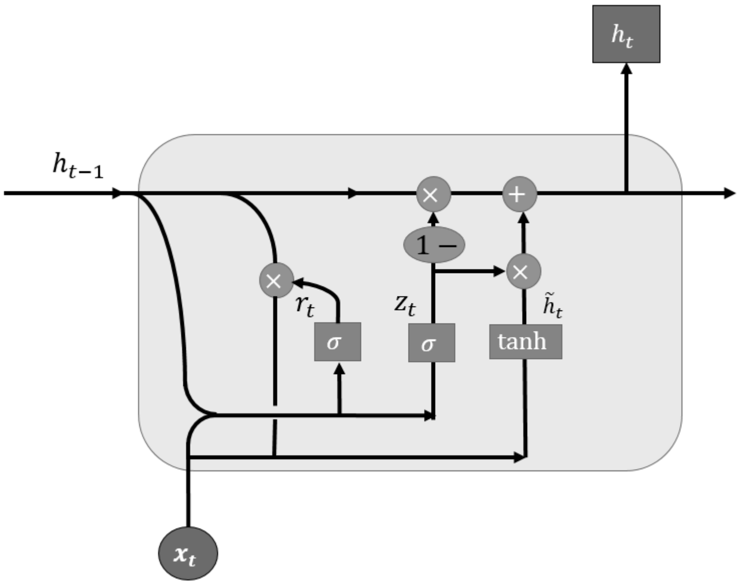



Figure 3.13: Repeating module in Gated Recurrent Unit

and evaluated on two fake news datasets.

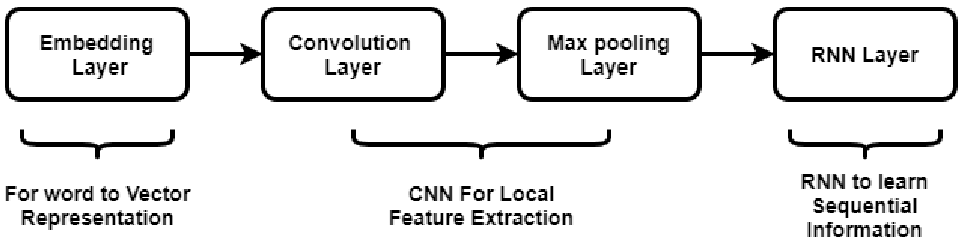



Figure 3.14: Hybrid (CNN RNN) Neural Network For Fake News Stance Detection

The CNN-RNN combination has been shown efficient in deep learning applications because they can capture both sequential and local features of input data. They have been used to recognize emotion [30] and sign language detection from video streams[31], due to their capacity to learn scene characteristics with the CNN and sequential features with the RNN. In NLP tasks, RNN helps to learn temporal and context characteristics from text, as well as capture long-term relationships between text entities, which are recognized using CNN's spatial relations capability.

## 3.7 Evaluation

In order to determine the consistency and correctness of a classification model, typical assessment metrics are determine according to,

a) True Positive (TP): A true positive is when the model predicts the positive class accurately.

b) False Positive (FP): A false positive occurs when the model estimates the positive class inaccurately.

c) True Negative (TN): A true negative is when the model predicts the negative class accurately.

d) False Negative (FN): A false negative is when the model predicts the negative class inaccurately.

### 3.7.1 Confusion Matrix

A confusion matrix is a table that shows the results of the prediction model. The number of observations made by the model where it categorized the groups correctly or incorrectly is expressed by entry in a confusion matrix. The confusion matrix has peculiar table organization which helps the output to be visualized, usually supervised learning. It shows not only a predictive model's results, but also which groups are correctly predicted, which are incorrectly predicted, and what types of errors are being made.

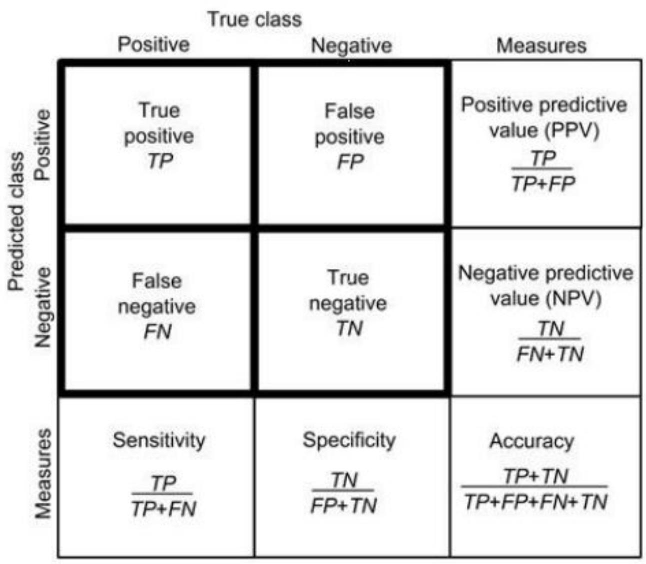



Figure 3.15: Confusion matrix and Classification Measures

We have four stances agree, disagree, discuss and unrelated stances. Each cell corresponds to the number of corresponding true label and predicted label. An example of confusion matrix for a four-label fake news stance classification system is shown in Figure 3.16. The vertical side represents the true label of the stance from the testing dataset and horizontal side represents the predicted label using the neural network model. The highlighted cell symbols for the true prediction of each stances. The true prediction is that which dataset has the same true label and predicted label. So, the confusion matrix is also helpful to calculate the evaluation metrics (accuracy, precision, recall and F1- score).

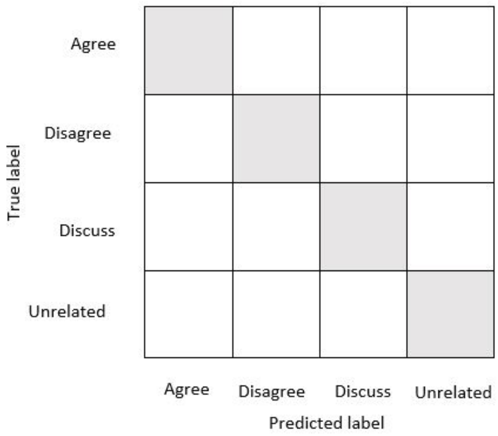



Figure 3.16: Confusion matrix for fake news detection system

### 3.7.2 Classification Evaluation Metrics

a) Accuracy: The accuracy of a model is a metric that sums up how well it performs across all classes. When all of the classes are equally essential, accuracy is beneficial. The ratio between the number of right predictions and the total number of predictions is used to compute it.

$$Accuracy = \frac{Total\ no.\ of\ accurate\ predictions}{Total\ no.\ of\ predictions\ made} \tag{3.12}$$

b) Precision: Precision is defined as the ratio of properly predicted positive observations to total expected positive observations.

$$Precision = \frac{TP}{TP + FP} \tag{3.13}$$

c) Recall: Recall is defined as the proportion of properly predicted positive observations to all observations in the actual label.

$$Recall = \frac{TP}{TP + FN} \tag{3.14}$$

d) F1-Score: The harmonic mean between recall and precision is the F1 Score. As a result, this score takes both false positives and false negatives into account. F1-score is more useful than accuracy when the class distribution is unbalanced.

$$F1 - Score = \frac{2\ \times\ Recall\ \times\ Precision}{(Recall + Precision)} \tag{3.15}$$

### 3.7.3 ROC-AUC Curve

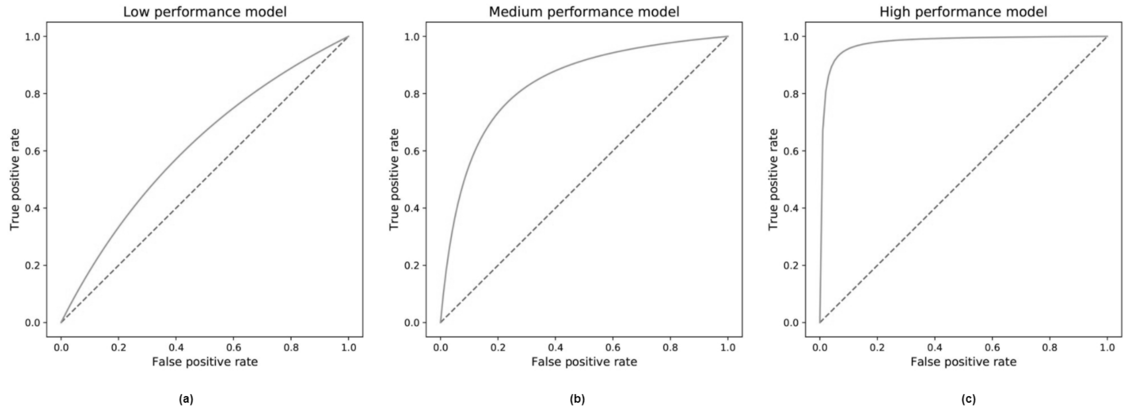



Figure 3.17: An example of ROC-AUC Curve: (a) Low performance model (b) medium performance model (c) high performance model

ROC-AUC curve helps to see how the threshold plays out the decision of the model at different threshold settings. Our model will predict the True Positives and Negatives and all other metrics mentioned above based on the assumption of a threshold that dictates what output label is considered positive and negative. This value of the threshold however could have been any random number hence this is an arbitrary choice and should not affect the decision provided by our model. It provides us the measure of goodness of fit, summarizes the model output across all thresholds, and provides a good sense of the discriminative power of a given model.

CHAPTER 4

IMPLEMENTATION AND DEVELOPMENT

## 4.1 Hybrid Models

For this thesis study, we make three different hybrid neural network models that uses combination of CNN and RNN in a separate encoder and decoder networks. The encoder and decoder, both consists of the first CNN network with convolutional, max pool layer and then RNN layer.

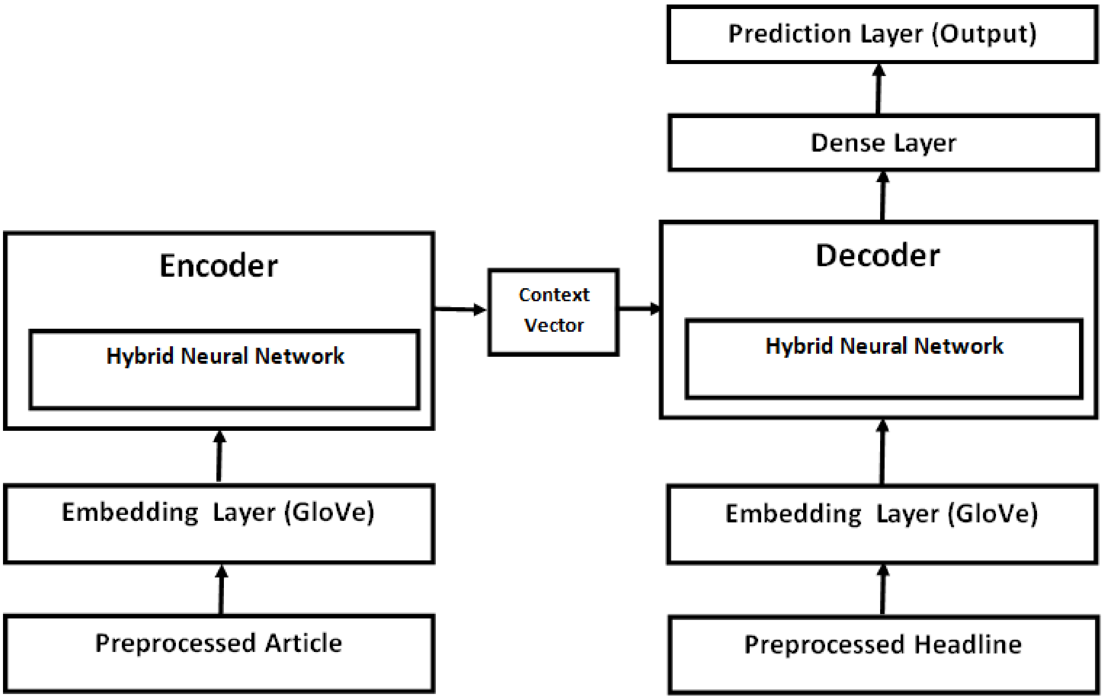



Figure 4.1: Neural Network Architecture for Fake News Stance Detection

CNN is a kind of neural network that processes data in the form of vectors. The convolutional layers read the inputs as a series of one-dimensional data from the embedding layer. The function of convolutional layer is to extract features from the data being fed into the network - for example, if the network is provided with an embedding vector of texts, the convolution layer extracts the relationship between the neighboring texts and stores the values in vectors. The reading's outcomes are projected onto a filter map that captures the input's interpretation. The pooling layers work on the obtained feature maps and use maximum pooling techniques to derive the most important features. RNNs have a unique feature that separates them apart from other deep neural networks that they include feedback loops. The output of the pooling layer is passed to the RNN layer, which uses the input data to learn long-term dependency. Three distinct models used as hybrid network in both encoder and decoder are, CNN+SimpleRNN, CNN+GRU and CNN+BiLSTM.

The encoder takes the input sequence from the article body embedding vector and summarizes it in internal context vectors, using hybrid neural network model. Context vector is the final hidden state, generated by the encoder. This vector tries to capture all input element information in order to support the decoder in making correct predictions. In our study, we employ the decoder to take inputs as context vector from encoder and word embedding vector of the news headlines from embedding layer. The decoder begins producing the output sequence based on the context vectors from encoder and headline embedding vectors. So, the decoder tries to find the relatedness between the news article and headline. The dense layer is that each layer's neurons are coupled to those in the previous layers. The hidden units uses the relu as an activation function and prediction layer uses the softmax activation function. Softmax function normalizes the output of a network into a probability distribution of each of possible output.

Google Colab is used to implement the deep learning model. Colab is a cloud environment for Jupyter notebooks that includes GPUs and TPUs for high computation. The experimental code is written using Python programming. The Keras and Tensorflow Python library is used to implement the hybrid CNN-RNN model in encoder decoder architecture. For reading datasets and processing arrays, respectively, the Pandas and Numpy libraries are utilized. For the data preprocessing task, NLTK package is used. The Scikit-learn package is used to analyse the data, evaluate the results. For graph visualization, the Matplotlib is used.

### 4.1.1 Model 1(CNN+ Simple RNN)

The combined CNN+ Simple RNN models for fake news stances detection are shown in Figure 4.2. The network has separate encoder and decoder network. Both encoder and decoder consists for the hybrid CNN+ Simple RNN model, which includes 1D convolutional layers followed by pooling layer and simple RNN layer.

In the encoder side, the preprocessed news article is fed to the GloVe embedding layer for the word to vector representation. Embedding layer gives the 100 dimensional vector representation for the each tokens in the headlines and news body, then creating a embedding matrix. The vectorized data then passed to the convolutional layer. The convolutional layer is used for local feature extraction that is activated by the relu activation function. The max pooling layer is used for dimensionality reduction of the given input text. The function map is transferred to the simple RNN layer to extract sequential information. Similarly, in decoder side, the preprocessed headline fed with the embedding layer, Convolutional layer, max pool layer and simple RNN layer. The decoder Simple RNN layer takes the encoder context vector and headlines processed vector. The output vector from the decoder connected to the fully connected layer of 50 nodes to predict whether they belong under any of the given categories. In this experiment, FNC-1 dataset

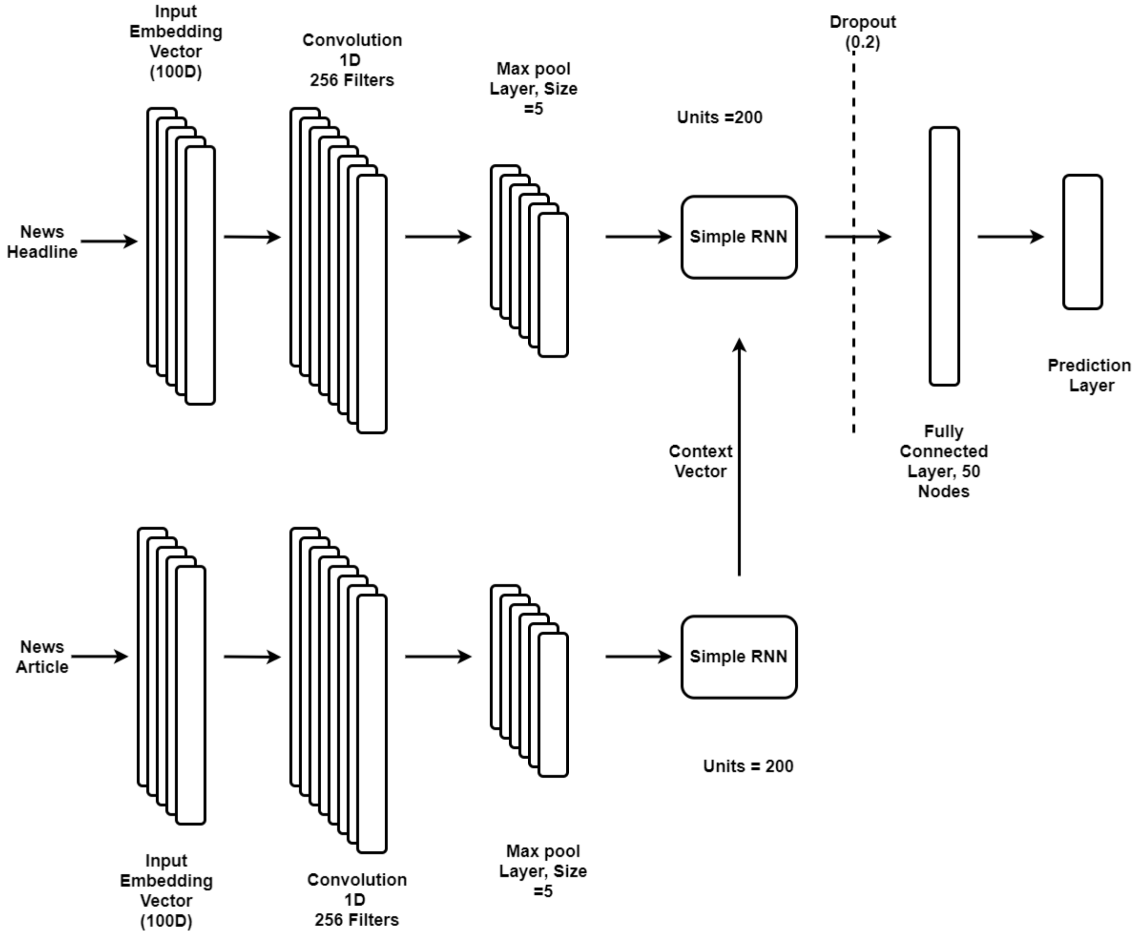



Figure 4.2: Model 1 (CNN+Simple RNN) for fake news stance detection

has four output labels and ISOT has two classes. So, in the prediction layer of FNC-1 and ISOT dataset has four and two number of nodes respectively.

To train Model1 (CNN+ Simple RNN), we used a learning rate of 0.001 which is the default learning rate value for rms prop optimizer in Keras. A dropout value of 0.2 was applied in the after the decoder RNN layer so that the model does not over fit the input data. The batch size, the number of training samples used by the optimizer before updating the model parameters, of 64 datasets was used.

Table 4.1: Parameter and values setting for Model 1 (CNN+ Simple RNN)

| **Parameter** | **Experimental Range** | **Selected Value** |
|---|---|---|
| Headline word lenth | - | 20 |
| Article word lenth | - | 400 |
| GloVe Word Embedding dimension | - | 100 |
| Encoder CNN Filter size | 128,256 | 256 |
| Encoder CNN Kernel Size | 3,5 | 5 |
| Decoder CNN Filter size | 128,256 | 256 |
| Decoder CNN Kernel Size | 3,5 | 5 |
| Maxpool Size | 3,5 | 3 |
| Encoder Simple RNN Unit | | 200 |
| Decoder Simple RNN Unit | | 200 |
| Dense Layer Neurons | 50, 100 | 50 |
| Hidden layer Activation Functions | - | Relu |
| Output layer Activation Functions | - | Softmax |

### 4.1.2 Model 2 (CNN+GRU)

The combined CNN+ GRU models for fake news stances detection are shown in Figure 4.3. It has separate encoder and decoder circuit on the network. The hybrid CNN+ GRU model contains 1D convolutional layers, a pooling layer, and a GRU layer in both the encoder and decoder. The preprocessed news article is sent to the GloVe embedding layer for word to vector representation on the encoder. The embedding layer creates an embedding matrix by giving each token in the headlines and news body a 100-dimensional vector representation. The vectorized data then passed to the convolutional layer. The relu activation function activates the convolutional layer, which is utilized for local feature extraction. The max pooling layer is used to reduce the dimensionality of the input text. To extract sequential information, the function map is passed to the GRU layer.

Similarly, the embedding layer, Convolutional layer, max pool layer, and GRU layer are applied to the preprocessed headline in the decoder. The encoder context vector and headlines processed vector are sent to the decoder GRU layer. The decoder's output vector is connected to a fully connected layer of 50 nodes to predict whether they belong under any of the given categories. In this experiment, FNC-1 dataset has four output labels and ISOT has two classes. So, in the prediction layer of FNC-1 and ISOT dataset has four and two number of nodes respectively.

To train Model 2(CNN+ GRU), we used a learning rate of 0.001 which is the default learning rate value for rms prop optimizer in Keras. A dropout value of 0.2 was applied after the decoder GRU

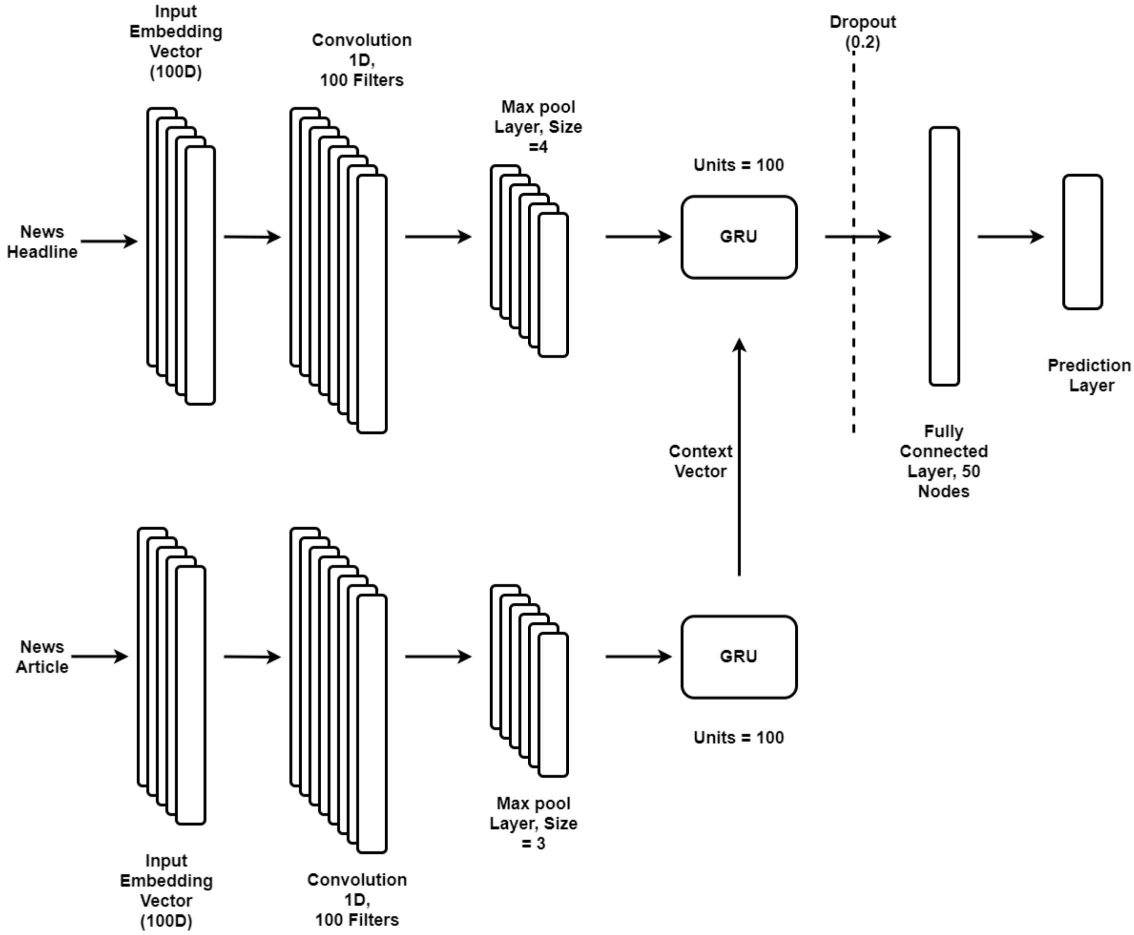



Figure 4.3: Model 2 (CNN+GRU) for fake news stance detection

layer so that the model does not over fit the input data. The batch size, the number of training samples used by the optimizer before updating the model parameters, of 64 datasets was used. Softmax is a activation function use to classify multiclass problem. It normalizes the output of a network into a probability distribution of each of possible output.

Table 4.2: Parameter and values setting for Model 2 (CNN+GRU)

| **Parameter** | **Experimental Range** | **Selected Value** |
|---|---|---|
| Headline word lenth | - | 20 |
| Article word lenth | - | 400 |
| GloVe Word Embedding dimension | - | 100 |
| Encoder CNN Filter size | 100,200 | 100 |
| Encoder CNN Kernel Size | 3,5 | 5 |
| Decoder CNN Filter size | 100,200 | 100 |
| Decoder CNN Kernel Size | 3,5 | 5 |
| Encoder Maxpool Size | 3,4 | 4 |
| Decoder Maxpool Size | 3,4 | 3 |
| Encoder GRU Unit | | 100 |
| Decoder GRU Unit | | 100 |
| Dense Layer Neurons | 50, 100 | 50 |
| Hidden layer Activation Functions | - | Relu |
| Output layer Activation Functions | - | Softmax |

### 4.1.3 Model 3 (CNN+BiLSTM)

The model make use of the CNN to retrieve local features and the BiLSTM to learn long-term dependencies of both headline and news article of the dataset, which of them carries the sequential information. Figure 4.4 illustrates the combined CNN+ BiLSTM models for fake news stance detection. The network has separate encoder and decoder network. Both encoder and decoder consists for the hybrid CNN+ BiLSTM model, which includes 1D convolutional layers followed by pooling layer, BiLSTM layer.

The preprocessed news article is sent to the GloVe embedding layer for word to vector representation on the encoder side. The embedding layer creates an embedding matrix by giving each token in the headlines and news body a 100-dimensional vector representation. After that, the vectorized data is sent to the convolutional layer. The convolutional layer is used for local feature extraction that is activated by the relu activation function. The max pooling layer is used for dimensionality reduction of the given input text. The function map is transferred to the BiLSTM layer to extract sequential information. Similarly, in decoder side, the preprocessed headline fed with the embedding layer, Convolutional layer, max pool layer and BiLSTM layer. The decoder BiLSTM layer takes the encoder context vector and headlines processed vector.

The output vector from the decoder connected to the fully connected layer of 50 nodes to predict whether they belong under any of the given categories. In this experiment, FNC-1 dataset has

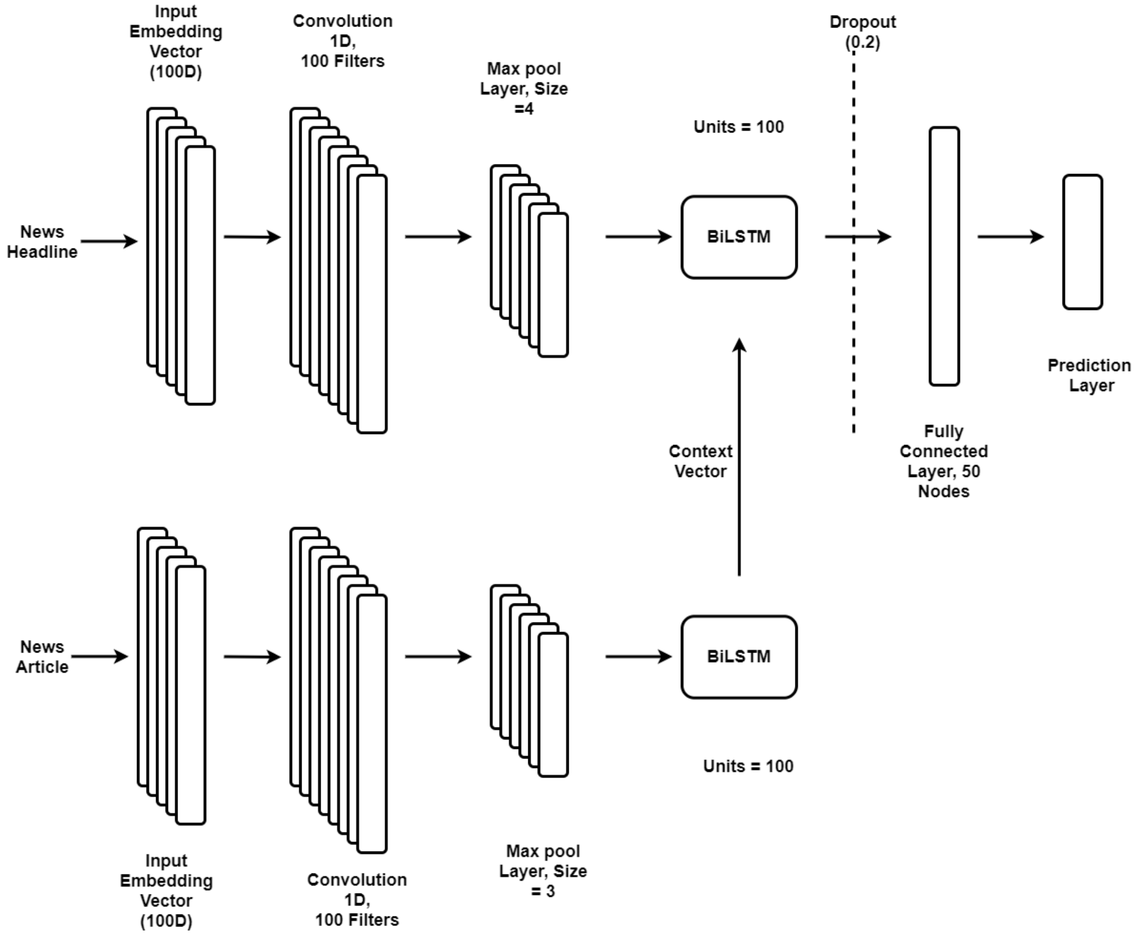



Figure 4.4: Model 3 (CNN+BiLSTM) for fake news stance detection

four output labels and ISOT has two classes. So, in the prediction layer of FNC-1 and ISOT dataset has four and two number of nodes respectively.

To train Model 3(CNN+BiLSTM), we used a learning rate of 0.001 which is the default learning rate value for rms prop optimizer in Keras. A dropout value of 0.2 was applied in the after the decoder BiLSTM layer so that the model does not over fit the input data. The batch size, the number of training samples used by the optimizer before updating the model parameters, of 64 datasets was used. Softmax is a activation function use to classify multiclass problem. It normalizes the output of a network into a probability distribution of each of possible output. It is also called a vector of probability distribution. It produces values in the range of 0-1 therefore will be use as the final layer in the classification models. A stance detection of headlines and article body for fake news detection system have been successfully completed on Model 3 on both dataset.

Table 4.3: Parameter and values setting for Model 3 (CNN+BiLSTM)

| **Parameter** | **Experimental Range** | **Selected Value** |
|---|---|---|
| Headline word lenth | - | 20 |
| Article word lenth | - | 400 |
| GloVe Word Embedding dimension | - | 100 |
| Encoder CNN Filter size | 100,200 | 100 |
| Encoder CNN Kernel Size | 3,5 | 5 |
| Decoder CNN Filter size | 100,200 | 100 |
| Decoder CNN Kernel Size | 3,5 | 5 |
| Encoder Maxpool Size | 3,4 | 4 |
| Decoder Maxpool Size | 3,4 | 3 |
| Encoder GRU Unit | | 100 |
| Decoder GRU Unit | | 100 |
| Dense Layer Neurons | 50, 100 | 50 |
| Hidden layer Activation Functions | - | Relu |
| Output layer Activation Functions | - | Softmax |

# CHAPTER 5

# RESULTS AND DISCUSSION

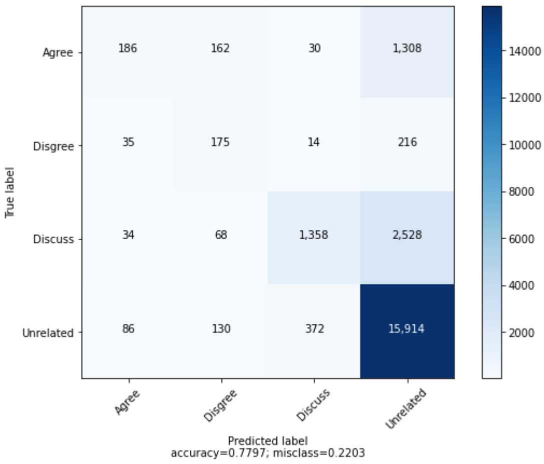
For the model building and evaluation, we use two different datasets and three different neural network models. FNC-1 dataset is the four label dataset and ISOT is two label (binary classification) dataset. A stance detection of headlines and article body for fake news detection system have been successfully completed on three hybrid model on both dataset. The accuracy, precision, recall, and F1 score were used to assess the performance.

Table 5.1: Model building parameters on both dataset

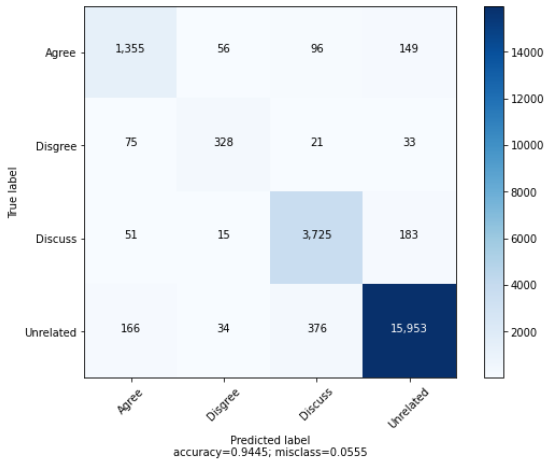
| **Parameter** | **FNC-1** | **ISOT** |
|---|---|---|
| No. of headline and article pair for training dataset | 42215 | 25143 |
| No. of headline and article pair for validation dataset | 10554 | 6286 |
| No. of headline and article pair for testing dataset | 22616 | 13469 |
| Optimizer | Rms prop | Rms prop |
| Loss Function | Categorical cross entropy | Binary cross entropy |
| Batch size | 64 | 64 |
| Epoch | 15 | 10 |

## 5.1 FNC-1 Dataset

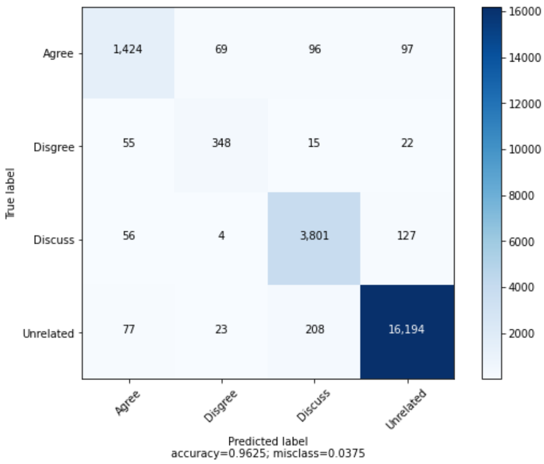
### 5.1.1 Confusion Matrix




(a) Model 1 (CNN+ Simple RNN)




(b) Model 2 (CNN+ GRU)




(c) Model 3(CNN+ BiLSTM)

Figure 5.1: Confusion matrix of three models on FNC-1 testing dataset (a) Model 1 (b) Model 2 (c) Model 3

The confusion matrix for our three models on the testing dataset of the FNC-1 dataset is shown in Figure 5.1. The confusion matrix represents the number of TP, TN, FP and FN on individual classes. It shows the accuracy on FNC-1 testing dataset 77.97%, 94.45% and 96.25% using CNN+Simple RNN, CNN+GRU and CNN+BiLSTM hybrid models on our encoder decoder architecture respectively. The FNC-1 dataset is unbalanced dataset, so we also have to evaluate and analyse other evaluation metrics precision, recall and F1 score. The detail of classwise different evaluation metrics are tabulated in next section Classification Evaluation Metrics.

### 5.1.2 Classification Evaluation Metrics

The FNC-1 dataset is unbalanced dataset, so we also evaluate and analyse evaluation metrics accuracy, precision, recall and F1 score.

**Model 1 (CNN+ Simple RNN)**

The model 1 is built using the combination of CNN and simple RNN layers in encoder decoder architecture. Table 5.2 presents the classwise evaluation metrics values achieved by feding the testing set of FNC-1 dataset in the trained model using model 1. The evaluation metrics are accuracy, precision, recall and F1- score. Based on TP, TN, FP and FN values of the individual classes, the evaluation metrics are calculated. These TP, TN, FP and FN values can be found from the confusion matrix of model 1.

Table 5.2: Classwise Evaluation Metrics of Model 1 (CNN+ Simple RNN) on FNC-1 testing dataset

| **Stance** | **Accuracy(%)** | **Precision(%)** | **Recall(%)** | **F1-Score(%)** |
|---|---|---|---|---|
| Agree | 92.68 | 54.55 | 11.03 | 18.35 |
| Disagree | 97.24 | 32.71 | 39.77 | 35.9 |
| Discuss | 86.53 | 76.55 | 34.05 | 47.14 |
| Unrelated | 79.48 | 79.71 | 96.44 | 87.28 |
| **Overall** | **77.97** | **60.88** | **45.32** | **51.96** |

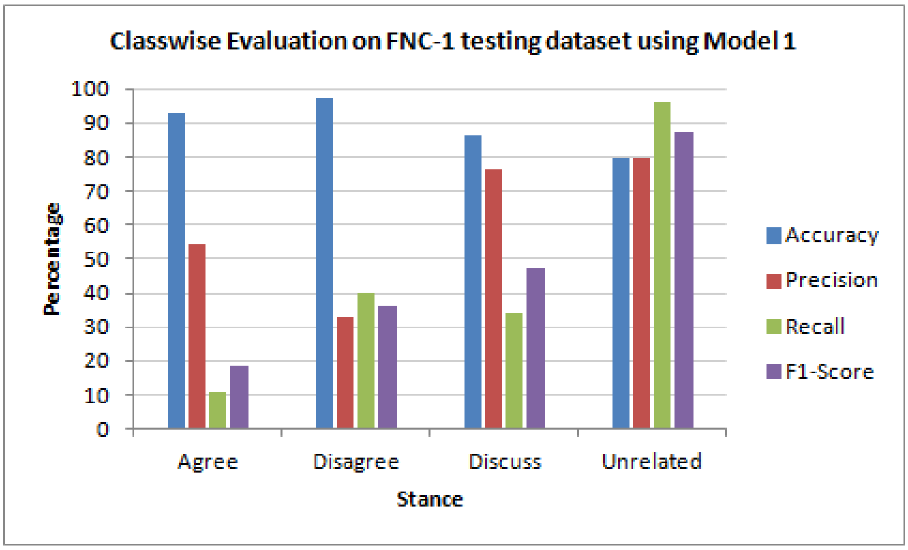



Figure 5.2: Bar Diagram for Performance Evaluation Metrics of Model 1 (CNN+ Simple RNN) on FNC -1 testing dataset

From the calculation of overall evaluation metrics, it is found that the accuracy, precision, recall and F1- score are 77.97, 60.88, 45.32 and 51.96 respectively.

### Model 2 (CNN+ GRU)

The model 2 is built using the combination of CNN and GRU layers in encoder decoder architecture. Table 5.3 presents the classwise evaluation metrics values achieved by feding the testing set of FNC-1 dataset in the trained model using model 2. The evaluation metrics are accuracy, precision, recall and F1- score. Based on TP, TN, FP and FN values of the individual classes, the evaluation metrics are calculated. These TP, TN, FP and FN values can be found from the confusion matrix of model 2.

Table 5.3: Classwise Evaluation Metrics of Model 2 (CNN+ GRU) on FNC-1 testing dataset

| Stance | Accuracy(%) | Precision(%) | Recall(%) | F1-Score(%) |
|---|---|---|---|---|
| Agree | 97.38 | 82.27 | 81.82 | 82.05 |
| Disagree | 98.97 | 75.75 | 71.77 | 73.71 |
| Discuss | 96.72 | 88.31 | 93.73 | 90.94 |
| Unrelated | 95.84 | 97.76 | 96.52 | 97.14 |
| **Overall** | **94.45** | **86.02** | **85.96** | **85.99** |

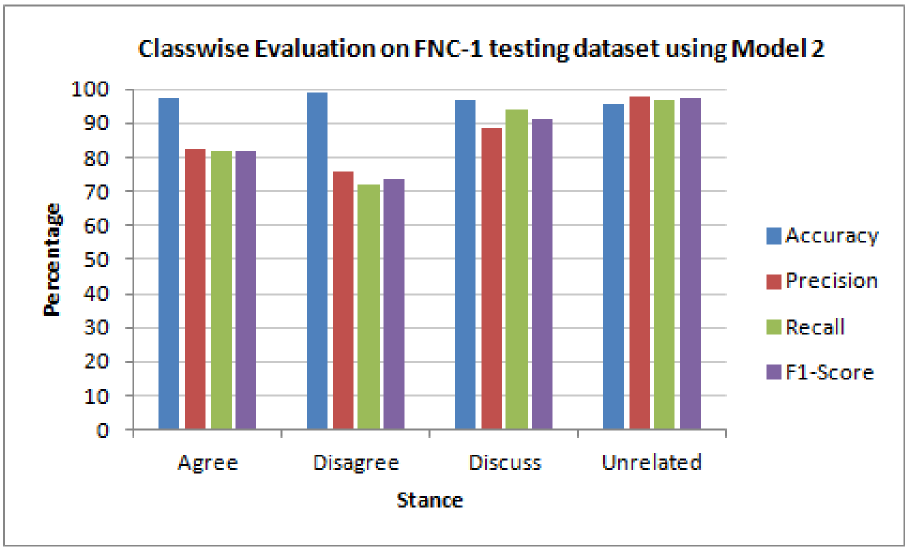



Figure 5.3: Bar Diagram for Performance Evaluation Metrics of Model 2 (CNN+ GRU) on FNC -1 testing dataset

From the calculation of overall evaluation metrics, it is found that the accuracy, precision, recall and F1- score are 94.45, 86.02, 85.96 and 85.99 respectively.

### Model 3 (CNN+ BiLSTM)

The model 3 is built using the combination of CNN and bidirectional LSTM layers in encoder decoder architecture. Table 5.4 presents the classwise evaluation metrics values achieved by feding the testing set of FNC-1 dataset in the trained model using model 3. The evaluation metrics are accuracy, precision, recall and F1- score. Based on TP, TN, FP and FN values of the individual classes, the evaluation metrics are calculated. These TP, TN, FP and FN values can be found from the confusion matrix of model 3.

Table 5.4: Classwise Evaluation Metrics of Model 3 (CNN+ BiLSTM) on FNC-1 testing dataset

| **Stance** | **Accuracy(%)** | **Precision(%)** | **Recall(%)** | **F1-Score(%)** |
|---|---|---|---|---|
| Agree | 98.01 | 88.34 | 84.46 | 86.36 |
| Disagree | 99.17 | 78.38 | 79.09 | 78.73 |
| Discuss | 97.76 | 92.26 | 95.31 | 93.76 |
| Unrelated | 97.55 | 98.5 | 98.13 | 98.32 |
| **Overall** | **96.25** | **89.37** | **89.25** | **89.31** |

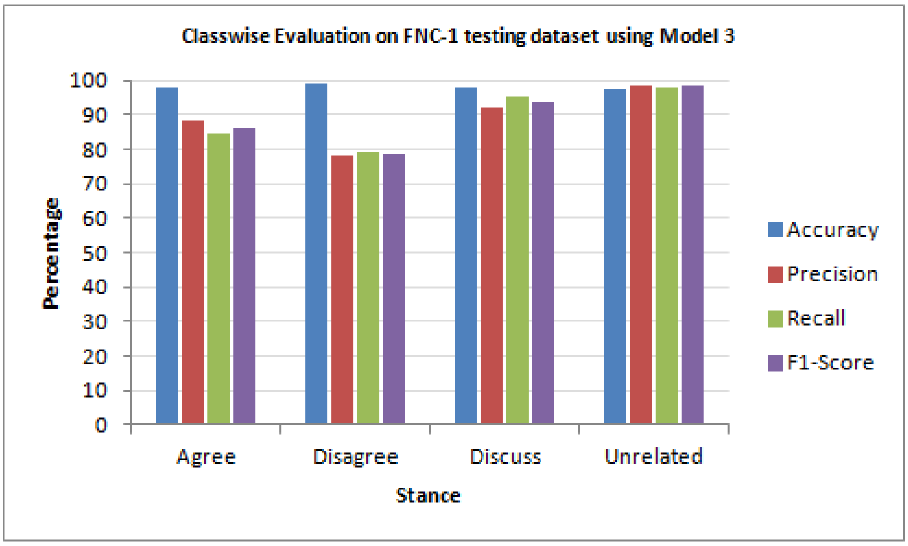



Figure 5.4: Bar Diagram for Performance Evaluation Metrics of Model 3 (CNN+BiLSTM) on FNC -1 testing dataset

From the calculation of overall evaluation metrics, it is found that the accuracy, precision, recall and F1- score are 96.25, 89.37, 89.25 and 89.31 respectively.

### Comparison of Three Models

Table 5.5: Evaluation Metrics Comparison of Three models on FNC-1 Dataset

| **Model** | **Accuracy(%)** | **Precision(%)** | **Recall(%)** | **F1-Score(%)** |
|---|---|---|---|---|
| Model 1 | 77.97 | 60.88 | 45.32 | 51.96 |
| Model 2 | 94.45 | 86.02 | 85.96 | 85.99 |
| Model 3 | **96.25** | **89.37** | **89.25** | **89.31** |

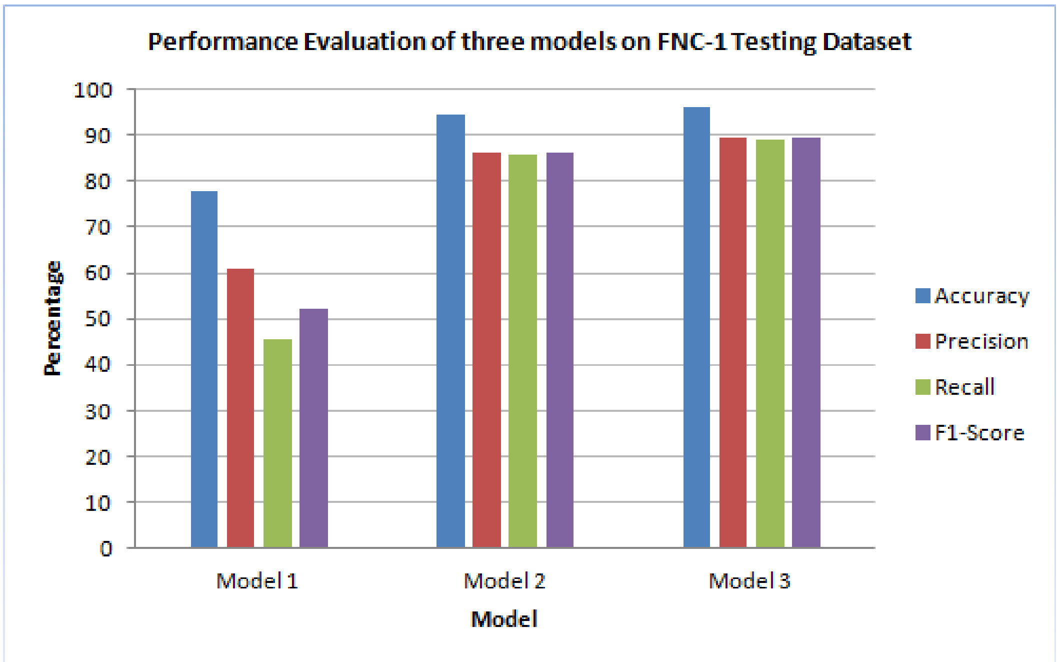


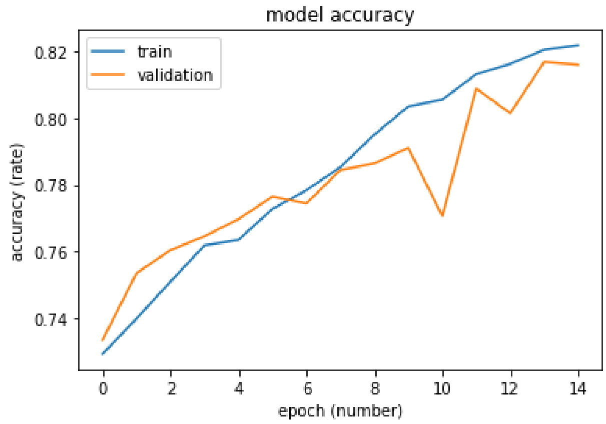

Figure 5.5: Bar Diagram for Performance Evaluation Metrics Comparison of Three Models on FNC -1 testing dataset

Finally, from the comparison of three models in Table 5.5, we found that the model 3 (CNN+BiLSTM) has the superior performance in accuracy, precision, recall and F1 score.

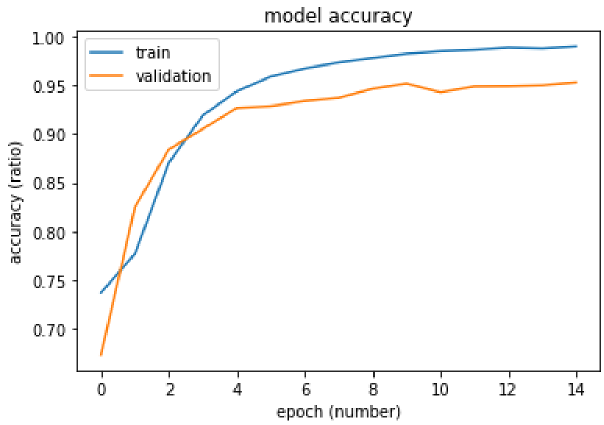
### 5.1.3 Learning Curves

Learning curves are commonly used in deep learning neural networks. Deep neural networks learn the hidden patterns from dataset progressively over time through optimization of their internal parameters. When one parameter on the horizontal axis varies, such as training set size or iteration/time, the learning curve displays progress in performance on the vertical axis. Training learning curve is a learning curve extracted from the training dataset that shows how much the model is learning and validation learning curve is the learning curve derived from a validation dataset in model building process. Both accuracy and loss curves are plotted for the training and validation. In machine learning, learning curves are a common diagnostic tool for algorithms that learn progressively from a training dataset. After each update during training, the model may be assessed on the training dataset and a hold out validation dataset, and graphs of the measured performance can be generated to demonstrate learning curves.
The training accuracy curve for both training and validation dataset during the model training process are shown in Figure 5.6. Similarly, the training loss curve for both training and validation dataset during the model training process are shown in Figure 5.7. The increase in accuracy and decrease in loss value at initial epochs are high, which shows the model is in learning process.

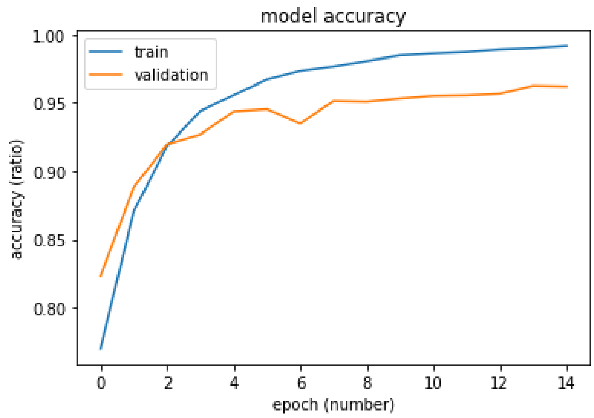
After few epochs, the model goes to saturation level with the high accuracy and low loss value.




(a) Model 1



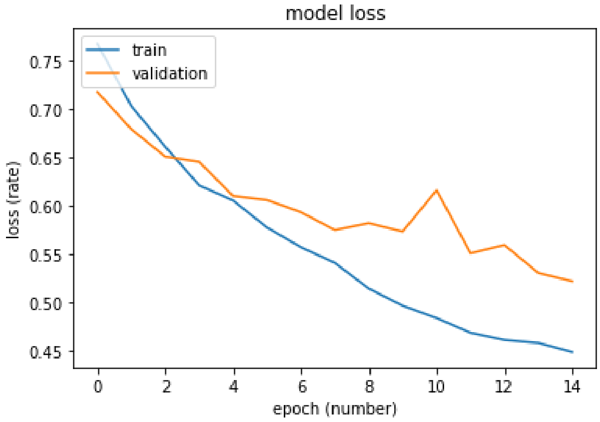

(b) Model 2



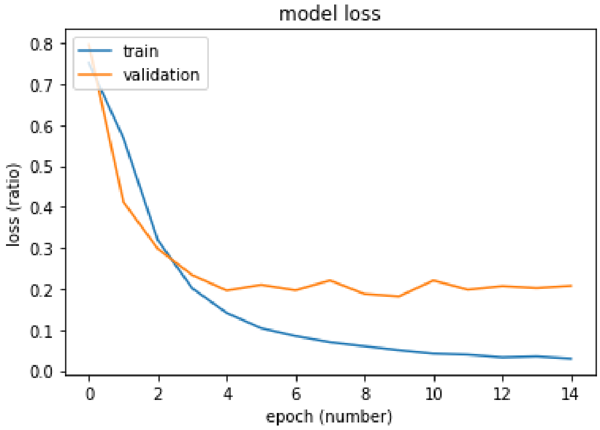

(c) Model 3

Figure 5.6: Training Vs Validation Accuracy Curves for three models on FNC-1 dataset: (a) Model 1 (b) Model 2 (c) Model 3

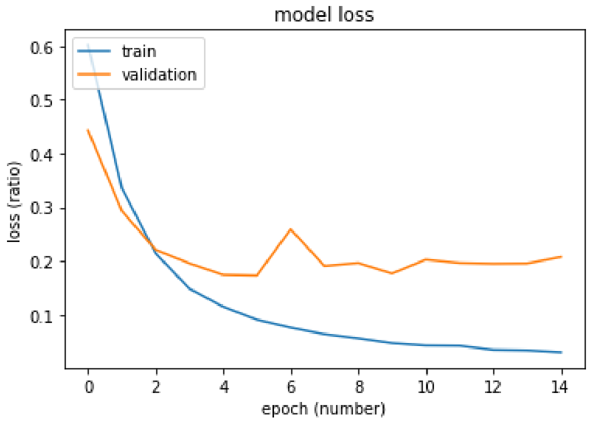
The plot of accuracy vs epoch and loss vs epoch curve for the three models are shown in Figure 5.6 and Figure 5.7. The model accuracy was increasing and loss was decreasing over time, which means the model was learning. With the increase in training accuracy value, the validation accuracy was also increased. The gap between the training accuracy and validation accuracy are small which represents the good fit of the model. On Model 2 and Model 3, both training/ validation accuracy and loss are at their best after 7 epochs.




(a) Model 1



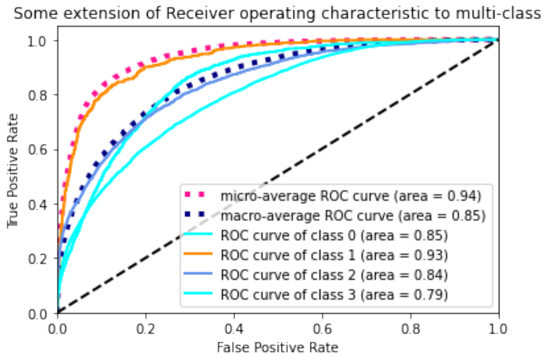

(b) Model 2



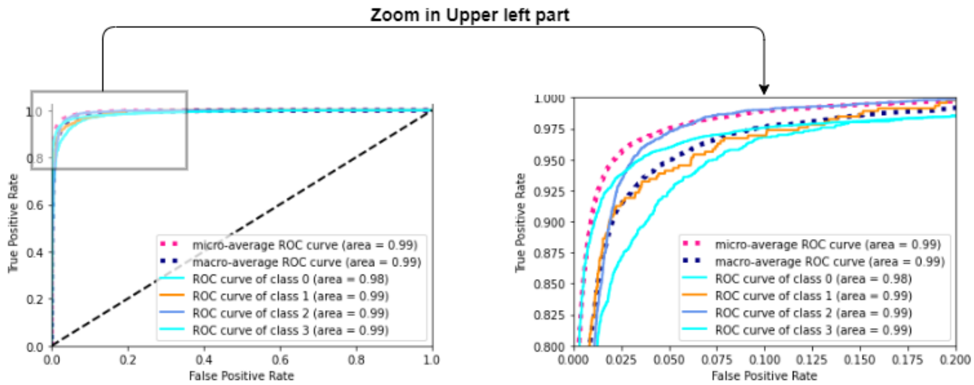

(c) Model 3

Figure 5.7: Training Vs Validation Loss Curves for three models on FNC-1 Dataset: (a) Model 1 (b) Model 2 (c) Model 3

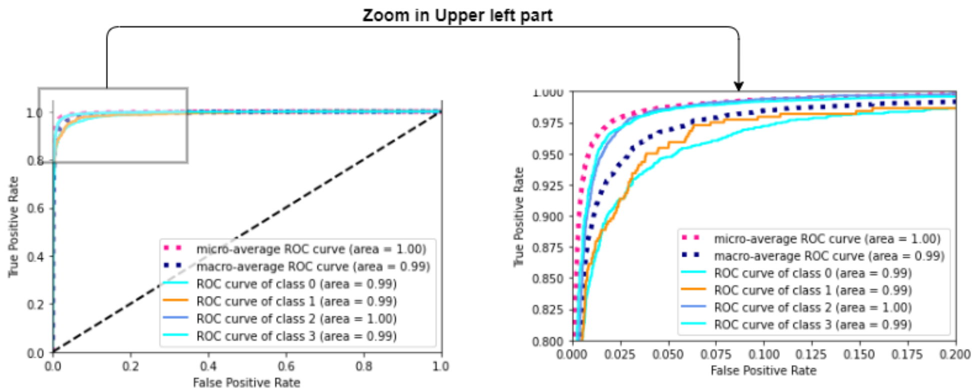
### 5.1.4 ROC AUC Curve




(a) Model 1 (CNN+ Simple RNN)




(b) Model 2 (CNN+ GRU)



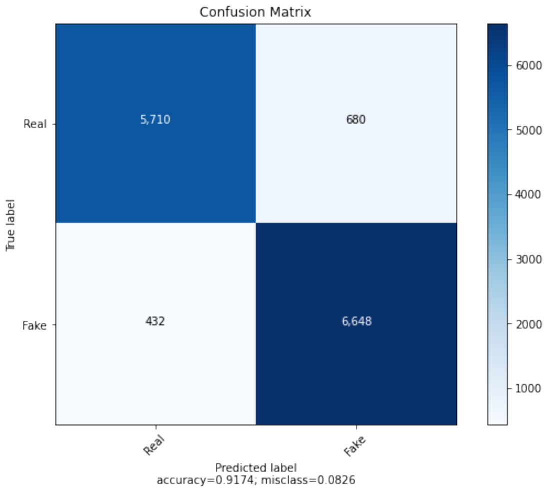

(c) Model 3 (CNN+ BiLSTM)

Figure 5.8: ROC AUC Curves for three models on FNC-1 dataset: (a) Model 1 (b) Model 2 (c) Model 3

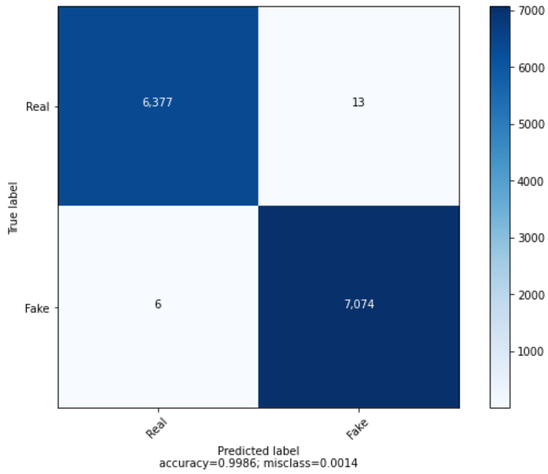
Finally, as shown in Figure 5.8, the ROC AUC curves for four classes of the FNC-1 dataset were displayed, with the X-axis indicating the false positive rate and the Y-axis indicating the true positive rate. The areas under the curve for the given three models were plotted as shown in above figures . These ROC AUC values represent the probability that the model will be able to differentiate between the classes.

## 5.2 ISOT Dataset

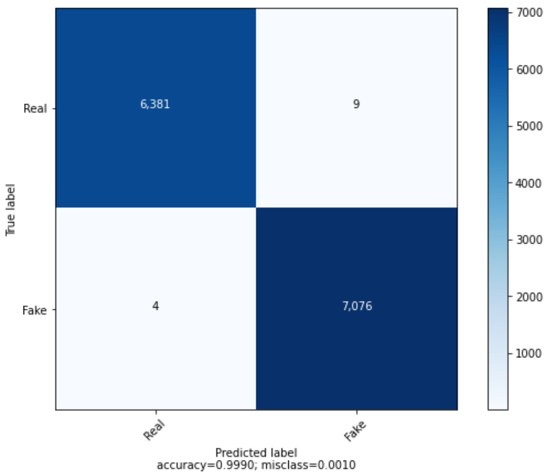
### 5.2.1 Confusion Matrix




(a) Model 1 (CNN+ Simple RNN)




(b) Model 2 (CNN+ GRU)




(c) Model 3 (CNN+ BILSTM)

Figure 5.9: Confusion matrix of three models on ISOT testing dataset: (a) Model 1 (b) Model 2 (c) Model 3

Figure 5.9 shows the confusion matrix our three models on testing dataset of ISOT dataset. The confusion matrix represents the number of true positive, true negative, false positive and false negative. It shows the accuracy on ISOT testing dataset 91.74%, 99.86% and 99.9% using CNN+Simple RNN, CNN+GRU and CNN+BiLSTM hybrid models on our encoder decoder architecture respectively. The detail of classwise calculations of evaluation metrics are tabulated in next section Classification Evaluation Metrics.

### 5.2.2 Classification Evaluation Metrics

#### Model 1 (CNN+ Simple RNN)

The model 1 is built using the combination of CNN and simple RNN layers in encoder decoder architecture. Table 5.6 presents the classwise evaluation metrics values achieved by feding the testing set of ISOT dataset in the trained model using model 1. The evaluation metrics are accuracy, precision, recall and F1- score.

Table 5.6: Classwise Evaluation Metrics of Model 1 on ISOT testing dataset

| **News Type** | **Accuracy(%)** | **Precision(%)** | **Recall(%)** | **F1-Score(%)** |
|---|---|---|---|---|
| Real | 91.74 | 92.97 | 89.36 | 91.13 |
| Fake | 91.74 | 90.72 | 93.9 | 92.28 |
| **Overall** | **91.74** | **91.62** | **91.63** | **91.62** |

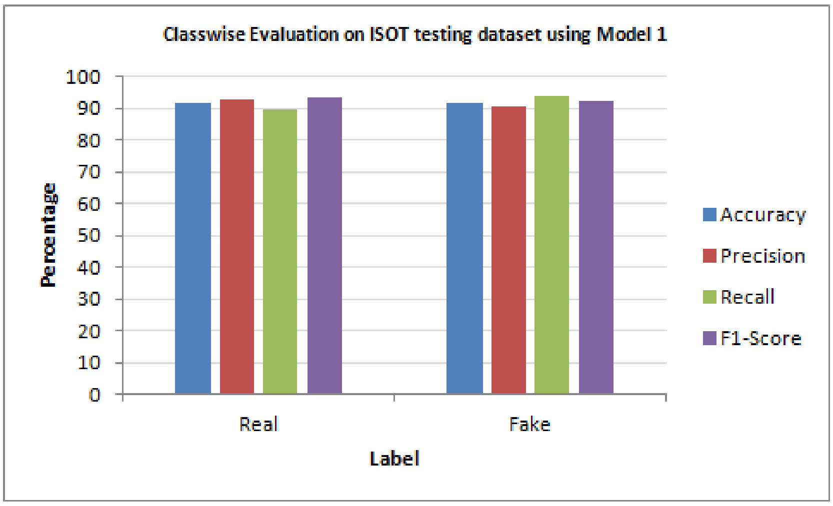



Figure 5.10: Bar Diagram for Performance Evaluation Metrics of Model 1 on ISOT testing dataset

Finally, in calculation of overall evaluation metrics, it is found that the accuracy, precision, recall and F1- score are 91.74, 91.62, 91.63 and 91.62 respectively.

#### Model 2 (CNN+ GRU)

The model 2 is built using the combination of CNN and GRU layers in encoder decoder architecture. Table 5.7 presents the classwise evaluation metrics values achieved by feding the testing set of ISOT dataset in the trained model using model 2. The evaluation metrics are accuracy, precision, recall and F1- score.

Table 5.7: Classwise Evaluation Metrics of Model 2 on ISOT testing dataset

| News Type | Accuracy(%) | Precision(%) | Recall(%) | F1-Score(%) |
|---|---|---|---|---|
| Real | 99.86 | 99.9 | 99.8 | 99.85 |
| Fake | 99.86 | 99.82 | 99.92 | 99.87 |
| **Overall** | **99.86** | **99.86** | **99.86** | **99.86** |

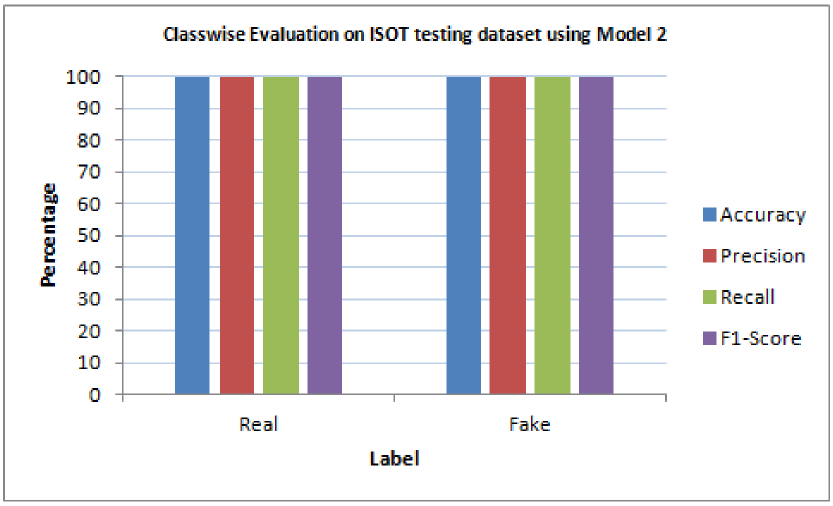



Figure 5.11: Bar Diagram for Performance Evaluation Metrics of Model 2 on ISOT testing dataset

Finally, in calculation of overall evaluation metrics, it is found that the accuracy, precision, recall and F1- score are 99.86, 99.86, 99.86 and 99.86 respectively.

**Model 3 (CNN+ BiLSTM)**

The model 3 is built using the combination of CNN and GRU layers in encoder decoder architecture. Table 5.8 presents the classwise evaluation metrics values achieved by feding the testing set of ISOT dataset in the trained model using model 3. The evaluation metrics are accuracy, precision, recall and F1- score.

Table 5.8: Classwise Evaluation Metrics of Model 3 on ISOT testing dataset

| News Type | Accuracy(%) | Precision(%) | Recall(%) | F1-Score(%) |
|---|---|---|---|---|
| Real | 99.9 | 99.94 | 99.85 | 99.9 |
| Fake | 99.9 | 99.87 | 99.94 | 99.9 |
| **Overall** | **99.9** | **99.9** | **99.9** | **99.9** |

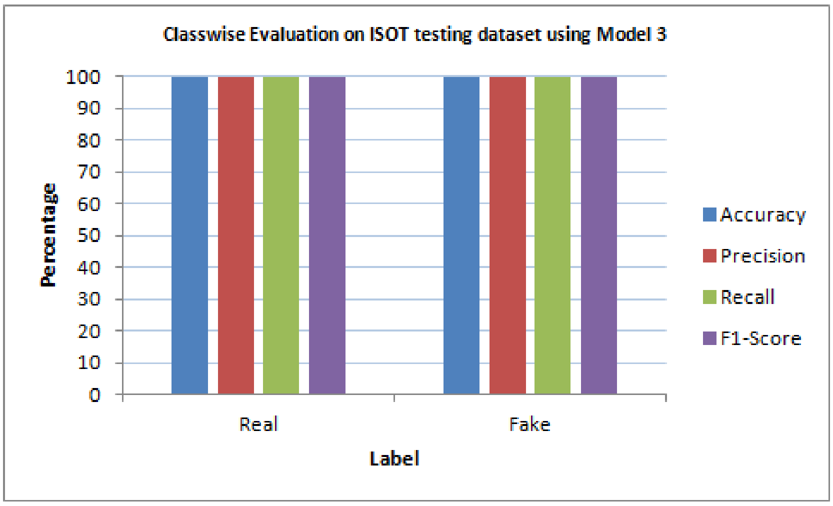



Figure 5.12: Bar Diagram for Performance Evaluation Metrics of Model 3 on ISOT testing dataset

Finally, in calculation of overall evaluation metrics, it is found that the accuracy, precision, recall and F1- score are 99.9, 99.9, 99.9 and 99.9 respectively.

### Comparison of three models

Table 5.9: Evaluation Metrics Comparison of Three models on ISOT Dataset

| **Model** | **Accuracy(%)** | **Precision(%)** | **Recall(%)** | **F1-Score(%)** |
|---|---|---|---|---|
| Model 1 (CNN+ Simple RNN) | 91.74 | 91.62 | 91.63 | 91.62 |
| Model 2 (CNN+ GRU) | 99.86 | 99.86 | 99.86 | 99.86 |
| Model 3 (CNN+ BiLSTM) | **99.9** | **99.9** | **99.9** | **99.9** |

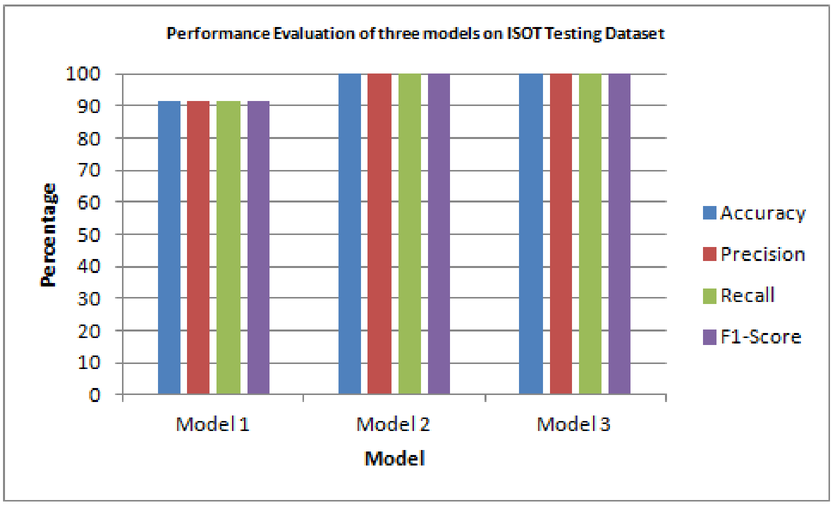
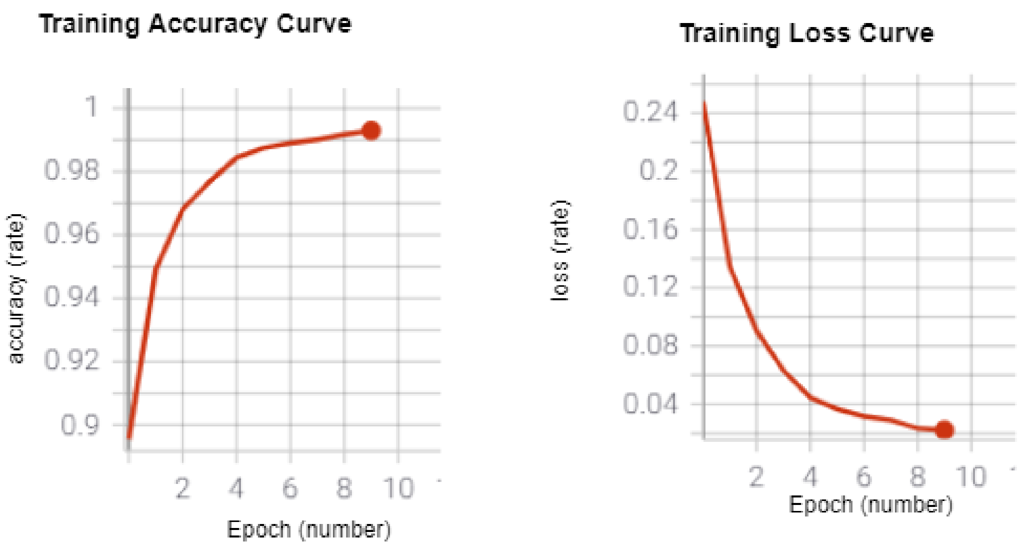



Figure 5.13: Bar Diagram for Performance Evaluation Metrics comparison of Three models on ISOT testing dataset

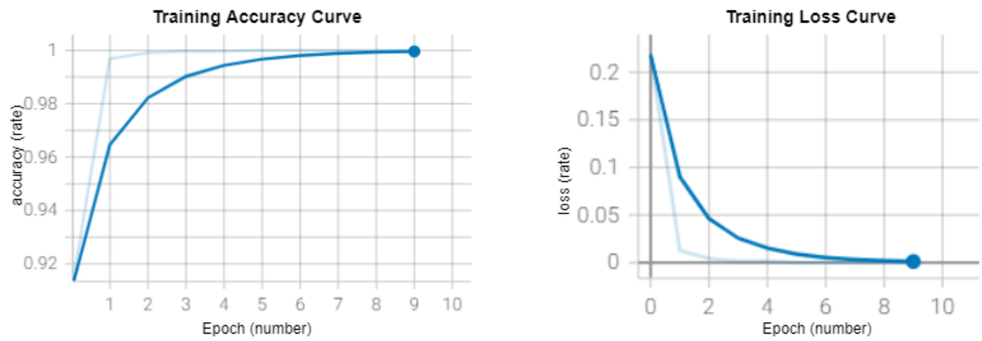
Finally, from the comparison of three models in Table 5.9, we found that the model 3 (CNN+BiLSTM) has the superior performance in accuracy, precision, recall and F1 score.

### 5.2.3 Learning Curves

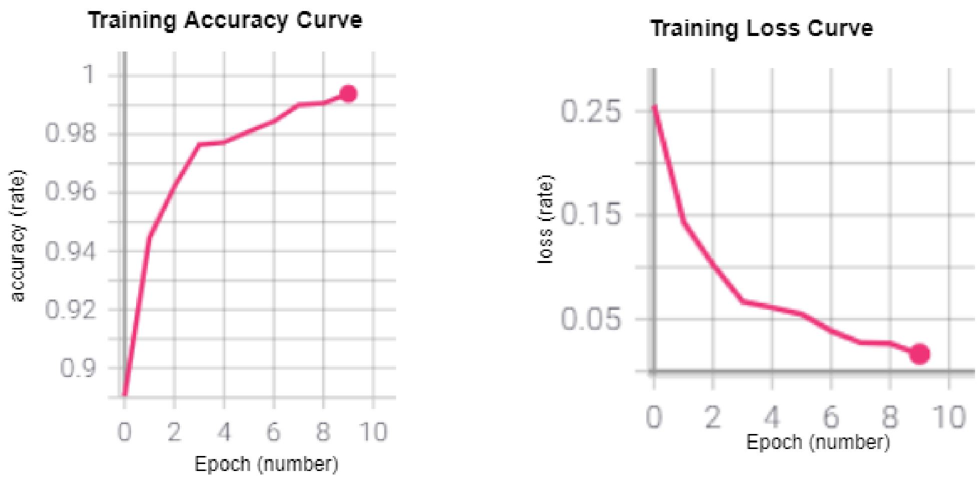
Deep neural networks learn the hidden patterns from dataset progressively over time through optimization of their internal parameters. When one parameter on the horizontal axis varies, such as training set size or iteration/time, the learning curve displays progress in performance on the vertical axis. Training learning curve is a learning curve extracted from the training dataset that shows how much the model is learning and validation learning curve is the learning curve derived from a validation dataset in model building process. Both accuracy and loss curves are plotted for the training process. In machine learning, learning curves are a common diagnostic tool for algorithms that learn progressively from a training dataset. After each update during training, the model may be assessed on the training dataset and a hold out validation dataset, and graphs of the measured performance can be generated to demonstrate learning curves.




(a) Model 1(CNN+ Simple RNN)




(b) Model 2(CNN+ GRU)

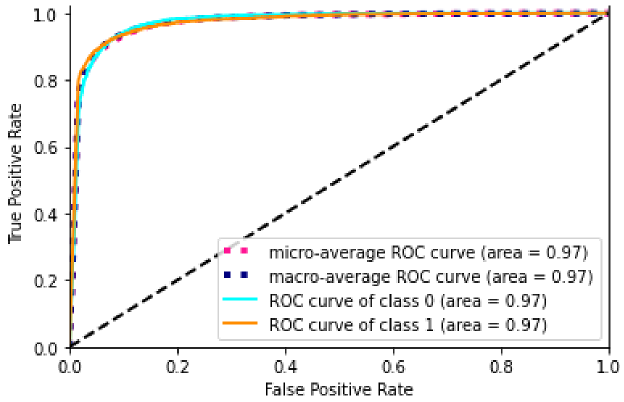



(c) Model 3(CNN+ BiLSTM)

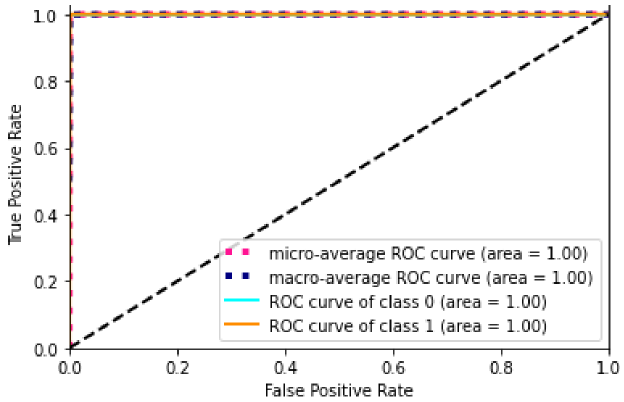
Figure 5.13: Training Accuracy and Loss Curves for three models on ISOT dataset (a) Model 1 (b) Model 2 (c) Model 3

Figure 5.13 shows a plot of accuracy vs epochs and loss vs epochs curve for the respective models. The model accuracy was increasing and loss was decreasing over time, which means the model was learning.

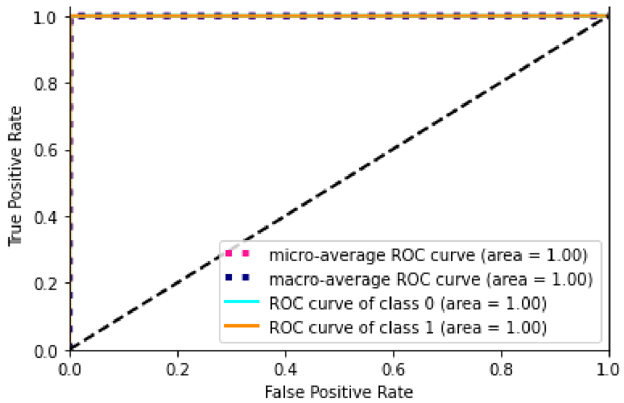
### 5.2.4 ROC AUC Curve




(a) Model 1 (CNN+ Simple RNN)




(b) Model 2 (CNN+ GRU)




(c) Model 3 (CNN+ BiLSTM)

Figure 5.14: ROC AUC Curves for three models on ISOT dataset (a) Model 1 (b) Model 2 (c) Model 3

Finally, as shown in Figure 5.14 , the ROC AUC curves for two classes of ISOT dataset were displayed, with the X-axis indicating the false positive rate and the Y-axis indicating the true positive rate. The areas under the curve for the given three models were plotted as shown in above figures . These ROC AUC values represent the probability that the model will be able to differentiate between the classes.

## 5.3 Evaluation on Own Collected Dataset

From the evaluation comparison of three models, Model 3 (CNN +BiLSTM) performed well on both FNC-1 and ISOT dataset. We also evaluate the best performing model (Model 3) on generalization test of the model. Generalization is a model evaluation that determines how effectively a trained model can categorize an unseen data. The hybrid CNN-BiLSTM model was trained on the ISOT dataset and evaluated on the own collected dataset to enhance work in model generalization test. The dataset is made up of CSV files. Each article contains article title, text and type of news. True news is taken from reputable news organizations such as the BBC, New York Times and CNN (Cable News Network). Fake news is taken from a fact-checking website politifact.com. There are 153 articles in the dataset, 81 of which are true and 72 are fake.

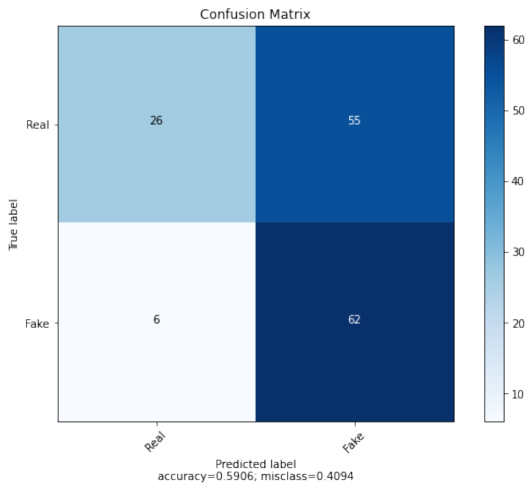



Figure 5.15: Confusion matrix on own collected dataset

Table 5.10: Performance Evaluation Report on own collected dataset

| | Labels | True Positives | True Negatives | False Positives | False Negatives | Accuracy | Precision | Recall | Specificity | F1 Score |
|---|---|---|---|---|---|---|---|---|---|---|
| 0 | Real | 26 | 62 | 6 | 55 | 0.590604 | 0.812500 | 0.320988 | 0.911765 | 0.460177 |
| 1 | Fake | 62 | 26 | 55 | 6 | 0.590604 | 0.529915 | 0.911765 | 0.320988 | 0.670270 |

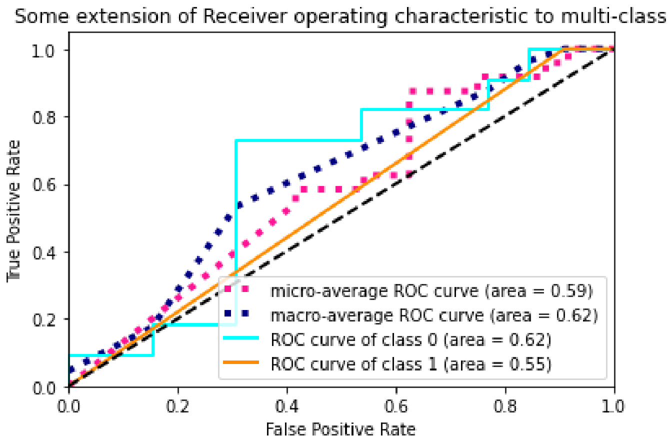



Figure 5.16: ROC AUC curve on own collected dataset

# CHAPTER 6

# CONCLUSION

Fake news is usually created to confuse and attract audiences for commercial and political benefit. The way to observe the fake news is using stance detection technique, is the focus of this research. Given a set of news body and headline pair, stance detection is the task of automatic detection of textual relationship among texts. In this thesis work, we analyze three hybrid models, CNN+simple RNN, CNN+GRU and CNN+BiLSTM in encoder decoder architecture to predict the stance between headline and article of the news. The models were successfully trained and tested on both binary (ISOT) and multiclass (FNC-1) fake news datasets. The accuracy on FNC-1 testing dataset using three models CNN+Simple RNN, CNN+ GRU and CNN+ BiLSTM are 77.97%, 94.45% and 96.25% respectively. Also on comparing other evaluation metrics precision, recall and F1-score, the CNN+ BiLSTM performed well on multiclass dataset FNC-1. Similarly, the accuracy on ISOT testing dataset using three models CNN+Simple RNN, CNN+ GRU and CNN+ BiLSTM are 91.74%, 99.86% and 99.9% respectively. Also on comparing other evaluation metrics precision, recall and F1-score, the CNN+ BiLSTM performed well on binary labelled dataset ISOT. Hence, it is concluded that the CNN+ BiLSTM model had better results than other two hybrid models in both binary and multiclass classification task for the fake news stance detection system.

## REFERENCES


[1] A. Thota, P. Tilak, S. Ahluwalia, and N. Lohia, "Fake news detection: A deep learning approach," *SMU Data Science Review*, vol. 1, no. 3, p. 10, 2018.

[2] A Lavoipierre, "Fake news' named 2016 word of the year by macquarie dictionary," *ABC News*, vol. 25, 2017.

[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, 2017.

[4] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 2014, pp. 1532–1543.

[5] K Kowsari, K Jafari Meimandi, M Heidarysafa, S Mendu, L Barnes, and D Brown, "Text classification algorithms: A survey. information 10 (4): 150," *arXiv preprint arXiv:1904.08067*, 2019.

[6] R. Collobert and J. Weston, "A unified architecture for natural language processing: Deep neural networks with multitask learning," in *Proceedings of the 25th international conference on Machine learning*, 2008, pp. 160–167.

[7] S. Mohammad, S. Kiritchenko, P. Sobhani, X. Zhu, and C. Cherry, "Semeval-2016 task 6: Detecting stance in tweets," in *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, 2016, pp. 31–41.

[8] M. Walker, P. Anand, R. Abbott, and R. Grant, "Stance classification using dialogic properties of persuasion," in *Proceedings of the 2012 conference of the North American chapter of the association for computational linguistics: Human language technologies*, 2012, pp. 592–596.

[9] D. Pomerleau and D. Rao, "The fake news challenge: Exploring how artificial intelligence technologies could be leveraged to combat fake news," *Fake News Challenge*, 2017.

[10] K. Popat, S. Mukherjee, J. Strötgen, and G. Weikum, "Where the truth lies: Explaining the credibility of emerging claims on the web and social media," in *Proceedings of the 26th International Conference on World Wide Web Companion*, 2017, pp. 1003–1012.

[11] B. MacCartney and C. D. Manning, *Natural language inference*. Citeseer, 2009.

[12] W. Ferreira and A. Vlachos, "Emergent: A novel data-set for stance classification," in *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: Human language technologies*, 2016, pp. 1163–1168.

[13] B. Sean, S. Doug, and P. Yuxi, "Talos targets disinformation with fake news challenge victory," *Accessed: Oct*, vol. 29, p. 2019, 2017.

[14] G. Bhatt, A. Sharma, S. Sharma, A. Nagpal, B. Raman, and A. Mittal, "Combining neural, statistical and external features for fake news stance identification," in *Companion Proceedings of the The Web Conference 2018*, 2018, pp. 1353–1357.

[15] K. Shu, A. Sliva, S. Wang, J. Tang, and H. Liu, "Fake news detection on social media: A data mining perspective," *ACM SIGKDD explorations newsletter*, vol. 19, no. 1, pp. 22–36, 2017.

[16] Y. Shukla, N. Yadav, and A. Hari, "A unique approach for detection of fake news using machine learning," *International Journal for Research in Applied Science & Engineering Technology (IJRASET), ISSN*, pp. 2321–9653,

[17] Y. Chen, N. J. Conroy, and V. L. Rubin, "Misleading online content: Recognizing clickbait as false news," in *Proceedings of the 2015 ACM on workshop on multimodal deception detection*, 2015, pp. 15–19.

[18] V. Singh, R. Dasgupta, D. Sonagra, K. Raman, and I. Ghosh, "Automated fake news detection using linguistic analysis and machine learning," in *International conference on social computing, behavioral-cultural modeling, & prediction and behavior representation in modeling and simulation (SBP-BRiMS)*, 2017, pp. 1–3.

[19] A. Meyer-Baese and V. J. Schmid, *Pattern recognition and signal analysis in medical imaging*. Elsevier, 2014.

[20] X. Zhang, J. Zhao, and Y. LeCun, "Character-level convolutional networks for text classification," *arXiv preprint arXiv:1509.01626*, 2015.

[21] K. Shuang, X. Ren, J. Chen, X. Shan, and P. Xu, "Combining word order and cnn-lstm for sentence sentiment classification," in *Proceedings of the 2017 International Conference on Software and e-Business*, 2017, pp. 17–21.

[22] 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.

[23] H. Ahmed, I. Traore, and S. Saad, "Detection of online fake news using n-gram analysis and machine learning techniques," in *International conference on intelligent, secure, and dependable systems in distributed and cloud environments*, Springer, 2017, pp. 127–138.

[24] M. H. Goldani, S. Momtazi, and R. Safabakhsh, "Detecting fake news with capsule neural networks," *Applied Soft Computing*, vol. 101, p. 106 991, 2021.

[25] H. Ahmed, I. Traore, and S. Saad, "Detecting opinion spams and fake news using text classification," *Security and Privacy*, vol. 1, no. 1, e9, 2018.

[26] J. Camacho-Collados and M. T. Pilehvar, "On the role of text preprocessing in neural network architectures: An evaluation study on text categorization and sentiment analysis," *arXiv preprint arXiv:1707.01780*, 2017.

[27] A. See, P. J. Liu, and C. D. Manning, "Get to the point: Summarization with pointer-generator networks," *arXiv preprint arXiv:1704.04368*, 2017.

[28] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *arXiv preprint arXiv:1409.0473*, 2014.

[29] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Advances in neural information processing systems*, 2014, pp. 3104–3112.

[30] D. Kollias and S. P. Zafeiriou, "Exploiting multi-cnn features in cnn-rnn based dimensional emotion recognition on the omg in-the-wild dataset," *IEEE Transactions on Affective Computing*, 2020.

[31] S. Masood, A. Srivastava, H. C. Thuwal, and M. Ahmad, "Real-time sign language gesture (word) recognition from video sequences using cnn and rnn," in *Intelligent Engineering Informatics*, Springer, 2018, pp. 623–632.

# APPENDIX A

# Fake News Detection using Natural Language Processing and Deep Neural Network

Niroj Ghimire[1], Surendra Shrestha[2], Dipak Kumar Nidhi[3], and Rupesh Shrestha[4]

[1-4]*Tribhuvan University, Institute of Engineering*

075msice014.niroj@pcampus.edu.np[1], surendra@ioe.edu.np[2], 075msice005.dipak@pcampus.edu.np[3], *075msice018.rupesh@pcampus.edu.np[4]*

### *Abstract*

*With the advancement of technology, fake news is more widely exposed to users globally. Fake news can be found through popular platforms like social media and the Internet. There have been multiple solutions and efforts in the detection of fake news where it even works with artificial intelligence tools. The way to observe the fake news is using stance detection technique is the focus of this paper. Given a set of news body and headline pair, Stance Detection is the task of automatic detection of relationship among pieces of text. The stances between them can be described as 'agree', 'disagree', 'discuss' or 'unrelated'. In this paper, the LSTM-based encoding decoding model using pre-trained GloVe word embeddings achieved 93.69% accuracy on FNC-1 dataset.*

***Keywords:*** *Fake news; Stance detection; NLP; LSTM*

## 1. Introduction

Fake News represents false news or propaganda comprising disinformation transmitted via classical media outlets like newspapers and TV in addition to modern media sources such social media [1]. Fake news is characterized by two points: credibility and intent. Credibility assumes that fake news contains false facts and can be verified and intent implies that the false data was written in order to confuse the reader. The word of the year was also dubbed "Fake news" by the Macquarie dictionary in 2016, taking into account the existence of this phenomenon [2].

To achieve the desired result, fake news is often generated and distributed via social media. Identifying the vocabulary that is used to mislead readers is the essential task of identifying fake news. A difficult challenge is the concept of classifying fake news through learning word-level context. Therefore, the way to observe the fake news is using stance detection technique will be the objective of our paper. Given a set of news body and headline pair, stance Detection is the task of automatic detection of relationship among pieces of text.


ISSN : 2635-4586