# Fake News Detection:
# Covid19 Tweet Data

by

Sanchana Mohankumar

A Project submitted to a faculty of

Northeastern University

August 2022

# Table of Contents:

# 1. Introduction

Technology has been dominating our lives for the past few decades. It has changed the way we communicate and share information. The sharing of in-formation is no longer constrained by physical boundaries. It is easy to share information across the globe in the form of text, audio, and video. An integral part of this capability is the social media platforms. These platforms help in sharing personal opinions and information with much a wider audience. They have taken over traditional media platforms because of speed and focussed content. However, it has become equivalently easy for nefarious people with malicious intent to spread fake news on social media platforms.

Fake news is defined as a verifiably false piece of information shared intentionally to mislead the readers. It has been used to create a political, social, and economic bias in the minds of people for personal gains. It aims at exploiting and influencing people by creating fake content that sounds legit. On the extreme end, fake news has even led to cases of mob lynching and riots. Thus, it is extremely important to stop the spread of fake content on internet platforms. It is especially desirable to control fake news during the ongoing Covid-19 crisis. The pandemic has made it easy to manipulate a mentally stranded population eagerly waiting for this phase to end. Some people have reportedly committed suicide after being diagnosed with covid due to the misrepresentation of covid in social and even mainstream media. The promotion of false practices will only aggravate the covid situation.

# 2. Problem Statement

In this project we are going to implement machine learning, neural network and transformer model to predict whether or not a tweet is real and also to find the better performing model comparing all 3 cases which is machine learning, neural network and transformer models. So, in order to do so we will first train our model using covid19 tweet training dataset. Once our model is capable of doing it, we will run it on test dataset and predict whether the posted tweet is real or fake. So in our case the dataset is of social media posts and articles on COVID-19 with real and fake labels.

# 3. Data

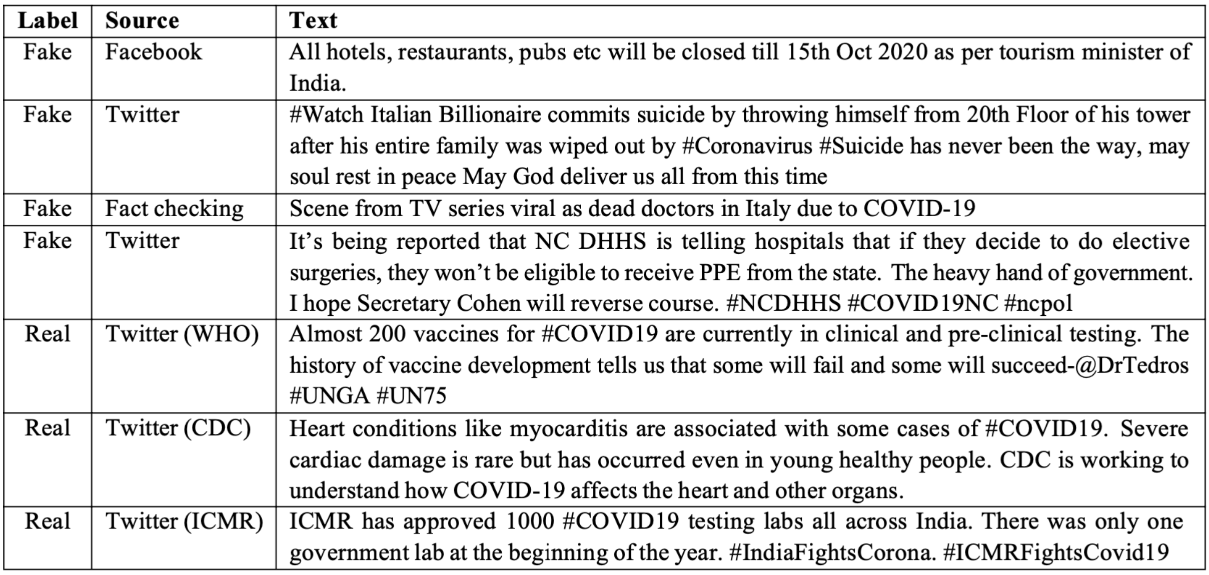
## 3.1 Data Description

In Table 1, the real tweets are from verified twitter accounts and Fake tweets are not verified twitter accounts. As we can see there are even tweets from sources such as Fact Checking where the source name is in itself deceiving to people. The real tweets are from official government websites which can be trusted as its owned by government not by an unknown user.

• **Real** - Tweets from verified sources and give useful information on COVID-19.

• **Fake** - Tweets, posts, articles which make claims and speculations about COVID-19 which are verified to be not true.

Classification: As we have categorical label with binary outcome our dataset is a classification task



| Label | Source | Text |
|---|---|---|
| Fake | Facebook | All hotels, restaurants, pubs etc will be closed till 15th Oct 2020 as per tourism minister of India. |
| Fake | Twitter | #Watch Italian Billionaire commits suicide by throwing himself from 20th Floor of his tower after his entire family was wiped out by #Coronavirus #Suicide has never been the way, may soul rest in peace May God deliver us all from this time |
| Fake | Fact checking | Scene from TV series viral as dead doctors in Italy due to COVID-19 |
| Fake | Twitter | It's being reported that NC DHHS is telling hospitals that if they decide to do elective surgeries, they won't be eligible to receive PPE from the state. The heavy hand of government. I hope Secretary Cohen will reverse course. #NCDHHS #COVID19NC #ncpol |
| Real | Twitter (WHO) | Almost 200 vaccines for #COVID19 are currently in clinical and pre-clinical testing. The history of vaccine development tells us that some will fail and some will succeed-@DrTedros #UNGA #UN75 |
| Real | Twitter (CDC) | Heart conditions like myocarditis are associated with some cases of #COVID19. Severe cardiac damage is rare but has occurred even in young healthy people. CDC is working to understand how COVID-19 affects the heart and other organs. |
| Real | Twitter (ICMR) | ICMR has approved 1000 #COVID19 testing labs all across India. There was only one government lab at the beginning of the year. #IndiaFightsCorona. #ICMRFightsCovid19 |

**Table 1:** Example of real and fake news from the dataset. Fake news is collected from various sources. The real news is collected from verified Twitter accounts

## 3.2 Data Source

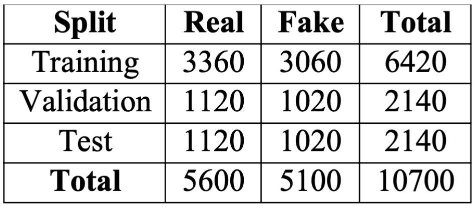
The project describes systems evaluated for Contraint@AAAI 2021 Covid-19 Fake news detection shared task. The task aims in improving the classification of the news based on Covid-19 as fake or real. The dataset shared is created by collecting data from various social media sources such as Instagram, Facebook, Twitter, etc.

The dataset is available in Kaggle. A link to the dataset is provided below.

https://www.kaggle.com/datasets/elvinagammed/covid19-fake-news-dataset-nlp

## 3.3 Data Statistics

The dataset is split into train (60%), validation (20%), and test (20%). Table 2 shows the class-wise distribution of all data splits. The dataset is class-wise balanced as 52.34% of the samples consist of real news and 47.66% of the data consists of fake news. Moreover, we maintain the class-wise distribution across train, validation, and test splits. The diagram below shows the split of Training, Validation, and Test data using BarCharts.



| Split | Real | Fake | Total |
|---|---|---|---|
| Training | 3360 | 3060 | 6420 |
| Validation | 1120 | 1020 | 2140 |
| Test | 1120 | 1020 | 2140 |
| **Total** | 5600 | 5100 | 10700 |

**Table 2:** Distribution of data

The figure below depicts the pictorial distribution of data between real and fake news using Bar Charts. In which figure 1 represents Training tweet data with  3360 real and 3060 fake tweet, figure 2 and 3 represents Validation and Test tweet data with  1120 real and 1020 fake tweet. As we know if data is  imbalanced the accuracy will be misleading so this helps understand distribution of data.

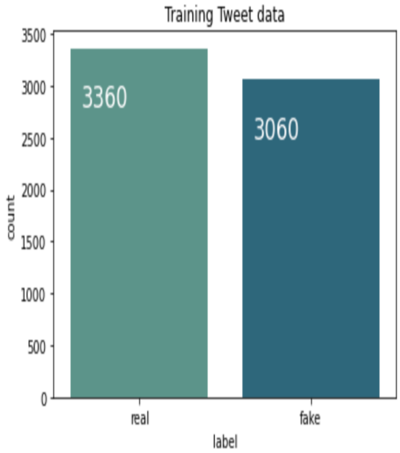


Fig 1:Pictorial distribution  of data

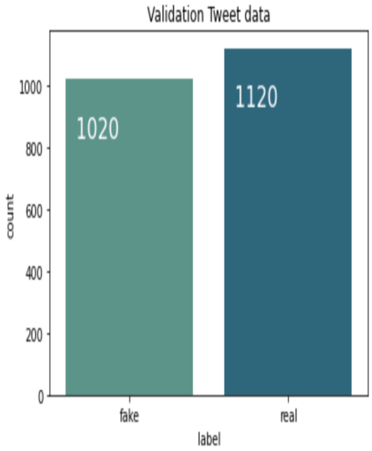


Fig 2:Pictorial distribution  of data

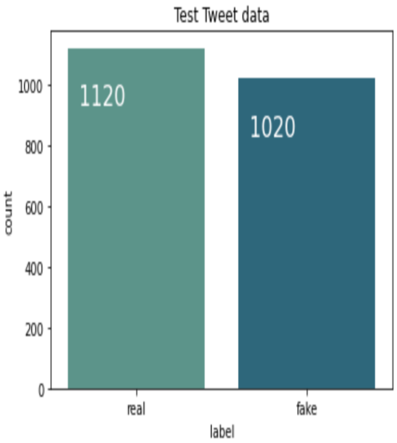


Fig 3:Pictorial distribution of data

## 3.4 Data Preprocessing

Data preprocessing, a component of data preparation, describes any type of processing performed on raw data to prepare it for another data processing procedure. It is considered the preliminary step in the data mining process. The steps performed are listed below

- **Removal of HTML tags:** Often in the process of gathering dataset, web or screen scraping leads to the inclusion of HTML tags in the text. These tags are often not paid heed to but it is necessary to get rid of them.
- **Removal of Special Characters:** Special characters are not readable because they are neither alphabets nor numbers. They include characters like ”*”, ”&”, ”$”, etc.
- **Noise Removal:** Noisy text includes unnecessary new lines, white spaces, etc. Filtering of such text is done in this process.
- **Normalization:** The entire text is converted into lowercase characters due to the case sensitive nature of NLP libraries.
- **Removal of stop-words:** English language stop words include words like ‘a’, ‘an’, ‘the’, ‘of’, ‘is’, etc which commonly occur in sentences and usually add less value to the overall meaning of the sentence. To ensure less processing time it is better to remove these stop words and let the model focus on the words that convey the main focus of the sentence.
- **Stemming**: This step reduces the word to its root word after removing the suffixes. But it does not ensure that the resulting word is meaningful. Among many available stemming algorithms, the one used for this project is Porter’s Stemmer algorithm.
- **Removing Unicode:** Some tweets could contain a Unicode character that is unreadable when we see it on an ASCII format. Filtering of such code is done in this process.
- **Removing Emojis:** Some tweets contains emojis which is removed in the process

# 4. Word Embedding Techniques

In natural language processing(NLP), word embedding is a term used for the representation of words for text analysis, typically in the form of a real-valued vector that encodes the meaning of the word such that the words that are closer in the vector space are expected to be similar in meaning. Word embeddings can be obtained using a set of language modelling and feature learning techniques where words or phrases from the vocabulary are mapped to vectors of real numbers.

## 4.1 tf-idf(Term frequency-inverse document frequency)

TF-IDF stands for Term Frequency Inverse Document Frequency of records. It can be defined as the calculation of how relevant a word in a series or corpus is to a text. The meaning increases proportionally to the number of times in the text a word appears but is compensated by the word frequency in the corpus (data-set). We are using the tf-idf vectorizer as it gives more importance to frequently occurring words and helps in identifying the main context of the sentence.

## 4.2 fastText

fastText is another word embedding method that is an extension of the word2vec model. Instead of learning vectors for words directly, fastText represents each word as an n-gram of characters. So, for example, take the word, "*artificial*" with n=3, the fastText representation of this word is <*ar, art, rti, tif, ifi, fic, ici, ial, al*>, where the angular brackets indicate the beginning and end of the word. Here for our project, we are using a pretrained fastText model trained on 2 million word vectors on Common Crawl. As we know pretrained models help in better performance results, avoid overfitting and reduce training time.

## 4.3 GloVe

Glove captures both global statistics and local statistics for generating the embeddings. The gloVe is a count-based model, which learns vectors by performing dimensionality reduction on a co-occurrence counts matrix. First, they construct a large matrix of co-occurrence information, which contains the information on how frequently each "word" (stored in rows), is seen in some "context" (the columns). Here for our project, we are using pretrained Stanford's GloVe 100d word embedding model as we know pretrained models help in better performance results, avoids overfitting, and reduce training time.

# 5. Models

## 5.1 Machine Learning Model

Machine learning is a branch of artificial intelligence (AI) and computer science which focuses on the use of data and algorithms to imitate the way that humans learn, gradually improving its accuracy.

### 5.1.1 SVM(Support Vector Machine)

SVM algorithm is a supervised learning algorithm categorized under Classification techniques. It is a binary classification technique that uses the training dataset to predict an optimal hyperplane in an n-dimensional space. This hyperplane is used to classify new sets of data. Being a binary classifier, the training data set the hyperplane divides the training data set into two classes. In our project, we will detect if the news is fake or real as we have 2 classes.

### 5.1.2 Logistic Regression

This type of statistical model (also known as the logit model) is often used for classification and predictive analytics. Logistic regression estimates the probability of an event occurring, such as voted or didn't vote, based on a given dataset of independent variables. Since the outcome is a probability, the dependent variable is bounded between 0 and 1. In our case, we will detect whether the tweet is fake or real.

## 5.2 Neural Network Models

### 5.2.1 LSTM

Long-Short Term Memory (LSTM) is an advanced version of Recurrent Neural Network (RNN), which makes it easier to remember past data in memory. LSTM is a well-suited model for sequential data, such as data for NLP problems. Thus, we utilized LSTM to perform fake news detection. In our model we have added Dense Layer, with activation relu followed by Dropout, and added Dense layer with activation sigmoid with compiling learning rate as 0.0001 and loss as binary cross entropy.

### 5.2.2 Bidirectional LSTM

Bidirectional LSTM (BiLSTM) consists of two LSTMs: one taking the input from a forward direction, and the other in a backward direction. BiLSTM effectively increases the amount of information available to the network, improving the context available to the algorithm. In our model we have added Dense Layer, with activation relu followed by Dropout, and added Dense layer with activation sigmoid with compiling learning rate as 0.0001 and loss as binary cross entropy.

## 5.3 Transformers

### 5.3.1 BERT:

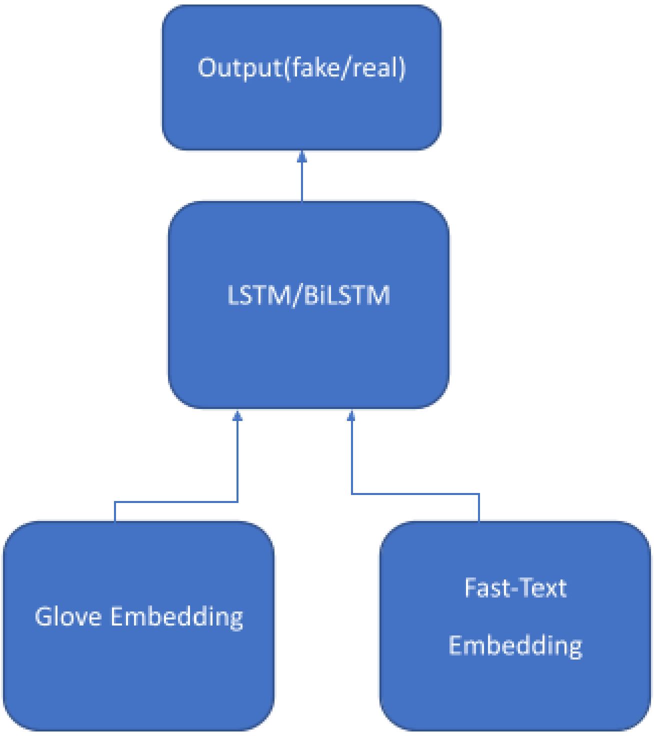
BERT. BERT-base [10] is a model that contains 12 transformer blocks, 12 self-attention heads, and a hidden size of 786. The input for BERT contains embeddings for a maximum of 512 words and it outputs a representation for this sequence. The first token of the sequence is always [CLS] which contains the special classification embedding and another special token [SEP] is used for separating segments for other NLP tasks. For a classification task, the hidden state of the [CLS] token from the final encoder is considered and a simple SoftMax classifier is added on top to classify the representation.

### 5.3.2 DistilBERT

DistilBERT offers a simpler, cheaper, and lighter solution that has a basic transformer architecture similar to that of BERT. Instead of distillation during the fine-tuning phase specific to the task, here the distillation is done during the pre-training phase itself. The number of layers is halved, and algebraic operations are optimized. Using a few such changes, Distil BERT provides competitive results even though it is 40% smaller than BERT.

## 6. Model Architecture

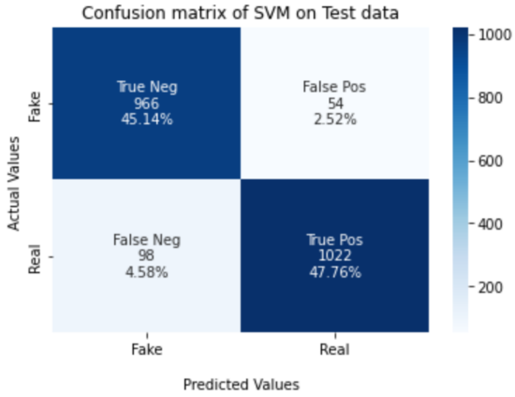
As per model architecture shown below its for neural network model as we can see we can use Glove or fastText Embedding Techniques followed by further we will implement LSTM and BI-LSTM

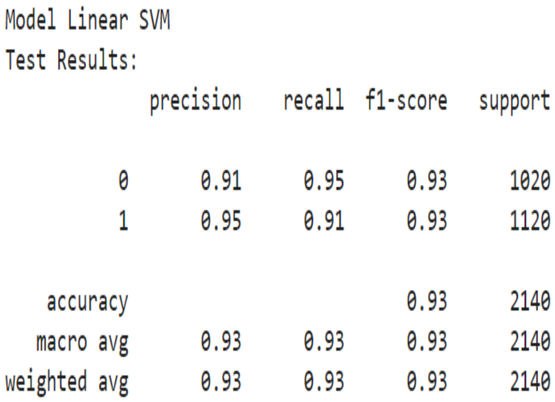


# 7. Experimental Results:

## 7.1 Machine Learning Models

### 7.1.1 Support Vector Machine

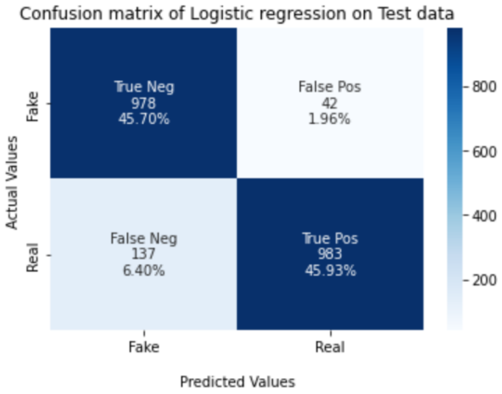


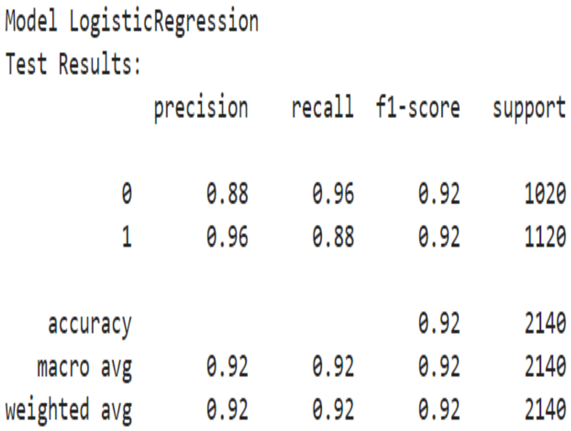


```
Model Linear SVM
Test Results:
              precision    recall  f1-score   support

           0       0.91      0.95      0.93      1020
           1       0.95      0.91      0.93      1120

    accuracy                           0.93      2140
   macro avg       0.93      0.93      0.93      2140
weighted avg       0.93      0.93      0.93      2140
```

**Fig 4:** Confusion matrix of SVM on Test data **Fig 5:** Classification report of SVM model on Test data

### 7.1.2 Logistic Regression





```
Model LogisticRegression
Test Results:
              precision    recall  f1-score   support

           0       0.88      0.96      0.92      1020
           1       0.96      0.88      0.92      1120

    accuracy                           0.92      2140
   macro avg       0.92      0.92      0.92      2140
weighted avg       0.92      0.92      0.92      2140
```

**Fig 6:** Confusion matrix of LR model on Test data **Fig 7:** Classification report of LR model on Test data

As we can analyze from Fig 4, 5, 6 and 7, the accuracy scores, confusion matrices and the classification reports of the two models, we can conclude that the Support Vector Classifier has outperformed Logistic Regression model in this task. The Support Vector classifier has given about 93% accuracy in classifying the fake news texts.

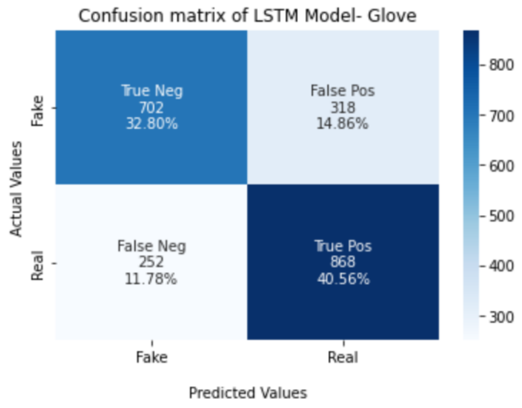
| Model | Accuracy | Precision | Recall | F1-Score |
|---|---|---|---|---|
| SVM Model | 93 | 93 | 93 | 93 |
| Logistic Regression | 92 | 92 | 92 | 92 |

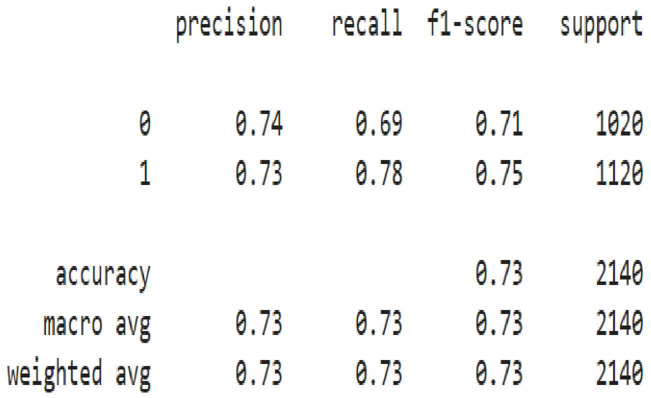
**Table 3:** Displays the evaluation metrics of SVM and Logistic Regression model

## 7.2 Neural Network Models

### 7.2.1 LSTM Model

### Glove Embedding



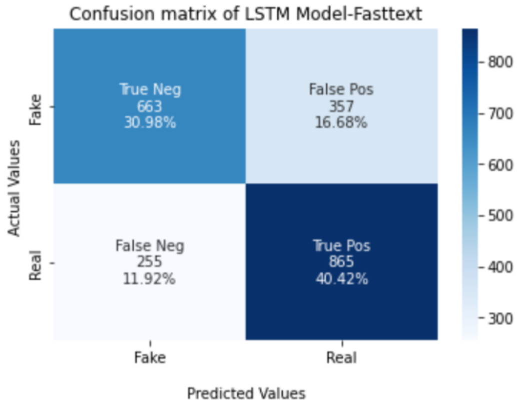


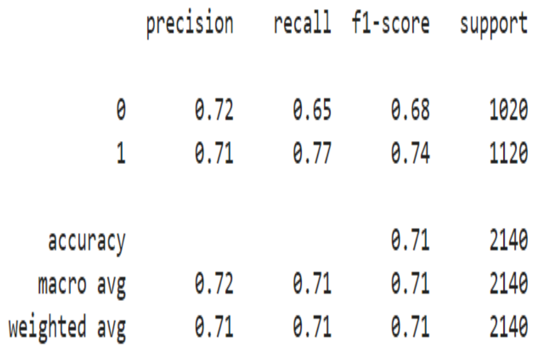
```
              precision    recall  f1-score   support

           0       0.74      0.69      0.71      1020
           1       0.73      0.78      0.75      1120

    accuracy                           0.73      2140
   macro avg       0.73      0.73      0.73      2140
weighted avg       0.73      0.73      0.73      2140
```

**Fig 8:** Confusion matrix of LSTM GloVe model

**Fig 9:** Classification report of LSTM GloVe model

### fastText Embedding





```
              precision    recall  f1-score   support

           0       0.72      0.65      0.68      1020
           1       0.71      0.77      0.74      1120

    accuracy                           0.71      2140
   macro avg       0.72      0.71      0.71      2140
weighted avg       0.71      0.71      0.71      2140
```

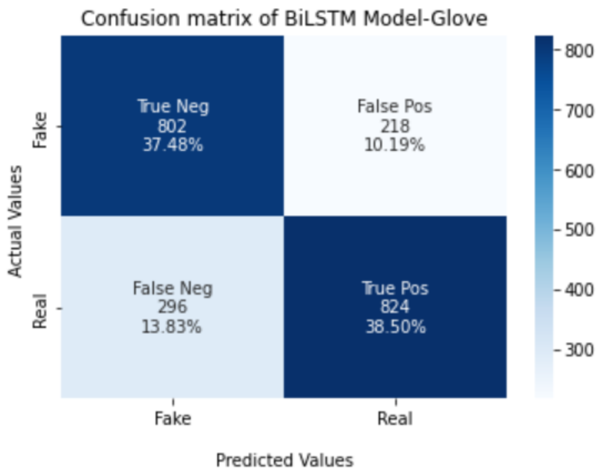
**Fig 10:** Confusion matrix of LSTM fastText model

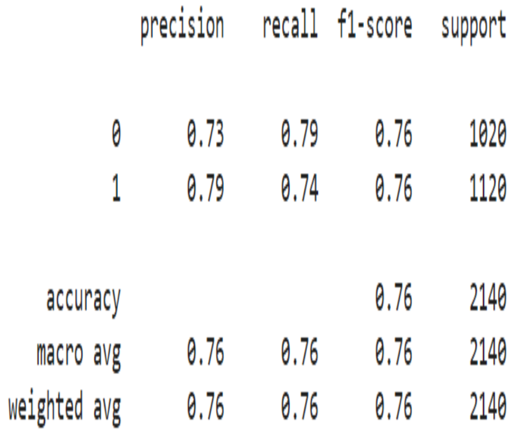
**Fig 11:** Classification report of LSTM fastText model

As we can analyze from Fig 8, 9,10 and 11, the accuracy scores, confusion matrices and the classification reports of the models, with respect to 2 embedding Technique we can conclude that LSTM model with GLove Embedding Technique has outperformed LSTM with fastText embedding technique in this task.

### 7.2.2 Bi-LSTM Model

### GloVe Embedding

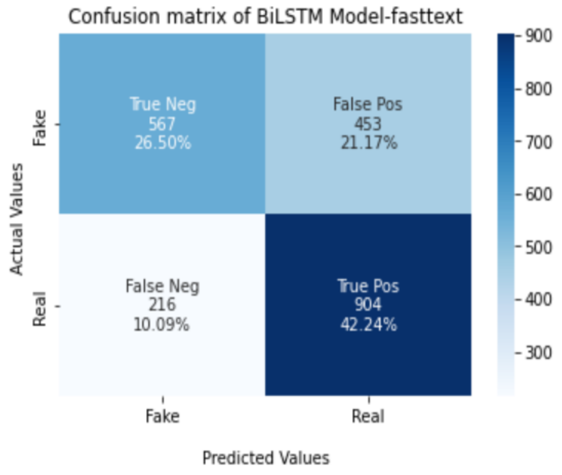


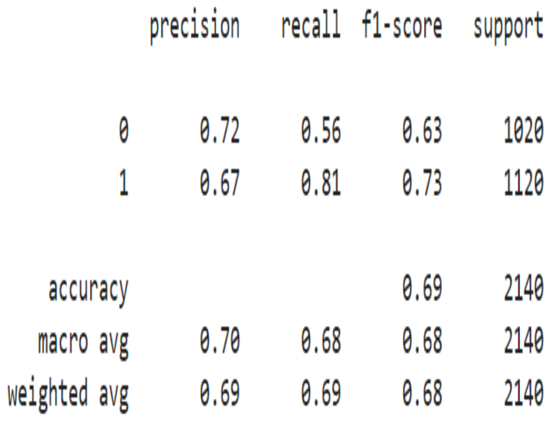


```
              precision    recall  f1-score   support

           0       0.73      0.79      0.76      1020
           1       0.79      0.74      0.76      1120

    accuracy                           0.76      2140
   macro avg       0.76      0.76      0.76      2140
weighted avg       0.76      0.76      0.76      2140
```

**Fig 12:** Confusion matrix of Bi-LSTM GloVe model **Fig 13:** Classification report of Bi-LSTM GloVe model

### fastText Embedding





```
              precision    recall  f1-score   support

           0       0.72      0.56      0.63      1020
           1       0.67      0.81      0.73      1120

    accuracy                           0.69      2140
   macro avg       0.70      0.68      0.68      2140
weighted avg       0.69      0.69      0.68      2140
```

**Fig 14:** Confusion matrix of Bi-LSTM fastText model **Fig 15:** Classification report of Bi-LSTM fastText model

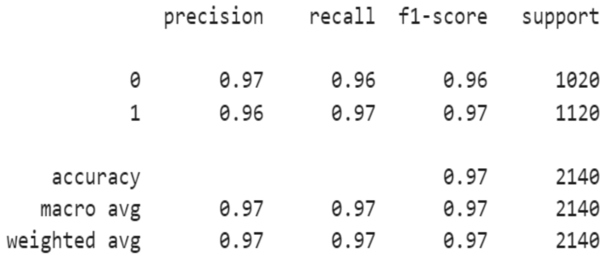
As we can analyze from Fig 12, 13, 14 and 15, the accuracy scores, confusion matrices and the classification reports of the models, with respect to 2 embedding Technique we can conclude that Bi-LSTM model with GLove Embedding Technique has outperformed Bi-LSTM with fastText embedding technique in this task.

| **Models** | **Embedding Technique** | **Accuracy** | **Precision** | **Recall** | **F1-Score** |
|---|---|---|---|---|---|
| **LSTM** | **GloVe** | **73** | **73** | **73** | **73** |
| **Bi-LSTM** | **GloVe** | **76** | **76** | **76** | **76** |
| **LSTM** | **fastText** | **71** | **71** | **71** | **71** |
| **Bi-LSTM** | **fastText** | **69** | **68** | **69** | **69** |

**Table 4:** Above table displays the summary of Neural Network model performance metrics results

## 7.3 Transformer Models

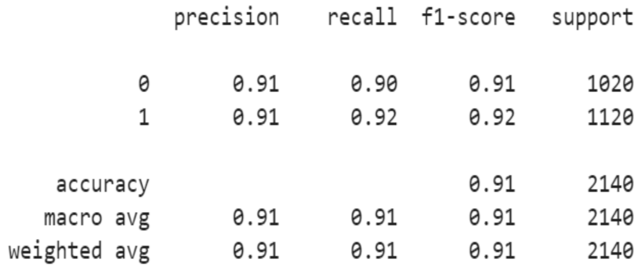
### 7.3.1 BERT (cased)



```
              precision    recall  f1-score   support

           0       0.97      0.96      0.96      1020
           1       0.96      0.97      0.97      1120

    accuracy                           0.97      2140
   macro avg       0.97      0.97      0.97      2140
weighted avg       0.97      0.97      0.97      2140
```

**Fig 16:** Classification report of BERT model

### 7.3.2 DistilBERT (cased)



```
              precision    recall  f1-score   support

           0       0.91      0.90      0.91      1020
           1       0.91      0.92      0.92      1120

    accuracy                           0.91      2140
   macro avg       0.91      0.91      0.91      2140
weighted avg       0.91      0.91      0.91      2140
```

**Fig 17:** Classification report of Distilbert model

As we can analyze from Fig 16 and 17 the accuracy scores and the classification reports of the two models, we can conclude that the BERT has outperformed Distilbert model in this task. The BERT has given about 97% accuracy in classifying the fake news texts.

| Model | Accuracy | Precision | Recall | F1 Score |
|---|---|---|---|---|
| **BERT** | **97** | **97** | **97** | **97** |
| **Distilbert** | **91** | **91** | **91** | **91** |

**Table 5:** Above table displays the summary of Transformers performance metrics results

# 8. Conclusion

| Model | Accuracy | F1 score | Precision | Recall |
|---|---|---|---|---|
| **SVM** | **93** | **93** | **93** | **93** |
| **BERT** | **97** | **97** | **97** | **97** |

**Table 6:** Above table displays the summary of best performing models

Scenario 1:
When a company has low budget ?
Scenario 2:
When a company has high budget?

As we can see transformer model BERT has high accuracy compared to Machine Learning model. So in our case, high budget project company will opt for BERT Transformer model and well in case of low budget project Machine Learning model will be opted and further trained with huge amount data to increase Machine Learning model accuracy

# 9. Future Work

**Word Embedding Technique:** By implementing state-of-the-art pre-trained model, ElMo embedding technique which has been created by Allen NLP, further improvement of model performance can be achieved as it has outperformed both word2Vec and GloVe

**Transformer Models:**We can implement ALBERT, ELECTRA, RoBERTa other Encoder models to increase the performance of our model

**Increase data to improve Model Performance**

## 10. Appendix

Please do find the below link for implementation of Machine Learning, Neural Network and Transformer Models code

**https://github.com/Sanchana1997/Covid19_Fake_News_Detection**

## 11. Reference

[1] Wani, A., Joshi, I., Khandve, S., Wagh, V., &amp; Joshi, R. (1970, January 1). Evaluating deep learning approaches for covid19 fake news detection. SpringerLink. Retrieved August 14, 2022, from https://link.springer.com/chapter/10.1007/978-3-030-73696-5_15

[2] Patwa, P., Sharma, S., Pykl, S., Guptha, V., Kumari, G., Akhtar, M. S., Ekbal, A., Das, A., &amp; Chakraborty, T. (1970, January 1). Fighting an Infodemic: Covid-19 fake news dataset. SpringerLink. Retrieved August 14, 2022, from https://link.springer.com/chapter/10.1007/978-3-030-73696-5_3

[3] Wang, Y., Zhang, Y., Li, X., &amp; Yu, X. (2021, October 1). Covid-19 fake news detection using bidirectional encoder representations from Transformers based models. arXiv.org. Retrieved August 14, 2022, from https://arxiv.org/abs/2109.14816

[4] Sharif, O., Hossain, E., &amp; Hoque, M. M. (2021, January 9). Combating hostility: Covid-19 fake news and hostile post detection in social media. arXiv.org. Retrieved August 14, 2022, from https://arxiv.org/abs/2101.03291v1