National College of Ireland

Fraudulent News Detection on Social Media

MSc Research Project
Data Analytics

Archana Uday Mahajan
Student ID: x20198825

School of Computing
National College of Ireland

Supervisor: Prof. Taimur Hafeez

National College of Ireland
Project Submission Sheet
School of Computing

Student Name:	Archana Uday Mahajan
Student ID:	X20198825
Programme:	Data Analytics
Year:	2022
Module:	MSc Research Project
Supervisor:	Prof. Taimur Hameez
Submission Due Date:	15/08/2022
Project Title:	Fraudulent News Detection on Social Media
Word Count:	5843
Page Count:	20

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	Archana Uday Mahajan
Date:	15th August 2022

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	☐
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	☐
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	☐

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Fraudulent News Detection on Social Media

Archana Uday Mahajan
20198825

Guide : Prof. Taimur Hafeez

Abstract - Today, technological advances and easy and open access to the Internet and social media have increased the international dissemination of information and news through social networks. In many cases, social media has become a primary source of information for the general public, governments and brands. Online posts are valuable and important because they can deliver news and reach even the most remote regions and people quickly and efficiently. These days, with the power of social media to reach and influence such a large audience, we realize that a lot of fake news and information is causing confusion and disorder in people's minds. The objective of this research is to identify fraudulent news in social media posts, to increase the reliability of online news, and determine how text analysis and machine or deep learning algorithms will work together to make the outcome more accurate. This research focuses on using natural language processing, text recognition, analytics, and machine learning techniques to separate fraudulent and genuine content from a dataset created by combining various Kaggle datasets, with an accuracy score of 97% from ML models and 94% using BERT.

Keywords - NLP, Machine Learning, LSTM, Naive Bayes, BERT, RNN

Contents

1 Introduction

In the 21st century, the rise of social media and the Internet has allowed information and news to spread rapidly among individuals around the world. Online posting is valuable and important because it can deliver news and reach even the most remote regions and people quickly and efficiently. Most of this information is provided by leading social media companies in online networks such as Twitter, Instagram, Facebook and WhatsApp. But this comes at a price. It is the public dissemination of disinformation that has a significant impact on people's political, economic and social lives (Scheufele and Krause, 2019). It can also lead to violence, political interference, division, etc. According to The Washington Post (Fisher and Cox, 2018), a man with a semi-automatic rifle walked into a Washington, D.C. pizza parlor and opened fire after being convinced of a fake message retweeted by The Washington Post. And so do the many bots who claim Hillary Clinton was hiding a pedophile trafficking ring in her pizzeria. Therefore, as stated by (Agrawal et al., 2021), fraudulent messages have become a major problem and challenge in today's 'digital economy' world, and to overcome this, they use natural language processing and machine learning. This research discusses the role of the base approach of detection of abusive content in social media posts. Similarly, this proposal seeks to address this issue with research topics presented below.

Research Question: *To what extent can we apply text analysis algorithms to social media posts and apply natural language along with ML/DL techniques to detect fraudulent messages?*

This study uses the techniques given to answer previous research question. It aims to investigate the above issues and help people distinguish between real and fake news and maintain the integrity and credibility of social media. The survey describes how record feeds are processed and Social media posts are built by creating an empty data model and feeding it data from multiple ad-hoc sources. This has the desired result of extracting posts that contain deceptive messages.

The study is structured as follows. Section 2 describes related work from various papers on fraudulent message detection using text analysis, semantic analysis, and machine learning techniques by various researchers, and has three subsections starting with data collection and corpus building. is divided into This is a tedious and difficult task as the data must be related and machine learning and his NLP techniques must be applied to the constructed corpus. It also goes through the cons of the work done. This serves as the basis for Part 3, which discusses the research methodology and specifications of this research proposal and attempts to address the deficiencies of the previous section. Section 4 contains the survey design specifications and Section 5 describes the implementation. Section 6 presents the evaluation and results of the model used. Finally, Section 7 discusses research conclusions and future work, followed by references.

2 Related Work

Many significant writers have contributed to the field of fraud news detection; only a handful are covered here. Many articles, studies, and surveys have been published that employ various methods for identifying fraudulent postings, such as machine learning, artificial intelligence, neural networks, etc. This section reviews these works and compares the advantages and disadvantages of the ways used to identify fraud news, which will assist in addressing those difficulties and improve the outcome of this project.

2.1 Identification of Fraud News using ML and NLP techniques

Social media posts tend to be confusing and unstructured due to character limits, and may contain emoticons and URLs, making them unsuitable for using information as data for research. Tweets on Twitter may contain structured data, but this varies by tweeter. (Jayasiriwardene and Ganegoda, 2020) propose in their paper a keyword extraction solution for extracting important data from such posts. This can be used to identify fake news using NLP techniques. Keywords are a key determinant of article content, and you can use this strategy to effectively and efficiently search your data. Using Wordnet and the Stanford Core NLP Toolbox, a corpus of tweets on various topics was created, broad rather than domain-specific. A Turing test was used for the evaluation and an accuracy of 67.6% was achieved.

(Bara, 2021) use commercial ML and NLP techniques in a similar work to propose a theoretical method for developing a dynamic corpus for extracting fraud messages. Their main focus is multilingual submissions, and the process includes data collection, data analysis, data distribution via APIs, and a feedback system. This analysis helps uncover fraud stories, determine where they come from and what impact they have. This will update the database and allow new research to leverage previous material. They stressed that this needs to be confirmed in larger corpora and different languages. This can lead to storage and processing power issues.

(Vinothkumar et al., 2022) used a unique stacking approach on two datasets from KDnuggets and ISOT, but found that the results were inaccurate as they could only identify scam messages affecting a specific group; Therefore, they decided to get the dataset using real-time data. This allowed them to achieve an accuracy of 89.38 percent, a 33 percent improvement over previous results obtained with a word pack and TF-IDF technology. In a similar work (Kareem and Avan, 2019), used TF and TF-IDF on a Pakistani news corpus, which was then applied to seven different machine learning algorithms.The maximum accuracy was achieved using the K-NN method, which provided an accuracy of 70%. If a larger dataset had been used, the results might have improved.

In a different technique, (Smitha and Bharath, 2020) extracted data from 244 websites in their study, resulting in a data collection of 25k posts. They extracted the features using a count vector, word embedding, and TF-IDF mix and put them into seven machine learning methods. They evaluated the findings using accuracy, precision, F1 score, and recall

to determine the best algorithm for fraud news categorization. Their study (Jain et al., 2022) categorized fraud news using LSTM, Naive Bayes, and SVM algorithms. The Long Short-term Memory method scored the highest accuracy, with a precision level of 0.94.

It is time-consuming to apply one dataset to multiple algorithms and compare the results; in their paper (Kumar and Arora, 2021), they provide a comparison of various machine learning techniques that have been applied to different datasets from multiple sources, as well as discuss the challenges encountered in detecting fraud news. The dataset is sourced from Buzzfeed, Kaggle, ISOT, LIAR, PolitiFact, and other sites, and it is fed into a variety of machine learning algorithms, including SVM, KNN, RandomForest, XGBoost, and Decision Tree. In the end, it was determined that ensemble learner techniques had more accuracy than other methods because they train several models, which helps eliminate errors and enhance the overall performance of the model. This evaluation also shows that data from numerous sources might assist improve the model's accuracy.

Fraudulent message detection is another idea, as determining whether a particular post is fraudulent also requires looking at the context of the content. (Fahad et al., 2022) stated in their study that testing methods on biased and unbiased media sources and topics would be part of the longitudinal study. Then you can improve the prediction accuracy. These datasets can be used to build machine learning models that can be deployed as browser extensions. The extension not only suggests to the user possible fake news, but can also provide links to help clarify the information.

(Hirlekar and Kumar, 2020) explore in detail the numerous strategies and issues in applying these techniques to different datasets, as well as the accuracy of the models, in a review article. In addition to NLP and ML they looked for browser extensions and tools that could filter and notify the source of spam messages. These include STB, B.S. Detector, FraudrFact, etc. They also looked at the best construction techniques to achieve maximum precision. Deep learning approaches that can process large amounts of data in a short time have also been proposed.

(Ahn and Jeong, 2019) fine-tuned the dataset using a deep learning method called BERT. This model achieves an AUROC of 83.8 percent. The dataset is pre-trained using a Masked Languaged Model and the following sentence prediction model. They still face other obstacles, such as obtaining a complete dataset.

(Sawan et al., 2021) suggested a sentiment analysis approach for detecting fraud in Arabic Tweets. They investigated several NLP approaches, feature selection methods, and advanced ML algorithms to accomplish this goal. Recursive Feature Elimination was employed in conjunction with the Logistic Regression classifier to exclude uninformative features, which yielded the maximum accuracy of 82% and improved the model's overall performance. They explore the future use of larger datasets and deep learning technologies to increase the model's accuracy and effect.

SAME is a sentiment analysis model combined with a multimodal embedding approach

used to extract fraudulent messages (Cui et al., 2019). Using an end-to-end deep embedding system, we explored whether subconscious user sentiment can help distinguish between fake data and legitimate messages. This approach consists of three steps. We first use a multi-modal network to handle different moods of the data, then use an adversarial mechanism to learn the semantic space within the data source, and finally use a new regularization loss to find the significant Embed combinations closer together. together. They tested this hypothesis using two of his datasets, PolitiFact and GossipCop.

(Narang and Sharma, 2021) utilized a comparison analysis to analyze significant publications on research that deal with diverse and unique techniques for identifying fraud news. A comprehensive comparison of publications has also been undertaken, indicating that text content is often utilized to detect fraudulent news. As a result, NLP, Deep Learning, and ML approaches are widely employed. Sentiment analysis, which employs user sentiments in postings to classify news, is another central area of research on this subject. Most of the study is focused on English; however, it has expanded to other languages such as German, Chinese, Latin, and Slavic.

Figure 1 presents a summary of the literature review:

Paper Name	Year	Models	Dataset	Accuracy	Gaps
Lai et al., 2022	2022	Binomial Logistic Regression, Naive Bayes, Support Vector Machine, Random Forest	Kaggle, web scrapped	85%	Not used bidirectional encoder representations from transformers (BERT), applying LSTM sequences to sequences, implementing bigrams and trigrams in training traditional ML and neural network models.
Kozik et al., 2022	2021	CNN, BERT, LSTM	GRAFN, ISOT	80%	Larger Databases not used, which will be created from scratch by obtaining contents from publicly available websites using the web scraper techniques. Data classification will be made on the basis of the prevailing opinions about sources, i.e., the addresses of websites.
Granik et al., 2022	2022	Naive Bayes	2016 US Elections	95%	Used basic small database, and applied only one algorithm, no clear EDA and cleaning steps performed.
Shahid et al., 2022	2022	Survey Paper	Survey Paper	Survey Paper	Within ML framework, the common algorithms that have achieved better results include Neural Networks, Naive Bayes, Decision Trees and SVM. Platform Independent Classifiers, Multiple Types of Bot Detection, Multilingual Detection, Real Time Detection can be used
Fahad et al., 2022	2022	TF-IDF	WebScrapped	70%	Testing method on biased and unbiased media sources and topics will be a part of longitudinal research. Moreover, increasing the size of the training datasets may improve prediction accuracy. These data sets could be used to build a machine learning model that could be deployed as a browser extension. In addition to suggesting the possibility of fake news to the user, the extension could provide a link that helps clarify the information

Figure 1: A summary of the literature review

3 Methodology

According to literature reviews, most of the significant research on fraudulent message detection has been done by combining natural language processing and machine learning techniques to train datasets. This research proposal aims to draw on lessons learned, highlight their strengths and weaknesses, fill gaps from previous research, and improve project completion. This section details the methodological approaches and steps required to conduct the research project. The current study's technique is based on a hybrid of the KDD and CRISP-DM models.

Figure 2 shows the phases of the methodology:

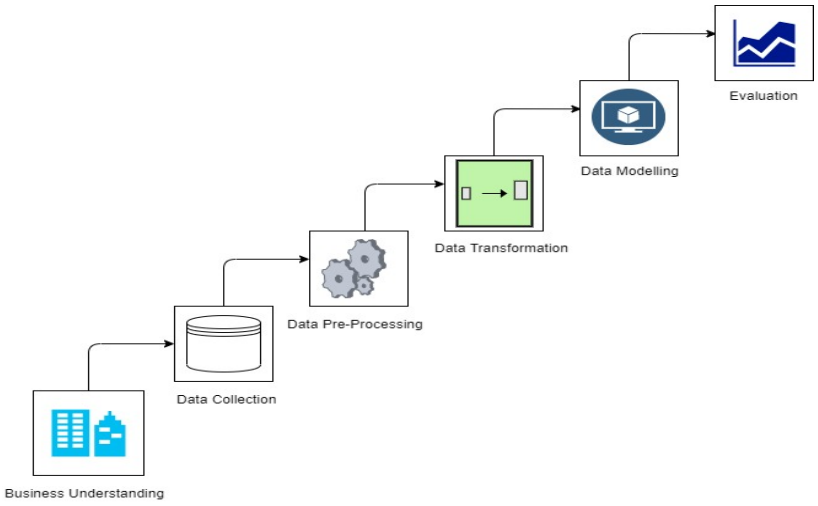


Figure 2: Research Methodology

- **Business Understanding -** Initially, it was required to comprehend the project's scope and domain knowledge. The first stage was to get business knowledge through a literature study that presents an overview of NLP and a summary of their gaps and drawbacks in earlier studies.
- **Data Gathering -** The data was collected from various Kaggle datasets and combined into one large dataset.
- **Data Pre-processing -** As the data is of text format, the pre-processing stage included removing StopWords, hashtags, HTTPS, slang, and Lemmatization. Bag-of-words was also used, which is an NLP method used to represent text as a bag of its words without considering word order or grammar. After that, the number of occurrences of a word was counted using the Count Vectorization method in Python's Scikit learn library and also the chi^2 method.
- **Data Transformation-** The similar words were then collected into documents using LDA, a topic modeling example, which was then transformed into tokens using the TF-IDF method.

- **Data Modeling -** The gathered, transformed, and tokenized output was then fed as input for multiple machine learning and deep learning models. The first model was Naive Bayes which achieved high accuracy, as the data was a little imbalanced even after pre-processing and removing duplicates. It was then fed to bi-directional LSTM an RNN model, overfitting, and BERT, a pre-trained model, which is smart enough to understand data imbalance and act upon it.
- **Evaluation -** The evaluation of the models was performed based on the accuracy, precision, f1-score, and loss. This is further discussed in the below section.

4 Design Specification

The architecture of the proposed framework is divided into three layers, as shown in Figure 3, consisting of the data, logical, and client tiers.

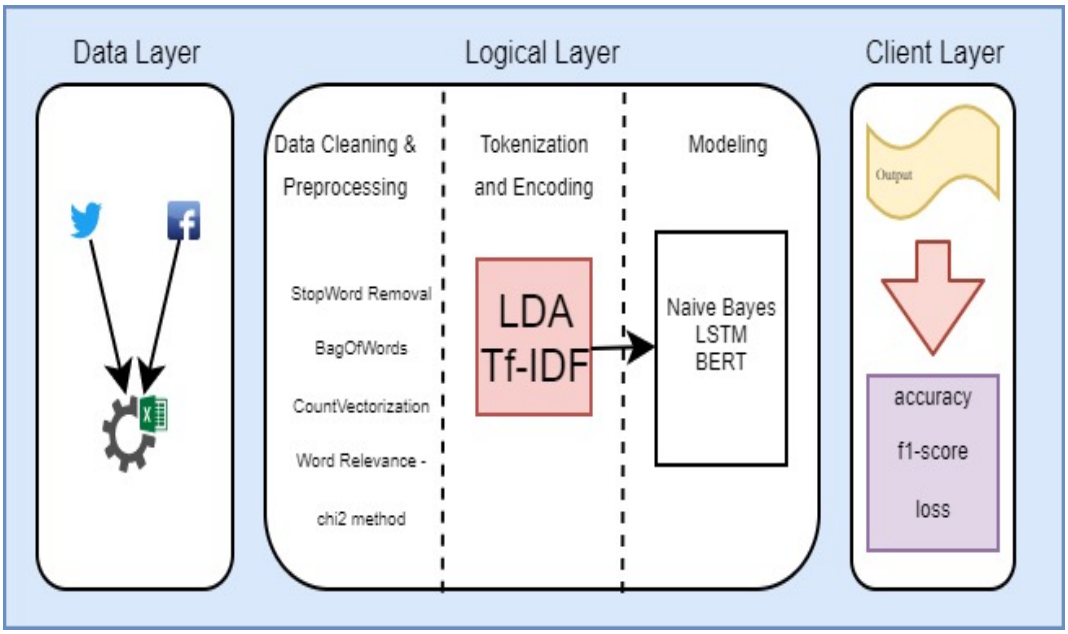


Figure 3: Project Architecture

As seen in the architecture, the data layer shows how data was sourced from multiple datasets from Kaggle and then stored in a single dataset, after annotation. The logical layer has a focus on the tokenizaion and encoding of the data afetr cleaning and preprocessing it using StopWord removal, BagOfWods, Count Vectorization and finding the words relevance. The tokenization is done using Latent Dirichlet allocation model which is an NLP process. It is among the most widely used topic modeling methods. Every document is composed of several words, and each topic is likewise made up of various terms. LDA's goal is to determine which topics a document falls under based on its words. Below diagrams shows the equation and representation of the LDA model.

```
p(word w with topic t) = p(topic t | document d) * p(word w | topic t)
```

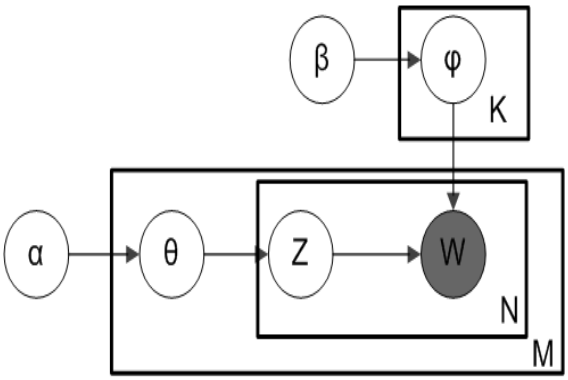


Figure 4: LDA Equation and Representation (Jalilifard et al., 2021)

Further, the documents were divided into tokens using a statistical method called TF-IDF tokenizer, whose task is to calculate how relevant a word is to a collection or a single document by multiplication of two metrics and the word's inverse document frequency over a group of documents. (Smitha and Bharath, 2020) inspired this method stated in the literature review. Below is the equation of the TF-IDF vectorizer:

$$tf\,idf\,(t, d, D) = tf\,(t, d)\,.\,idf\,(t, D)$$

Where:

$$tf\,(t, d) = log\,(1 + freq\,(t, d))$$

$$idf\,(t, D) = log\,(\frac{N}{count\,(d \in D : t \in d)})$$

This research also centers on implementing an upcoming NLP pre-trained model called Bidirectional Encoder Representations from Transformers, abbreviated as BERT, that has been proposed by Google emphasizing text data' context and then getting its' encoded value. this is a deep learning based model, built on a transformer architecture which is a network of encoder-decoder with self-attention for the encoder, and the decoder's side is attention. BERTs base model has 12 layers and 768 hidden units of the feed-forward network. This model was used because of the imbalanced dataset, as BERT is smart enough to understand the context and class weight. Naive Bayes and LSTM models are also used in this research, fed with an imbalanced dataset to show the advantage of the BERT model, the results of which have been compared in the evaluation section. Figure 5 shows the architecture of the BERT model:

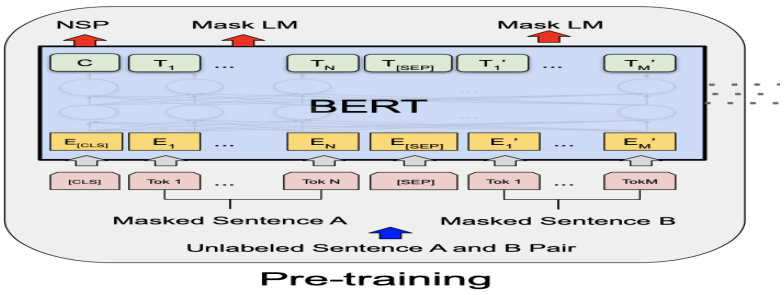


Figure 5: BERT Architecture - https://medium.com/dair-ai/a-light-introduction-to-bert

5 Implementation

The implementation of this research project was carried out from business understanding, data collection, preprocessing, tokenization, and encoding to the final modeling and result evaluation. Python 3.8.8 was used to implement the code, and the data was sourced into a single dataset after combining multiple datasets from Kaggle. Multiple libraries like Tensorflow, nltk, pandas, numpy, CountVector, sklearn, and Keras were used for data visualization and modeling. The training took time as the local machine was 8GB RAM, Intel Core i5 processor for Deep Learning models.

5.1 Data Collection and Exploration

The dataset used for this project was sourced from multiple datasets from Kaggle and stored in one dataset after proper annotation. After importing the dataset into the jupyter notebook, the following explorations were done to understand it and decide the preprocessing and feature extraction steps necessary to get effective results.

The first exploration was the twitter mentions in the fake and real news. The outcomes suggested that there were many more @ mentions in the fake news than the real news, meaning that the real news might have come from forums that are jumbled with Twitter as seen in Figure 6. This enforces that the dataset might be imbalanced:

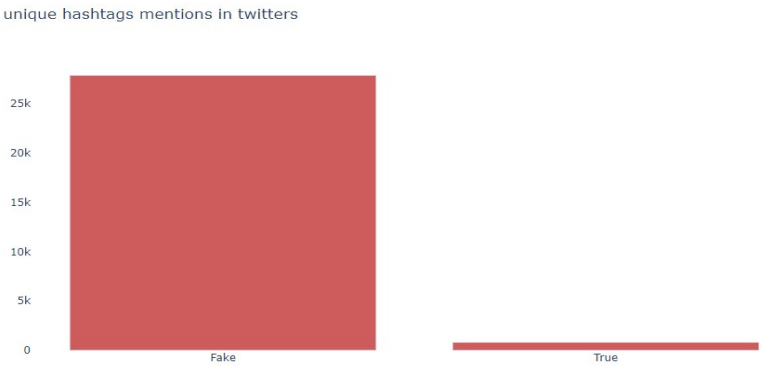


Figure 6: Hashtag mentions in posts

Although this does not conclude that the dataset does not fit to determine fraudulent news, preprocessing can remove the mentions. So, the next step was to check whether there was any contrasting distinction in the texts by checking the text size. The below diagram representing this shows that fake news has many more tokens as compared to real news, which is confusing as real news usually contains more details to give in-depth information to the reader, but this could be due to the fact established earlier that fake news dataset is a mix of news and tweets, seen in Figure 7:

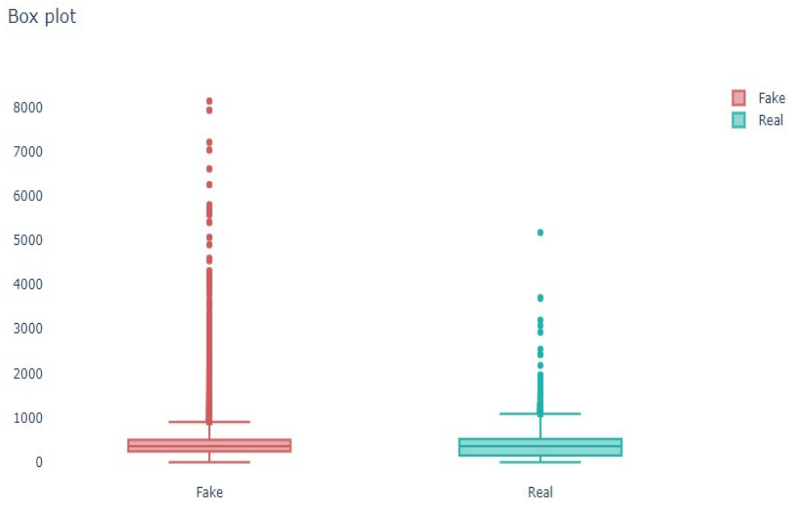


Figure 7: Text Size of Tweets

However, this does not prove a biased dataset, as this can be handled by using text that has fewer tokens after extracting features. Another simple thing to observe that could explain the bias is the existence of duplicate values, shown in Figure 8:

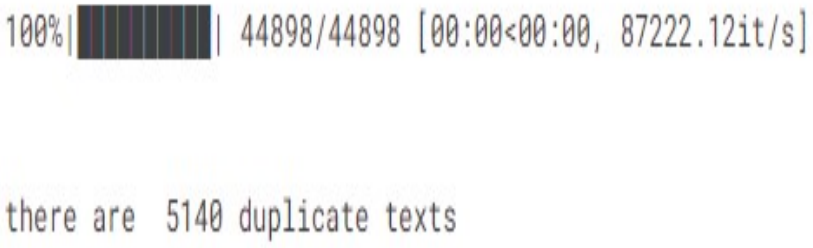


Figure 8: Duplicate Values

There are not enough duplicates, which leads the model to be overfitting, so the next step was to check the presence of tokens in the datasets. This showed that fake news has many tokens, such as slang, abbreviations, and informal writing. After that, in Figure 9 the occurrence of words that did not belong to the English Dictionary was observed.

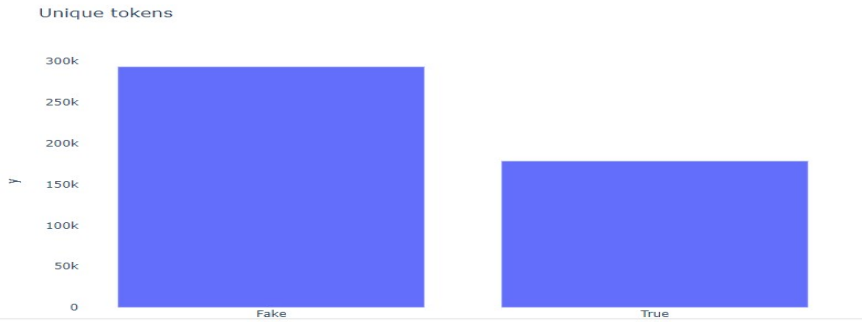


Figure 9: Unique Tokens

Here it was found out that above 70% words did not belong to the English Dictionary, which was used to verify, meaning many words were misspelled in Figure 10, but all of this can be solved using preprocessing, which has been explained in the next section.

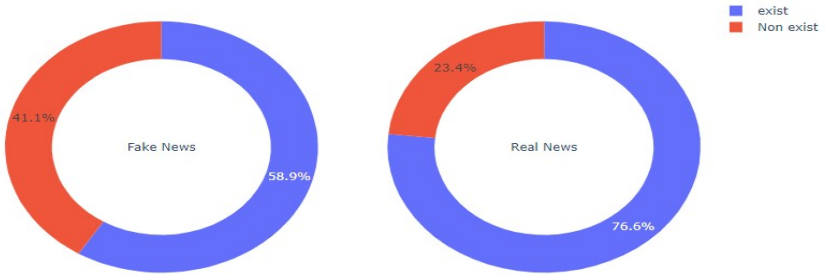


Figure 10: Misspelled Words

5.2 Data Preprocessing

The next step was to preprocess data that would clear some characters and normalize text for comparing it again, for which the nltk, pandas, and re packages were used. Stopwords removal, hashtags, HTTPS, and normalization were done on the dataset, in Figure 11.

```
import nltk
import re
tqdm.pandas()
def preprocess(df):
    stopwords = nltk.corpus.stopwords.words('english')
    df['text_pre'] = df['text']
    df['text_pre'] = df['text_pre'].progress_apply(lambda x : x.lower())
    df['text_pre'] = df['text_pre'].progress_apply(lambda x : x.split(" "))
    df['text_pre'] = df['text_pre'].progress_apply(lambda x : [item for item in x if item not in stopwords])
    df['text_pre'] = df['text_pre'].progress_apply(lambda x : " ".join(x))
#     df['text_pre'] = df['text_pre'].str.replace('@[^\s]+', "")
    df['text_pre'] = df['text_pre'].str.replace('https?:\/\/.*[\r\n]*', '')
    df['text_pre'] = df['text_pre'].str.normalize('NFKD').str.encode('ascii', errors='ignore').str.decode('utf-8')
    df['text_pre'] = df['text_pre'].str.replace('\d+', '')
    df['text_pre'] = df['text_pre'].str.replace('[^\w\s]', '')
    return df

fake = preprocess(fake)
true = preprocess(true)
```

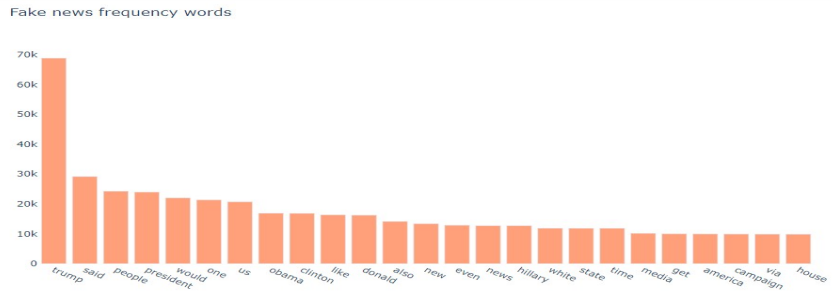
Figure 11: Preprocessing Code Snippet

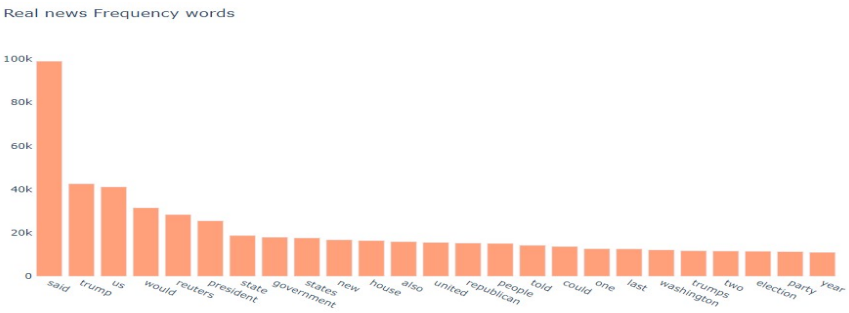
Figure 12 is the piechart created after applying preprocessing steps:

Figure 12: Misspelled Words after Preprocessing

5.3 Feature Extraction by Word Relevance

Feature extraction in NLP is a method used to select those tokens that prove to be most helpful in giving an accurate model. For this purpose, the feature_extraction.text was used from the CountVectorizer of Python, which gave the following output of relevant words from fake and real news datasets:





Although, it can be observed that the frequency of words from the fake and real datasets are not that different, the difference is not significant enough to establish any order. For which, the chi2 hypothesis test was used to get relevant words, the result of which is given in Figure 13:

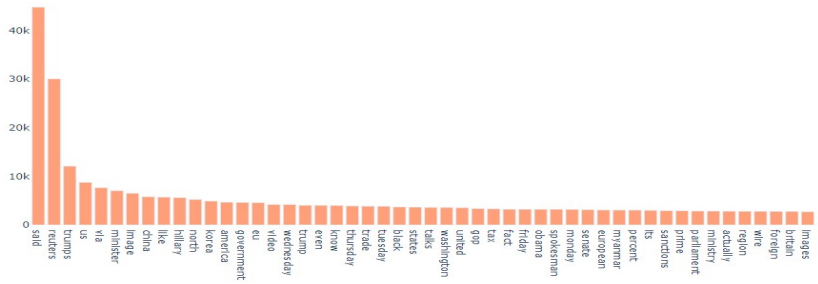


Figure 13: chi2 hypothesis for extracting relevant words

To understand in detail the context of the news, LDA, a popular topic modeling technique, was used to get the words from well-defined topics, for which the LatentDirichletAllocation was used from the sklearn.decomposition package. WordClouds were formed to get the relevant topics, the results of which are given below, which are given as input to the TF-IDF for vectorization of the textual data, explained in the next section:

Figure 14: Fake News Word Cloud for LDA (Using plot_clouds from Wordcloud package)

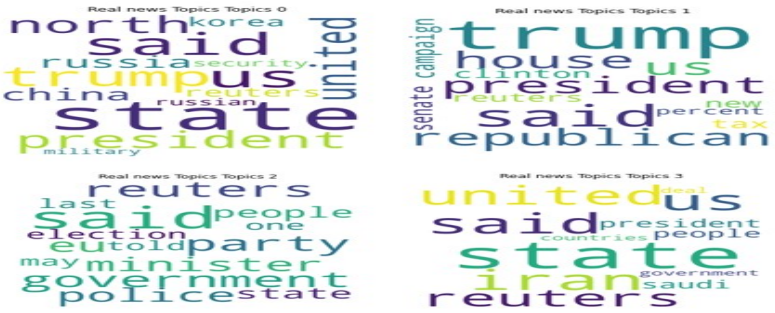


Figure 15: Real News Word Cloud for LDA (Using plot_clouds from Wordcloud package)

5.4 Model Building Approach

Since the data is of textual format, it was necessary to vectorize the text, assigning or converting it to a numerical value for input to the models. For this purpose, a TF-IDF

vectorizer was used. TF-IDF attempts to find terms that frequently appear in a document while omitting words such as “a” or “the” that are irrelevant and have no significance (Jalilifard et al., 2021). TF-IDF is divided into two sections, Term Frequency counts the number of times a word appears in a document, whereas Inverse Document Frequency selects insignificant terms like “a” or “the” and scales them down, enhancing the relevance of seldom used words. This approach is one of the most extensively utilized, and it often yields high levels of accuracy; it is consequently employed in this project, code snippet of which is given in Figure 16:

```
from sklearn.datasets import load_digits
from sklearn.feature_selection import SelectKBest, chi2
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.linear_model import LogisticRegression
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score


vect = TfidfVectorizer()
X = vect.fit_transform(concat2['text_pre'])
y = concat2['is_fake']
```

Figure 16: TF-IDF Code Snippet

The output of this vectorization is finally given as input to three models, Naive Bayes, which is a Machine Learning Model, as well as LSTM and BERT pre-trained model, which are Deep Learning based models, and a comparison between them is given in the next section of evaluation and results.

6 Evaluation and Results

The given research was conducted in a set of three experiments where the first experiment was conducted using an ML classification model called the Naive Bayes Model. The second experiment was conducted on a DL Model named the Bidirectional LSTM Model, where in both experiments, the dataset fed was imbalanced. The last experiment was done on a DL-based pre-trained model smart enough to understand the bias in the dataset, which is the BERT model. The results of which were compared and discussed in the Discussion section. The factors used for evaluation were accuracy, f1-score, and loss.

6.1 Experiment 1: Naive Bayes Classifier on Fraudulent News Dataset

The preprocessed and vectorized dataset was iterated over ten times, and each time the most relevant vectors were selected using the SelectKBest and chi2 methods of the sklearn.feature_selection package, whose results were then given to the classifier after splitting the data into training and testing datasets, with test_size=0.33. The final iteration gave an accuracy of 0.97, precision of 0.98, and f1-score = 0.97, given in depth in below classification report, also the below plot shows the accuracy scores of all ten iterations, given in Figures 17 and 18:

```
Accuracy score : 0.9752986434500911
Confusion Metrics :

 [[6905  134]
 [ 232 7546]]


Classification Report :

                precision    recall  f1-score   support

           0       0.97      0.98      0.97      7039
           1       0.98      0.97      0.98      7778

    accuracy                           0.98     14817
   macro avg       0.98      0.98      0.98     14817
weighted avg       0.98      0.98      0.98     14817
```

Figure 17: Naive Bayes Confusion Matrix and Classification Report

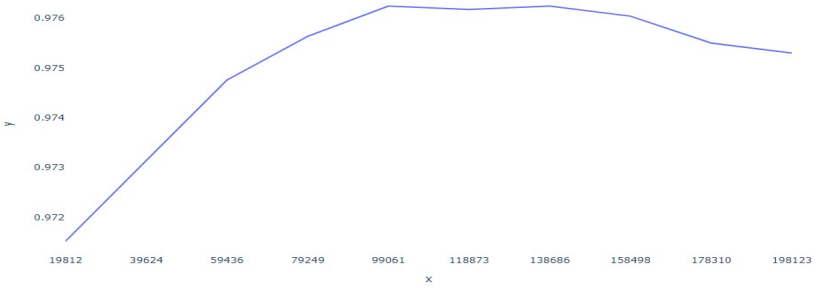


Figure 18: Naive Bayes Accuracy of ten iterations

6.2 Experiment 2: Bidirectional LSTM on Fraudulent News Dataset

The next model was a DL model, called Bidirectional LSTM, to get the difference between the ML and DL model, which is an RNN model and got an accuracy of 0.99 with 0.006 loss after five epochs. Figure 19 shows the summary and results of the Keras API.

```
Layer (type)                 Output Shape              Param #
=================================================================
embedding_10 (Embedding)     (None, None, 32)          320000

bidirectional_4 (Bidirectio  (None, None, 128)         49664
nal)

bidirectional_5 (Bidirectio  (None, 32)                18560
nal)

dense_20 (Dense)             (None, 64)                2112

dropout_2 (Dropout)          (None, 64)                0

dense_21 (Dense)             (None, 1)                 65

=================================================================
Total params: 390,401
Trainable params: 390,401
```

Figure 19: Bidirectional LSTM Model Summary

```
Epoch 1/5
1011/1011 [==============================] - 456s 435ms/step - loss: 0.2291 - accuracy: 0.8797 - val_loss: 0.0493 - val_accurac
y: 0.9860
Epoch 2/5
1011/1011 [==============================] - 435s 430ms/step - loss: 0.0307 - accuracy: 0.9934 - val_loss: 0.0159 - val_accurac
y: 0.9952
Epoch 3/5
1011/1011 [==============================] - 433s 428ms/step - loss: 0.0136 - accuracy: 0.9976 - val_loss: 0.0108 - val_accurac
y: 0.9979
Epoch 4/5
1011/1011 [==============================] - 433s 428ms/step - loss: 0.0086 - accuracy: 0.9983 - val_loss: 0.0180 - val_accurac
y: 0.9967
Epoch 5/5
1011/1011 [==============================] - 433s 429ms/step - loss: 0.0063 - accuracy: 0.9988 - val_loss: 0.0115 - val_accurac
y: 0.9979
```

Figure 20: Bidirectional LSTM Epochs Accuracy

6.3 Experiment 3: BERT on Fraudulent News Dataset

The final model used was BERT to overcome the dataset's bias, for which it was divided into three sets, the train, test, and validation dataset using the split function np.split(concat2.sample(frac=1), [int(.6*len(concat2)), int(.8*len(concat2))]). In BERT, the average loss was 0.074, and accuracy was 0.94. Below shown is the graph for loss and accuracy. The model was then validated on two randomly found articles on the internet, which gave an accuracy of fake at 95.24649381637573% and real at 88.04030418395996%, seen in Figures 21 and 22.

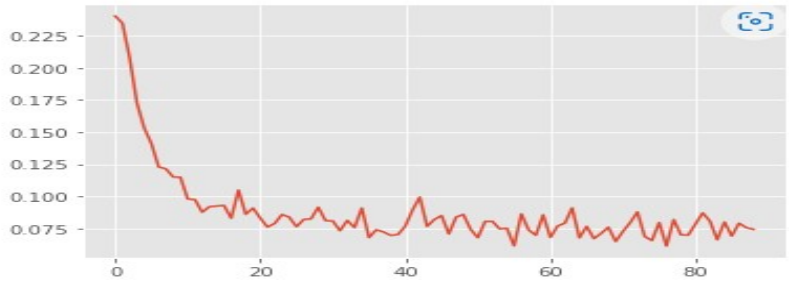


Figure 21: BERT loss graph plot

```
8400/8980. Current accuracy: 0.9478571428571428
8700/8980. Current accuracy: 0.9482758620689655
Accuracy on test data: 0.9478841870824053
```

```
27 A spokeswoman for the California Restau
28 """
29
30 test(fake1)
31 test(true1)
```

```
fake at 95.24649381637573%
real at 88.04030418395996%
```

Figure 22: BERT Test and Validation Accuracy

6.4 Discussion

The created dataset was preprocessed, and then after selecting the best features using the chi2 hypothesis was fed to three models, the results of which are given in the above section and have been summarized in the below table:

Model	Accuracy
Naïve Bayes	0.97
Bidirectional LSTM	0.99
BERT	0.94

Figure 23: Models Accuracy Summary

As seen from the above table, Naive Bayes and Bidirectional LSTM gets a score of 0.97 and 0.99, respectively, as they have been fed with a biased dataset and thus get an overfitting model. Although the BERT model gets a low accuracy, it was finally selected as the appropriate model for this research and dataset, as it is smart enough to understand the bias in the dataset and upsample and fit the model accordingly.

7 Conclusion and Futurework

The rise of social media and the internet in the twenty-first century has resulted in the quick distribution of information and news worldwide, spreading misinformation to the masses. As a result, this project explores the aforementioned issue and assists individuals in distinguishing between true and false news to safeguard social media's integrity. After getting data from various ad-hoc resources and piping them into one dataframe and performing many data explorations, data cleaning and preprocessing steps were applied to reduce the bias of the dataset. It was then fed to Naive Bayes, LSTM and BERT Models for traing and validation. To sum up, from this research, it can be concluded that as there was a bias in the dataset, even after relevant preprocessing and feature extraction and selection, the Naive Bayes and LSTM models were overfitting with 097 and 0.99 accuracy scores, respectively. BERT is thus suited to be the best model for this research as it performed well enough to understand the bias in the dataset and gave an accurate accuracy fit. For future work, the current research can be improved by applying SMOTE method to the dataset to upsample and reduce the bias. Also, data can be extracted from many other sources, like Instagram or WhatsApp.

Acknowledgement

I want to take this opportunity to thank and appreciate Prof. Taimur Hafeez for all his guidance, counselling, his insights, and detailed attention throughout this research.

References

Agrawal, C., Pandey, A. and Goyal, S. (2021). A Survey on Role of Machine Learning and NLP in fraud News Detection on Social Media.2021 IEEE 4th International Conference on Computing, Power and Communication
Technologies (GUCON).

Ahn, Y.-C. and Jeong, C.-S. (2019). Natural Language Contents Evaluation System for Detecting fraud News using Deep Learning. 2019 16th International Joint Conference on Computer Science and Software Engineering (JCSSE).

Bani-Hani, A., Adedugbe, O., Benkhelifa, E., Majdalawieh, M. and Al-Obeidat, F. (2020). A Semantic Model for Context-Based fraud News Detection on Social Media. [online] IEEE Xplore.
Available at: https://ieeexplore.ieee.org/document/9316504 [Accessed 9 Apr. 2022].

Bara, G.A. (2021). Building A Dynamic Corpus Of fraud News Using
Commercially Available Machine Translation and NLP Software. [online] IEEE Xplore.
Available at: https://ieeexplore.ieee.org/document/9475934 [Accessed 9 Apr. 2022].

Cui, L., Wang, S. and Lee, D. (2019). SAME. Proceedings of the 2019 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining.

Hirlekar, V.V. and Kumar, A. (2020). Natural Language Processing based Online fraud News Detection Challenges – A Detailed Review. [online] IEEE Xplore.
Available at: https://ieeexplore.ieee.org/document/9137915 [Accessed 9 Apr. 2022].

Kumar, S. and Arora, B. (2021). A Review of fraud News Detection Using Machine Learning Techniques. [online] IEEE Xplore.
Available at: https://ieeexplore.ieee.org/document/9532796 [Accessed 9 Apr. 2022].

Mugdha, S.B.S., Ferdous, S.M. and Fahmin, A. (2020). Evaluating Machine Learning Algorithms For Bengali fraud News Detection. [online] IEEE Xplore.
Available at: https://ieeexplore.ieee.org/document/9392662 [Accessed 9 Apr. 2022].

Narang, P. and Sharma, U. (2021). A Study on Artificial Intelligence Techniques for fraud News Detection. [online] IEEE Xplore.
Available at: https://ieeexplore.ieee.org/document/9673252 [Accessed 9 Apr. 2022].

Sawan, A., Thaher, T. and Abu-el-rub, N. (2021). Sentiment Analysis Model for fraud News Identification in Arabic Tweets. [online] IEEE Xplore. Available at: https://ieeexplore.ieee.org/document/
[Accessed 9 Apr. 2022].

Setiyaningrum, Y.D., Herdajanti, A.F., Supriyanto, C. and Muljono (2019).
Classification of Twitter Contents using Chi-Square and K-Nearest Neighbour Algorithm.

[online] IEEE Xplore.
Available at: https://ieeexplore.ieee.org/document/8884290 [Accessed 9 Apr. 2022].

Smitha, N. and Bharath, R. (2020). Performance Comparison of Machine Learning Classifiers for fraud News Detection. 2020 Second International Conference on Inventive Research in Computing Applications (ICIRCA).

Vinothkumar, S., Varadhaganapathy, S., Ramalingam, M., Ramkishore, D., Rithik, S. and Tharanies, K.P. (2022). fraud News Detection Using SVM Algorithm in Machine Learning. [online] IEEE Xplore.
Available at: https://ieeexplore.ieee.org/document/9740886 [Accessed 9 Apr. 2022].