FAKE NEWS DETECTION

A PROJECT REPORT

Submitted by

SHIVANSH GUPTA
[Reg No:RA2011026030111]

Under the guidance of

Mr. Gajender Kumar

(Assistant Professor, Department of Computer Science)

in partial fulfillment for the award of the degree of

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE & ENGINEERING

of

FACULTY OF ENGINEERING AND TECHNOLOGY

SRM INSTITUTE OF SCIENCE & TECHNOLOGY, NCR CAMPUS

NOV 2023

SRM INSTITUTE OF SCIENCE & TECHNOLOGY

(Under Section 3 of UGC Act, 1956)

BONAFIDE CERTIFICATE

Certified that this project report titled "**FAKE NEWS DETECTOR**" is the bonafide work of "**SHIVANSH GUPTA[Reg No: RA2011026030111]**", who carried out the project work under my supervision. Certified further, that to the best of my knowledge the work reported herein does not form any other project re- port or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

SIGNATURE

Mr. GAJENDER KUMAR
GUIDE
Assistant Professor
Department of Computer Science & Engineering

SIGNATURE

Dr. AVNEESH VASHISHTH
HEAD OF THE DEPARTMENT
Department of Computer Science & Engineering

Signature of the Internal Examiner

Signature of the External Examiner

ABSTRACT

The pervasiveness of misinformation in the digital realm poses a formidable threat to the truthness of information dissemination and the integrity of the public knowledge domain. This project addresses this critical challenge by developing and evaluating a model for fake news detection employing three distinct machine learning algorithms: K-Nearest Neighbors (KNN), Logistic Regression, and Random Forest. The project commences with the preprocessing of a collection of textual and contextual features extracted from an extensive dataset of news articles which is provided by Kaggle. Leveraging natural language processing techniques, the project seeks to unveil contextual cues that differentiate authentic news from fabricated content. The dataset encompasses a diverse array of titles,authors and writing styles to ensure the robustness and generalizability of the model. The K-Nearest Neighbours algorithm, a versatile non-parametric technique, will be employed to identify similarities between news articles and classify them accordingly based on their proximity in feature space. Logistic Regression, a widely utilized probabilistic model, will be used to model the probability of an article being genuine or deceptive based on the extracted features. Additionally, the Random Forest algorithm, renowned for its ensemble learning capabilities, will be implemented to augment predictive accuracy by aggregating the outputs of multiple decision trees. A rigorous comparative analysis of the three machine learning algorithms in terms of their accuracy, precision, recall, and F1 score, providing a comprehensive assessment of their performance will be performed.

ACKNOWLEDGEMENT

I would like to express my deepest gratitude to my guide, **MR.GAJENDER KUMAR** for his valuable guidance, consistent encouragement, personal caring, timely help and providing me with an excellent atmosphere for doing research. All through the work, in spite of his busy schedule, he has extended cheerful and cordial support to me for completing this research work.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF SYMBOLS

α, β	**Constants**
{f(t)}	**Fit Vector**
[K e]	**Element stiffness matrix**
[M e]	**Element mass matrix**
{q(t)}	**D Vector**
{q˙(t)}	**V Vector**
{q¨(t)}	**A Vector**
T	**The word**
d	**The document**
TF(t, d)	**The term frequency of t in d**
IDF(t)	**inverse document frequency of T**

CHAPTER 1

INTRODUCTION

The problem of fake news has become increasingly prevalent in recent years, particularly with the rise of social media. Fake news is defined as fabricated or misleading information presented as news. It is often spread intentionally to deceive people and can have a significant impact on public opinion and behavior.

It can erode trust in legitimate news sources. When people are constantly exposed to fake news, they may start to question the validity of all news sources, including those that are reliable. This can lead to a decline in trust in institutions and a rise in cynicism. It can spread misinformation and disinformation. Fake news can be used to spread false information about a variety of topics, including politics, health, and science. This can lead to people making decisions based on inaccurate information, which can have serious consequences. It can be used to manipulate people's emotions. Fake news is often designed to evoke strong emotions, such as fear or anger. This can make people more likely to share fake news stories without verifying their accuracy.

It can be used to harm individuals and groups. Fake news has been used to target individuals and groups with harassment and threats. It has also been used to incite violence and discrimination.

CHAPTER 2

LITERATURE SURVEY

2.1 Problems with Fake News

A number of studies have investigated the problem of fake news. Some of the key findings from these studies include:

- Fake news is often shared more widely than real news. This is because fake news is often more emotionally charged and more likely to evoke a strong reaction from readers.
- People are more likely to share fake news if they believe that it is true or if they agree with the message.
- People are less likely to share fake news if they are aware of the dangers of fake news and if they know how to identify it.

2.2 Existing approaches to address the issue of fake news

- Media literacy education. Media literacy education is the process of teaching people how to think critically about information and how to identify the signs of fake news.
- Fact-checking organizations. Fact-checking organizations are independent organizations that verify the accuracy of news stories. People can use these organizations to check the veracity of news stories before sharing them.

- Social media platforms. Social media platforms can take steps to identify and remove fake news from their sites. They can also work to educate their users about the dangers of fake news.
- Governments. Governments can play a role in addressing the problem of fake news by regulating social media platforms and by supporting media literacy education.

2.3 Challenges and future directions

The problem of fake news is complex and there is no easy solution. However, by taking a number of steps, we can work to address this problem and to create a more informed and responsible online environment

2.3.1 Challenges associated with Fake News

- Evasion of Detection: Fake news often employs sophisticated techniques to mimic credible sources and evade detection. This includes using similar fonts, layouts, and logos to reputable news organizations, making it difficult for users to distinguish between genuine and fabricated content.

- Rapid Dissemination: Social media platforms, with their vast reach and high-speed sharing capabilities, have facilitated the rapid dissemination of fake news. Once a piece of fake news is shared, it can reach millions of users within a matter of hours, making it challenging to contain its spread.

- Emotional Manipulation: Fake news often exploits human psychology, employing emotional appeals to fear, anger, or excitement to capture attention and encourage sharing. This emotional manipulation can cloud judgment and make it more likely for individuals to accept and share inaccurate information.

- Algorithmic Bias: Social media algorithms, designed to personalize content based on user preferences, can inadvertently amplify the spread of fake news. These algorithms may prioritize content that generates high engagement, regardless of its accuracy, leading to an echo chamber effect where users are exposed primarily to information that aligns with their existing beliefs.

- Financial Incentives: The creation and distribution of fake news can be motivated by financial gain. Malicious actors may spread fake news to drive traffic to their websites, generate advertising revenue, or promote specific products or services.

CHAPTER 3

SYSTEM DESIGN

3.1 LIBRARIES

In Python, libraries are collections of related modules that provide predefined functions and data structures for performing specific tasks. They are essential for extending Python's capabilities and enabling programmers to accomplish complex tasks with greater efficiency and ease.

3.1.1 Pandas

Pandas is a powerful, open-source Python library that provides high-performance, easy-to-use data structures and data analysis tools. It is designed to make working with "relational" or "labeled" data both easy and intuitive. Pandas is used for a wide range of data analysis tasks.

3.1.2 Numpy

NumPy (Numerical Python) is a fundamental library for scientific computing in Python. It provides a comprehensive set of functions and data structures for working with large multidimensional arrays and matrices. NumPy is widely used in various fields, including physics, engineering, finance, and data science.

3.1.3 Matplotlib

Matplotlib is a comprehensive library for creating static, animated, and interactive visualizations in Python. It is one of the most popular plotting libraries for Python and is widely used in data science, scientific computing, and software development. Matplotlib offers a variety of features for creating plots.

3.1.4 Pickle

Pickle is a module in Python that allows you to serialize and deserialize Python objects. This means that you can convert a Python object into a byte stream that can be stored on disk or transmitted over a network, and then later reconstruct the original object from the byte stream.

3.1.5 Seaborn

Seaborn is a high-level data visualization library for Python built on top of Matplotlib. It provides a more intuitive and aesthetically pleasing interface for creating plots, making it particularly well-suited for data exploration and communication. Seaborn is widely used in data science, statistics, and social science research.

3.1.6 Wordcloud

Wordcloud is a Python library that creates word clouds. Word clouds are visual representations of text data, where the size of each word indicates its frequency or importance. They are often used to visualize the most common words in a text corpus, and they can be used to identify trends and patterns in the data.

3.2 TOOLS USED FOR DATA MANIPULATION

When the data in the form of news article, title and author is inputted, it cannot be used as it is by the model instead it is lemmatised and then used as vector data.

3.2.1 Lemmatization

Lemmatization is the process of grouping together the inflected forms of a word so they can be analyzed as a single item, identified by the word's lemma, or dictionary form. In computational linguistics, lemmatization is the algorithmic process of determining the lemma of a word based on its intended meaning.

Lemmatization is a more complex process than stemming, which simply removes the inflectional endings from words. Lemmatization takes into account the grammatical context of the word, as well as its intended meaning, in order to determine the correct lemma.

3.2.1.1 Why is lemmatization important?

It improves information retrieval: When words are lemmatized, they can be matched more accurately to search queries. This can improve the results of search engines and other information retrieval systems.

It improves natural language processing (NLP) tasks: Lemmatization can improve the performance of NLP tasks such as machine translation and text summarization. This is because it ensures that words are being compared in a consistent way.
It can help to identify word relationships: Lemmatization can be used to identify word relationships, such as synonyms and antonyms. This information can be used to improve the performance of natural language processing tasks.

3.2.1.2 How does lemmatization work?

Lemmatization typically involves the following steps:

Tokenization: The text is broken down into individual words.

Part-of-speech tagging: Each word is assigned a part of speech, such as noun, verb, or adjective.

Morphological analysis: The word is analyzed to identify its morphological structure, such as its prefix, root, and suffix.

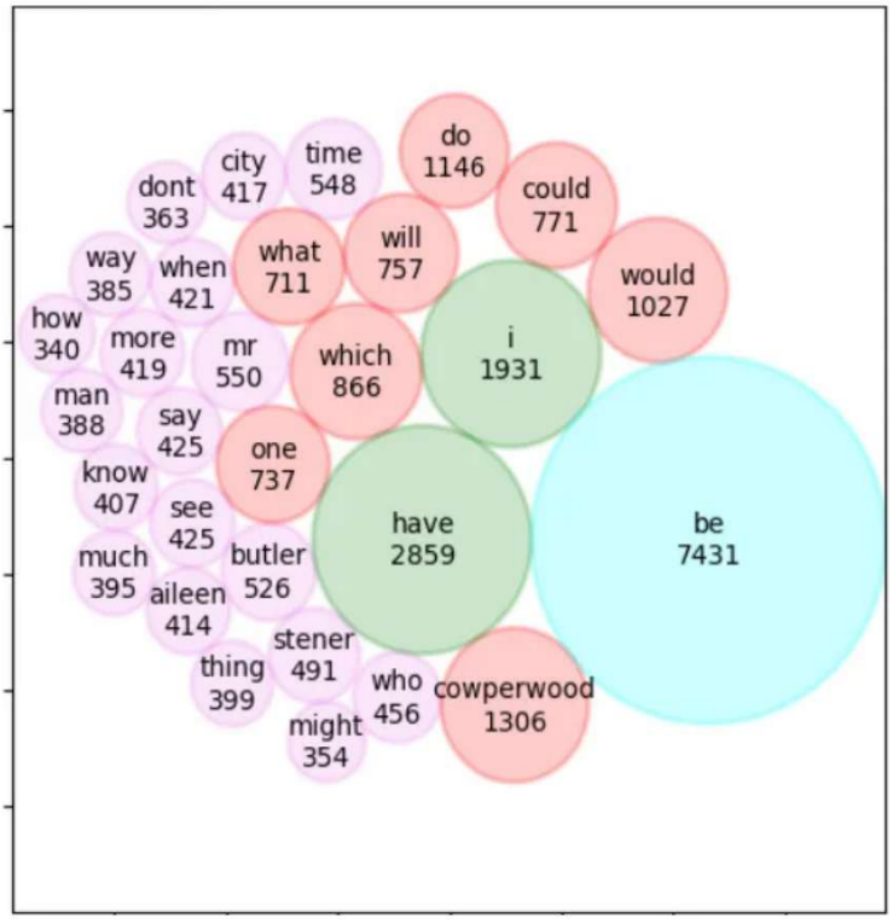
Lemmatization: The lemma of the word is determined based on its grammatical context and intended meaning.

3.2.1.3 NLTK Lemmatizer

The NLTK (Natural Language Toolkit) Lemmatizer is a tool that can be used to group together the inflected forms of a word so they can be analyzed as a single item. This is done by identifying the lemma, or dictionary form, of the word. The Lemmatizer can be used with any language, but it is most commonly used with English.

The NLTK Lemmatizer uses a combination of pattern matching and rule-based algorithms to determine the lemma of a word. It first looks up the word in a dictionary to see if it has a lemma entry. If the word does not have a lemma entry, the Lemmatizer applies a series of

rules to try to determine the lemma of the word.



[FIG 3.1]

As you can see, the Lemmatizer first looks up the word in a dictionary. If the word is found in the dictionary, the Lemmatizer returns the lemma of the word. If the word is not found in the dictionary, the Lemmatizer applies a series of rules to try to determine the lemma of the word.

The rules that the Lemmatizer applies are based on the morphological structure of the word. For example, the Lemmatizer will remove the -s ending from a noun to form the

plural form of the word. It will also remove the -ed ending from a verb to form the past tense form of the word.

3.2.2 TF-IDF

TF-IDF stands for Term Frequency-Inverse Document Frequency. It is a statistical measure that evaluates how relevant a word is to a document in a collection of documents. This is done by multiplying two different metrics:

Term frequency (TF): This is the number of times a word appears in a document. The more times a word appears in a document, the more relevant it is likely to be to that document. Inverse document frequency (IDF): This is a measure of how common or rare a word is across a set of documents. The rarer a word is, the more relevant it is likely to be to a document in which it appears.

It is represented as:

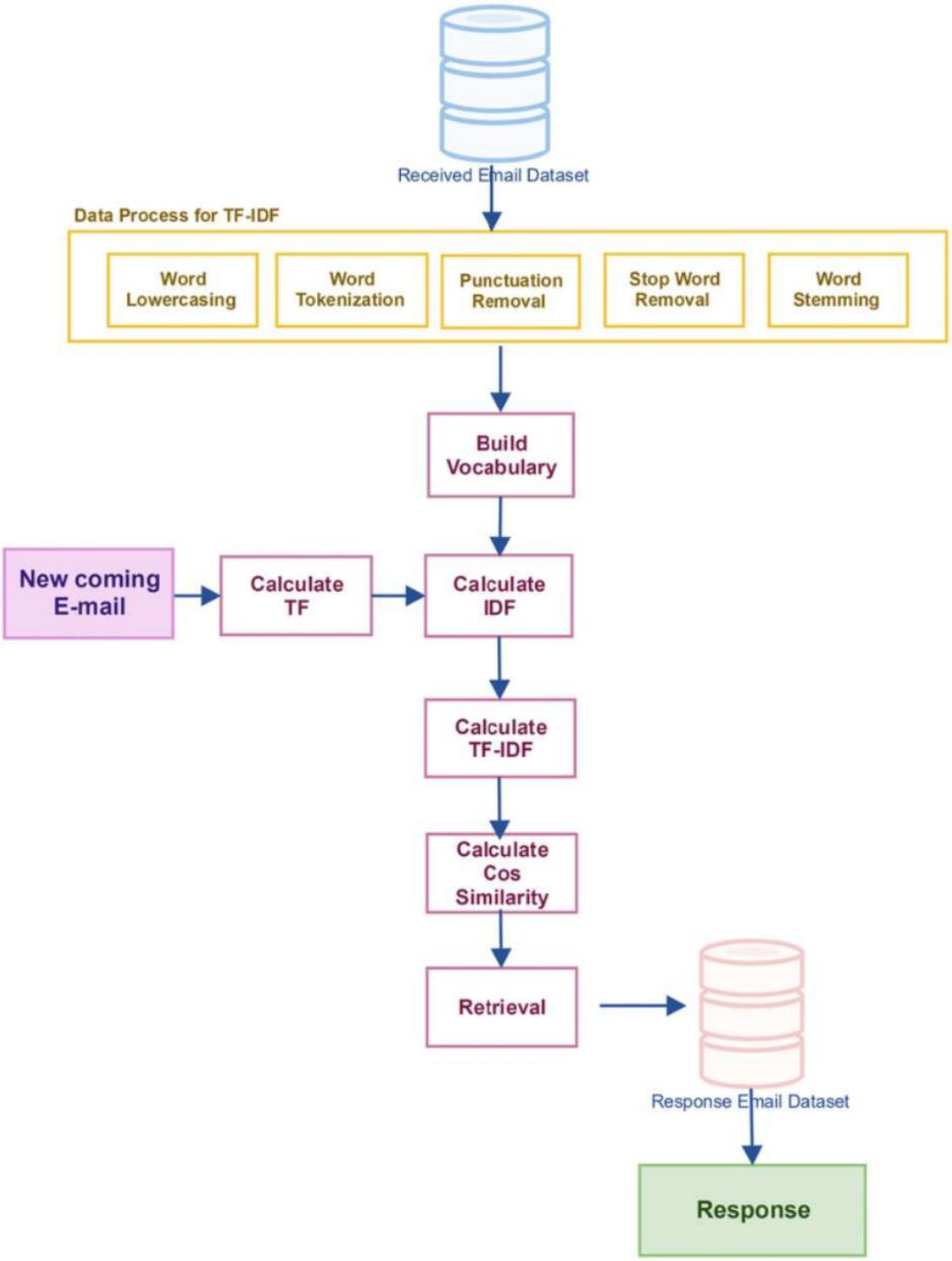
$$\textbf{TF-IDF}(t, d) = \textbf{TF}(t, d) * \textbf{IDF}(t)$$

where:

- t is the word
- d is the document
- TF(t, d) is the term frequency of t in d
- IDF(t) is the inverse document frequency of t

TF-IDF FLOWCHART

[FIG 3.2]



3.3 MACHINE LEARNING MODELS

This project utilizes a host of machine learning algorithms in tandem to train and then run the machine learning process required to discern fake news from genuine news articles.It uses three different models namely KNN(K-nearest neighbor), Logistic Regression and Random forest.

3.3.1 KNN(K-Nearest Neighbor)

K-Nearest Neighbors (KNN) is a non-parametric, instance-based learning algorithm used for classification and regression tasks. It is one of the simplest and most widely used algorithms in machine learning due to its ease of implementation and effectiveness in a variety of domains.

3.3.1.1 The KNN Algorithm

The KNN algorithm works by identifying the k nearest neighbors of a new data point in the training set. The neighbors are identified based on their Euclidean distance to the new data point. The Euclidean distance between two data points is calculated as follows:

$$d(x, y) = \text{sqrt}(\text{sum}((x\_i - y\_i)^2))$$

where:

- x and y are the two data points
- x_i and y_i are the values of the i-th feature for the two data points

Once the k nearest neighbors have been identified, the algorithm assigns a label to the new data point based on the majority class of its neighbors. In the case of regression, the algorithm predicts the value of the target variable for the new data point based on the average value of the target variable for its k nearest neighbors.

3.3.2 LOGISTIC REGRESSION

Logistic regression is a statistical model used for binary classification tasks. It is a powerful and versatile tool that is widely used in a variety of domains, including machine learning, data science, and finance.

3.3.2.1 The Logistic Regression Model

The logistic regression model is a linear model that predicts the probability of an event occurring. The event can be anything, such as whether an email is spam, whether a customer will earn, or whether a patient will develop a disease.

where:

- x and y are the two data points
- x_i and y_i are the values of the i-th feature for the two data points

Once the k nearest neighbors have been identified, the algorithm assigns a label to the new data point based on the majority class of its neighbors. In the case of regression, the algorithm predicts the value of the target variable for the new data point based on the average value of the target variable for its k nearest neighbors.

3.3.2 LOGISTIC REGRESSION

Logistic regression is a statistical model used for binary classification tasks. It is a powerful and versatile tool that is widely used in a variety of domains, including machine learning, data science, and finance.

3.3.2.1 The Logistic Regression Model

The logistic regression model is a linear model that predicts the probability of an event occurring. The event can be anything, such as whether an email is spam, whether a customer will earn, or whether a patient will develop a disease.

The logistic regression model is represented by the following equation:

$$P(y = 1 \mid x) = 1 / (1 + \exp(-\theta^T * x))$$

where:

- $P(y = 1 \mid x)$ is the probability of the event occurring given the input features x
- θ is the vector of model parameters
- x is the vector of input features
- The exponential term in the denominator of the equation ensures that the predicted probability is between 0 and 1.

3.3.2.2 Training the Logistic Regression Model

The logistic regression model is trained using a dataset of labeled examples. Each example in the dataset consists of a vector of input features x and a label y, which indicates whether the event occurred ($y = 1$) or not ($y = 0$).

The goal of training is to find the values of the model parameters θ that maximize the likelihood of the observed data. This is done using an optimization algorithm, such as gradient descent.

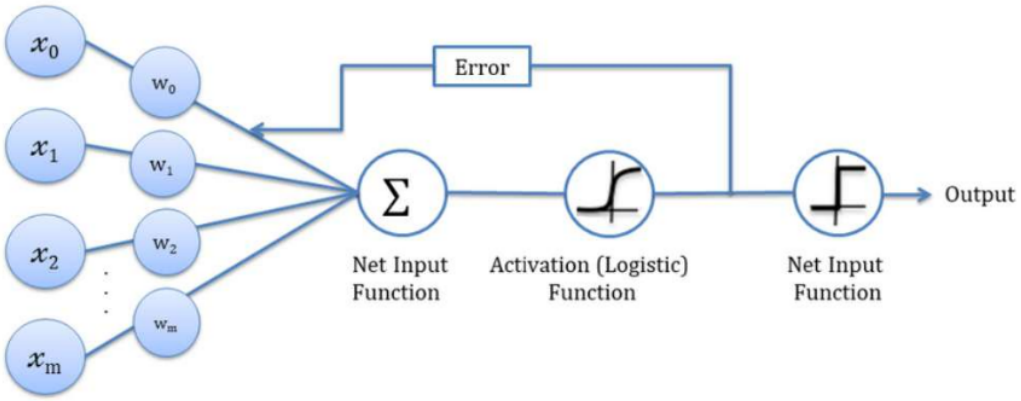


DIAGRAM FOR LOGISTIC REGRESSION [FIG 3.3]

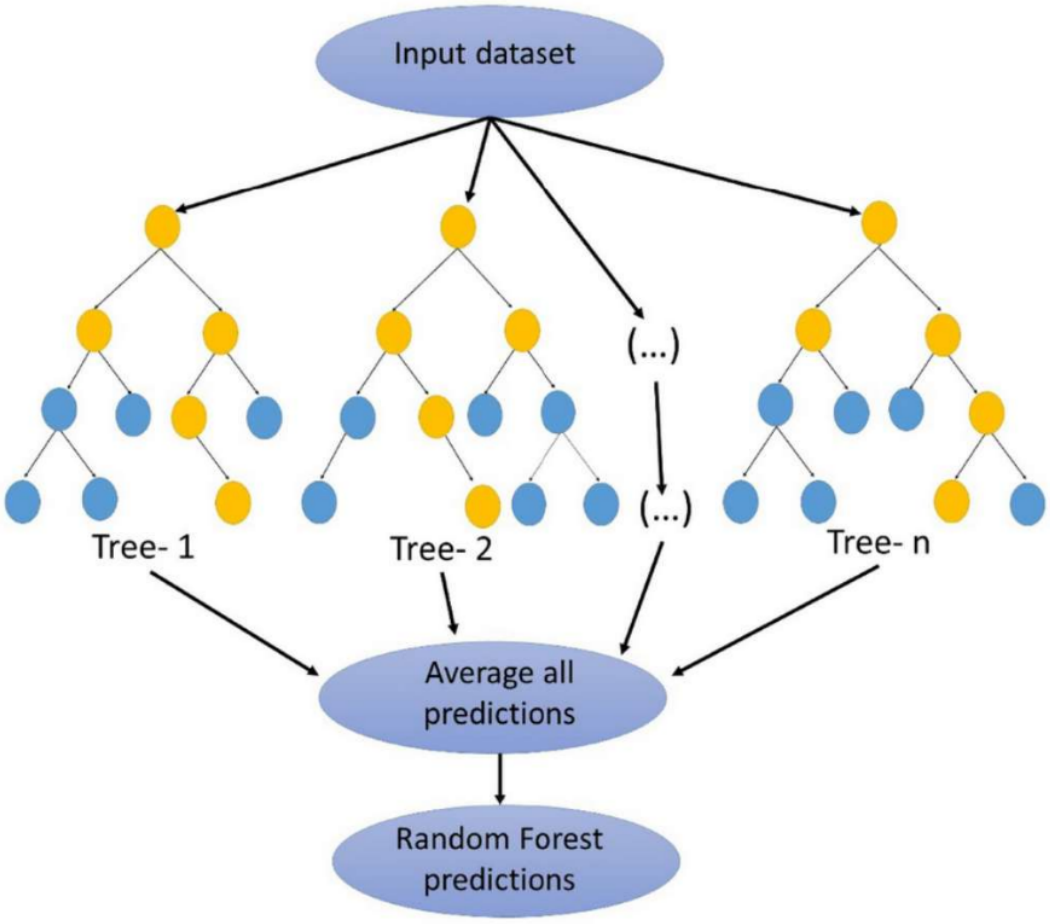
3.3.3 RANDOM FOREST

Random forest is an ensemble learning algorithm that combines multiple decision trees to produce a more accurate and robust prediction than a single decision tree. It is a popular algorithm for both classification and regression tasks, and it has been successfully applied to a wide variety of problems.

3.3.3.1 The Random Forest Algorithm

The random forest algorithm works by constructing a collection of decision trees, each of which is trained on a random subset of the data. The random subsets of the data are created by randomly sampling with replacement, which means that some data points may be included in multiple subsets.

Once the decision trees have been constructed, the random forest algorithm makes predictions by averaging the predictions of individual trees. In the case of classification, the class with the highest average predicted probability is selected. In the case of regression, the average of the predicted values from all trees is used.



[FIG 3.4]

DIAGRAM FOR RANDOM FOREST

CHAPTER 4

SYSTEM ANALYSIS

4.1 SPLITTING DATA INTO TEST DATA AND TRAINING DATA

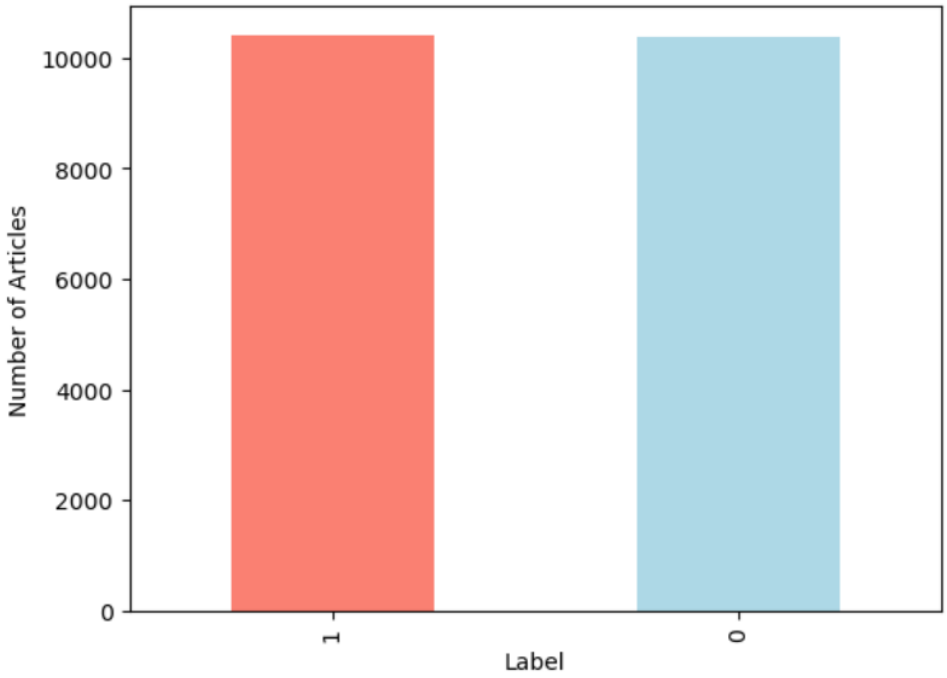
Here we will do some exploratory data analysis (EDA) and split the test data into genuine and fake news sets.

```
Int64Index: 20800 entries, 0 to 20799
Data columns (total 4 columns):
 #   Column  Non-Null Count  Dtype
---  ------  --------------  -----
 0   title   20242 non-null  object
 1   author  18843 non-null  object
 2   text    20761 non-null  object
 3   label   20800 non-null  int64
dtypes: int64(1), object(3)
memory usage: 812.5+ KB
```

[FIG 4.1]

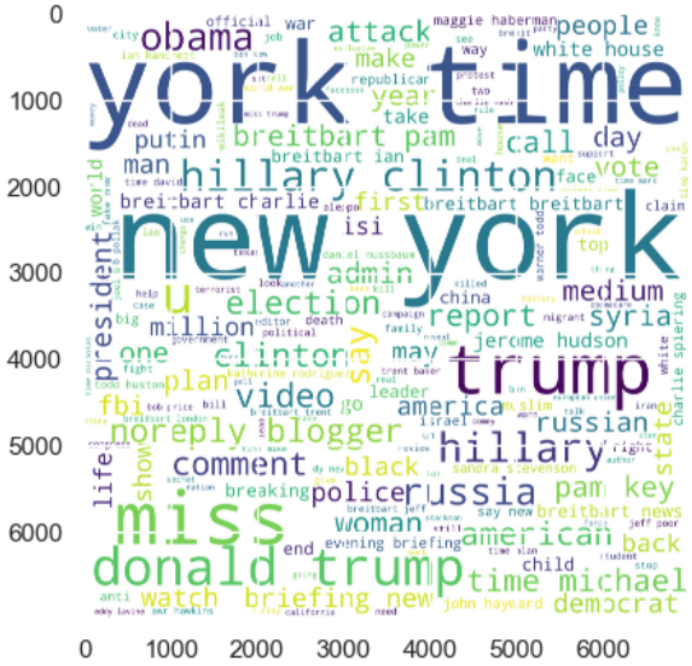
SYSTEM PERFORMANCE

4.1.1 Next we will split the data into two equal parts



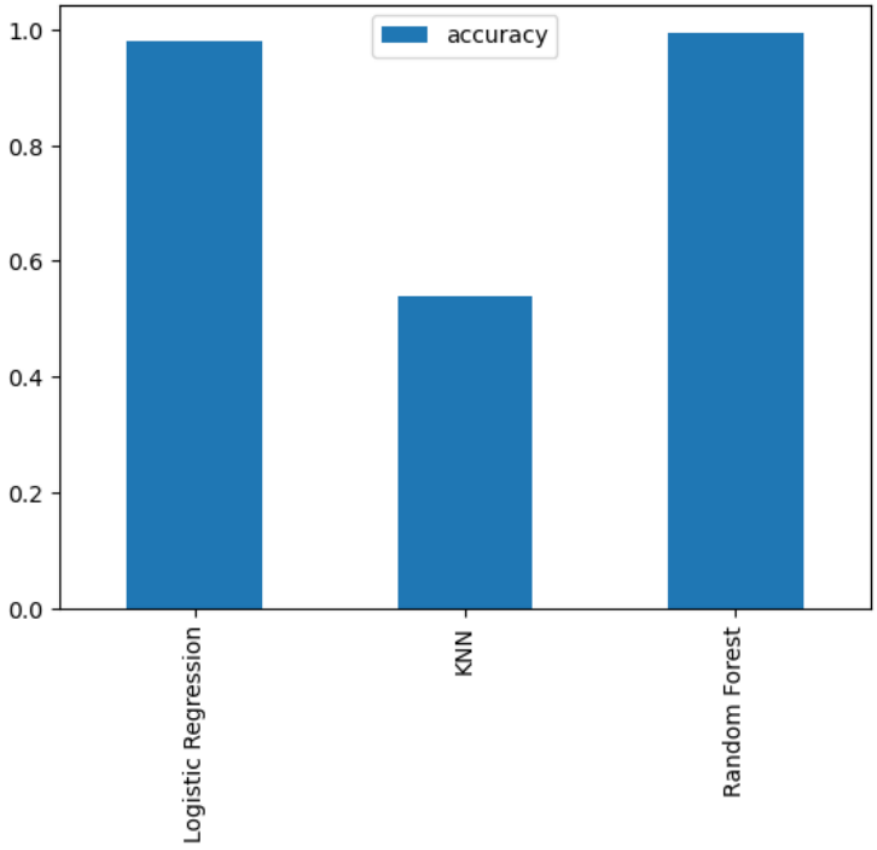
Data split into two parts with genuine and fake label [FIG 4.2]

4.2 GENERATING WORDCLOUD



Wordcloud from training data articles **[FIG 4.3]**

4.3 Model comparison



[FIG 4.4]

As we can see that KNN is the lowest performer here, we will continue to perform further tests.

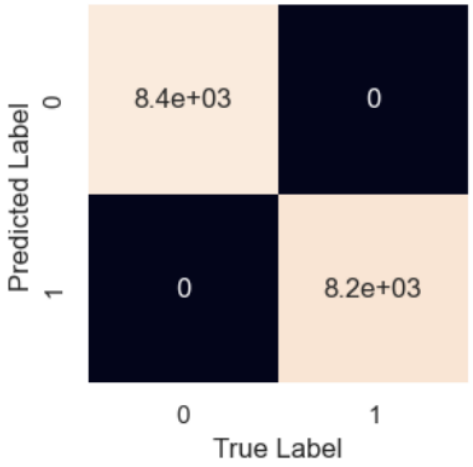
4.4 Model Evaluation

Here we will use a few tools to evaluate our model

4.4.1 Logistic Regression

Logistic regression: Accuracy **0.9940**

4.4.1.1 Confusion Matrix



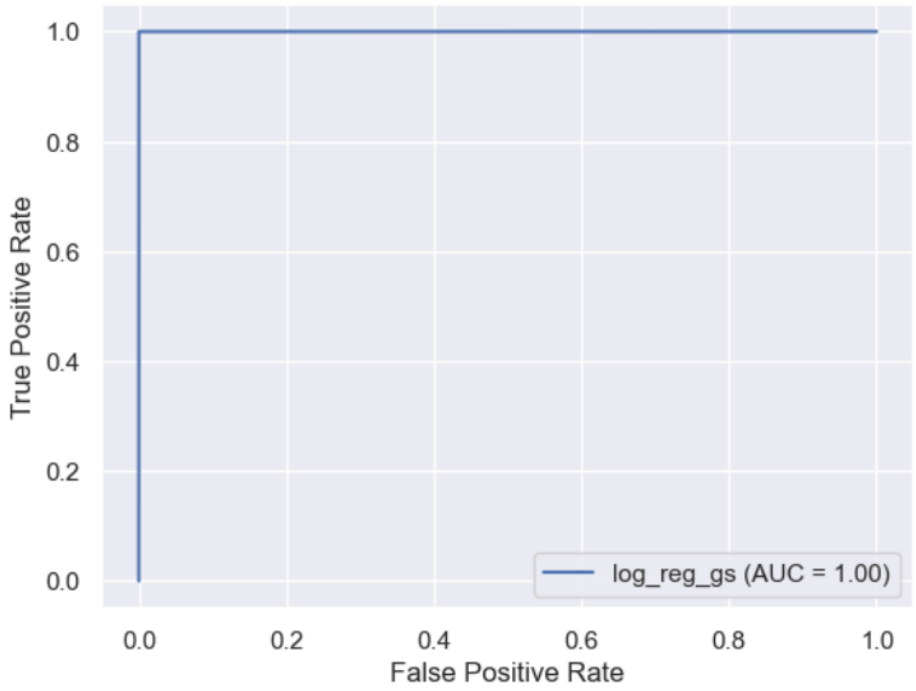
[FIG 4.5]

4.4.1.2 Classification Report

	precision	recall	f1-score	support
0	1.00	1.00	1.00	8396
1	1.00	1.00	1.00	8244
accuracy			1.00	16640
macro avg	1.00	1.00	1.00	16640
weighted avg	1.00	1.00	1.00	16640

[FIG 4.6]

4.4.1.3 ROC Curve

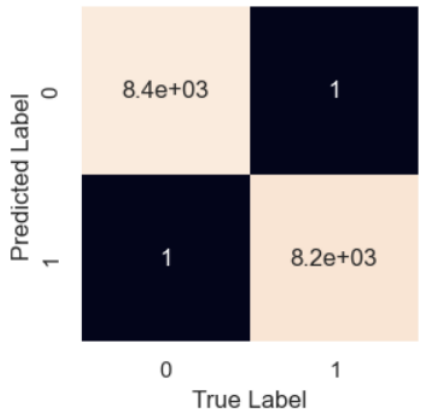


[FIG 4.7]

4.4.2 Random Forest Aggressor

Random Forest Accuracy: **0.9925**

4.4.2.1 Confusion Matrix



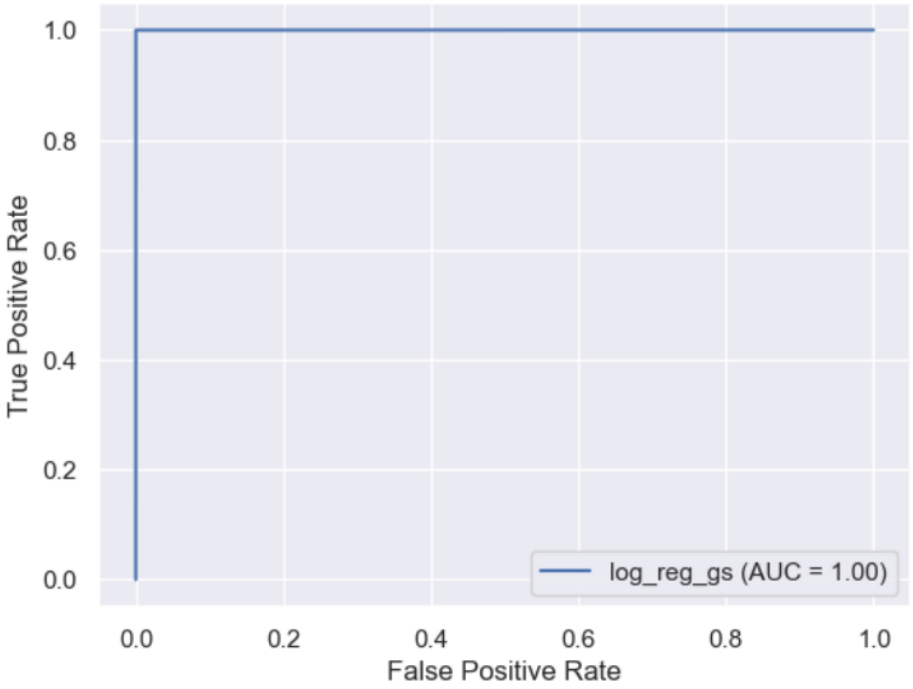
[FIG 4.8]

4.4.2.2 Classification Report

	precision	recall	f1-score	support
0	1.00	1.00	1.00	8396
1	1.00	1.00	1.00	8244
accuracy			1.00	16640
macro avg	1.00	1.00	1.00	16640
weighted avg	1.00	1.00	1.00	16640

[FIG 4.9]

4.4.2.3 ROC Curve



[FIG 4.10]

CHAPTER 5

COMPARISON

Our first discrepancy arose within using stemming and lemmatization and due to our limited word count and a good amount of processing power we utilized lemmatization because of its higher accuracy and because it uses a ML method to perform stem extraction unlike stemming which just removes suffixes.

As we noticed that in our testing the three different models we compared their accuracies and put them in a graph which evidently shows that KNN was indeed the worst performer out of all three.

The KNN model performed with a dismal accuracy of ~0.56, which is why we chose to drop it while performing our further tests on the model.

Next we performed rigorous system analysis on the remaining ml models which were Random Forest and Logistic Regression using tools such as Confusion Matrix, Classification Report and ROC Curve.

These operations were performed on two separate operating systems, Windows 10 and MacOS 14 which ran on intel i9-11400H and RTX 3060 and the mac uses an ARM SOC chipset which is the M1 chip. Both of these systems provided a similar output and was used for it’s high powered neural engine capabilities.

Chapter 6

CONCLUSION

The initial selection of lemmatization over stemming was driven by the need to balance accuracy with computational efficiency in the context of a limited word count and ample processing power. Lemmatization's superior accuracy and its utilization of machine learning for stem extraction, compared to stemming's rudimentary suffix removal technique, were crucial factors in our decision.

A comparative evaluation of the three models revealed KNN's consistently subpar performance, characterized by an accuracy of approximately 0.56. This underwhelming performance necessitated the exclusion of KNN from further model testing.

A comprehensive system analysis of the remaining machine learning models, Random Forest and Logistic Regression, was conducted using rigorous evaluation tools such as confusion matrices, classification reports, and ROC curves. These analyses were performed on two distinct operating systems, Windows 10 and macOS 14, both equipped with high-performance hardware configurations. The Windows system employed an Intel i9-11400H processor and an RTX 3060 graphics card, while the macOS system utilized the M1 chip, a powerful ARM SOC chipset. Both systems yielded comparable performance, with the macOS system's neural engine capabilities demonstrating particular effectiveness.

REFERENCES

1. Allcott, H., & Gentzkow, M. (2017). Social media and fake news in the 2016 election. Journal of Economic Perspectives, 31(2), 211-236.
2. Lazer, D. M., Baum, M. A., Benkler, Y., Berinsky, A. J., Druckman, J. N., Fowler, A., ... & Singer, J. (2018). The science of fake news. Science, 359(6379), 1094-1096.
3. Pennycook, G., & Rand, D. G. (2019). The psychology of fake news. MIT Press.
4. Vosoughi, S., Roy, D., & Dunbar, S. (2018). The spread of true and false news online. Science, 359(6379), 1146-1151.
5. www.researchgate.net/publication/333121583_Communicating_to_the_Public_in_the_Era_of_Conspiracy_Theory
6. medium.com/programminghero/10-python-libraries-that-every-python-developer-should-learn-to-boost-their-career-649881f3a013
7. www.xspdf.com/resolution/50628168.html
8. en.wikipedia.org/wiki/Lemmatization#:~:text=Lemmatization%20(or%20less%20commonly%20lemmatisation,or%20even%20an%20entire%20document.
9. https://www.researchgate.net/figure/Schematic-diagram-of-the-random-forest-algorithm_fig3_355828449
10. https://www.researchgate.net/figure/Schematic-diagram-for-logistic-regression-classification_fig2_333982722
11. https://www.researchgate.net/figure/A-simple-flowchart-for-the-k-nearest-neighbor-modeling_fig1_346429285

Plag Report

ORIGINALITY REPORT

0%	0%	0%	0%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

Exclude quotes	On	Exclude matches	< 10 words
Exclude bibliography	On		