**FAKE NEWS PREDICTION**

**PROJECT REPORT**

***Submitted by***

**ASHANK YADAV (2110993768)**

**HARSHIT BEHL (2110993788)**

**CHIRAG GOYAL** **(2110993871)**

Logo

Description automatically generated

**BE-CSE (Artificial Intelligence)**

***Guided by***

**Dr. Kamal Deep Garg**

**CHITKARA UNIVERSITY INSITUTE OF ENGINEERING & TECHNOLOGY**

**CHITKARA UNIVERSITY, RAJPURA**

**MAY 2023**

# ACKNOWLEDGEMENT

With immense pleasure **We**, **Mr. Ashank Yadav, Mr. Harshit Behl and Mr. Chirag Goyal** presenting “**Fake News Prediction System**” project report as part of the curriculum of ‘BE-CSE (AI).’

I would like to express my sincere thanks to **Mr. Rahul Dogra**, for his/her valuable guidance and support in completing my project.

I would also like to express my gratitude towards our dean **Dr Sushil Kumar Narang** for giving me this great opportunity to do a project on **Fake News Prediction**. Without their support and suggestions, this project would not have been completed.

Signature……….

Name: Ashank Yadav

Roll No. 2110993768

Signature……….

Name: Harshit Behl

Roll No. 2110993788

Signature……….

Name: Chirag Goyal

Roll No. 2110993871

# **ABSTRACT**

In recent years, due to the booming development of online social networks, fake news for various commercial and political purposes has been appearing in large numbers and widespread in the online world. With deceptive words, online social network users can get infected by this online fake news easily, which has brought about tremendous effects on the offline society already. An important goal in improving the trustworthiness of information in online social networks is to identify the fake news timely. This paper aims at investigating the principles, methodologies and algorithms for detecting fake news articles, creators and subjects from online social networks and evaluating the corresponding performance. Information preciseness on Internet, especially on social media, is an increasingly important concern, but web-scale data hampers, ability to identify, evaluate and correct such data, or so called "fake news," present in these platforms. In this paper, we propose a method for "fake news" detection and ways.

This method uses Random Forest model to predict whether a post on internet will be labelled as real or fake. The results may be improved by applying several techniques that are discussed in the paper. Received results suggest, that fake news detection problem can be addressed with machine learning methods. With the recent social media boom, the spread of fake news has become a great concern for everybody. It has been used to manipulate public opinions, influence the election - most notably the US Presidential Election of 2016, incite hatred and riots like the genocide of the Rohingya population. A 2018 MIT study found that fake news spreads six times faster on Twitter than real news. The credibility and trust in the news media are at an all-time low. It is becoming increasingly difficult to determine which news is real and which is fake. Various machine learning methods have been used to separate real news from fake ones. In this study, we tried to accomplish that using tf-idf and natural language processing. There are lots of machine learning models but this one has shown better progress. Now there is some confusion present in the authenticity of the correctness. But it definitely opens the window for further research. There are some of the aspects that has to be kept in mind considering the fact that fake news detection is not only a simple web interface but also a quite complex thing that includes a lot of backend work.

LIST OF FIGURES

Figure 1 The circulation of fake news over the years in India with an average increase in the production of fake stories from 2018. 8

Figure 2 Machine Learning Methodology 10

Figure 3 Reading CSV file. 12

Figure 4 Feature Engineering 13

Figure 5 Stemmed feature. 14

Figure 6 Dataset is balanced. 14

Figure 7 Logistic Regression: Cost Function 15

Figure 8 Logistic Regression 16

Figure 9 Random Forest 16

Figure 10 Support Vector Machine 17

Figure 11 Naive Bayes Algorithm 18

Figure 12 Logistic Regression Report 19

Figure 13 Support Vector Machine Report 19

Figure 14 Random Forest Report 20

Figure 15 Naive Bayes Report 20

Figure 16 Logistic Regression Confusion Matrix 21

Figure 17 Support Vector Machine Confusion Matrix 22

Figure 18 Random Forest Confusion Matrix 22

Figure 19 Naive Bayes Confusion Matrix 23

Figure 20 Model deployment: Homepage 25

Figure 21 Model deployment: Check page 26

Figure 22 Model deployment: Check page working. 26

# **LIST OF TABLES**

Table Models’ Accuracy……………………………………………………………………23

Chart Models’ Accuracies & F1 Scores…………………………………………………….23

Flowchart…………………………………………………………………………………….24

Table OF Contents

[ACKNOWLEDGEMENT I](#_Toc135872555)

[ABSTRACT II](#_Toc135872556)

[LIST OF FIGURES III](#_Toc135872557)

[LIST OF TABLES III](#_Toc135872558)

[1.INTRODUCTION 6](#_Toc135872559)

[2.PROBLEM STATEMENT 7](#_Toc135872560)

[3.PROBLEM FORMULATION 7](#_Toc135872561)

[4.OBJECTIVE 8](#_Toc135872562)

[5.OVERVIEW OF PROJECT 8](#_Toc135872563)

[6.EXPERIMENT SETUP 8](#_Toc135872564)

[7.METHODOLOGY 9](#_Toc135872565)

[8.DATASET AND FEATURES 9](#_Toc135872566)

[9.MODULE DESCRIPTION 10](#_Toc135872567)

[10. LOADING DATA 11](#_Toc135872568)

[11. DATA PRE-PROCESSING AND FEATURE ENGINEERING 12](#_Toc135872569)

[12. DATA VISUALIZATION 13](#_Toc135872570)

[13. ALGORITHMS USED 14](#_Toc135872571)

[14. MODEL PREPARATION 18](#_Toc135872572)

[15. EVALUATION 20](#_Toc135872573)

[16. RESULT 23](#_Toc135872574)

[17. FLOWCHART 23](#_Toc135872575)

[18. MODEL DEPLOYMENT 25](#_Toc135872576)

[19. CONCLUSION 27](#_Toc135872577)

[20. FUTURE WORK 27](#_Toc135872578)

[21. REFERENCES 27](#_Toc135872579)

1. INTRODUCTION

These days fake news is creating different issues from sarcastic articles to a fabricated news and plan government propaganda in some outlets. Fake news and lack of trust in the media are growing problems with huge ramifications in our society. Obviously, a purposely misleading story is fake news, but lately blathering social media’s discourse is changing its definition. Some of them now use the term to dismiss the facts counter to their preferred viewpoints. The importance of disinformation within American political discourse was the subject of weighty attention, particularly following the American president election. The term 'fake news' became common parlance for the issue, particularly to describe factually incorrect and misleading articles published mostly for the purpose of making money through page views. In this paper, it is sought to produce a model that can accurately predict the likelihood that a given article is fake news. Facebook and Twitter have been at the epicentre of much critique following media attention. They have already implemented a feature to flag fake news on the site when a user sees it; they have also said publicly they are working on to distinguish these articles in an automated way. Certainly, it is not an easy task. A given algorithm must be politically unbiased – since fake news exists on both ends of the spectrum – and also give equal balance to legitimate news sources on either end of the spectrum. In addition, the question of legitimacy is a difficult one. However, to solve this problem, it is necessary to have an understanding on what Fake News is.

India witnessed a 214% rise in cases relating to fake news in the pandemic year of 2019. There were numerous events across the country that were affected. For instance, during the 2016 Indian banknote demonetization, multiple fake news reports about spying technology added to the banknotes went viral. Around 5,000 social media handles were suspended by Indian security and intelligence agencies during the CAA protests the dissemination of fake content via WhatsApp was prevalent during India's 2019 general election. The proliferation of fake news in India is massive, and there is a dire need to consider solutions explicitly catering to this region.

To initiate the research for the Indian region, we introduced two resources: FakeNewslndia[3](https://cacm.acm.org/magazines/2022/11/265829-fake-news-in-india/fulltext#R3) in 2021 and Fact Drill, in 2022. Fake News India is a subset of Fact Drill—a data repository of 22,435 fact-checked social media stories to study fake news incidents in India. Its data spans over seven years, from 2013 to 2020. The samples are from the 11 Indian fact-checking websites certified by IFCN comprising 13 different languages. Fourteen varied attributes associated with each sample are categorized under meta-features, textual features, media features, social features, and event features.

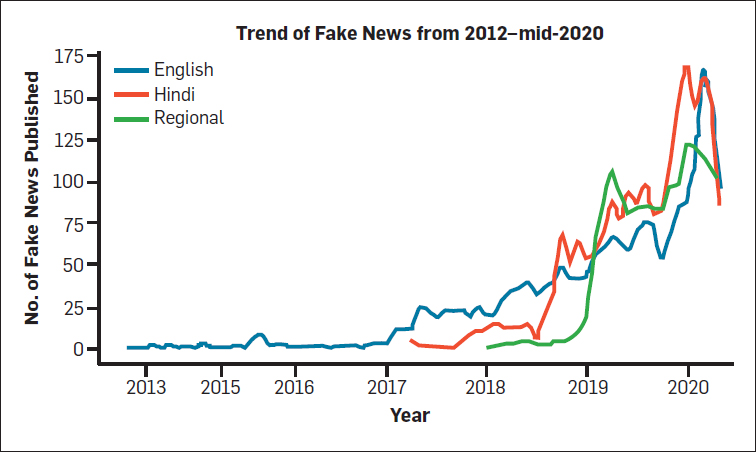


Figure 1 The circulation of fake news over the years in India with an average increase in the production of fake stories from 2018.

From Fact Drill, we noticed an average increase in the production of fake stories with regional languages entering the space in 2018. We also concluded from the domain/tags attribute present in the Fact Drill that political activity is the primary ground for exacerbating the proliferation of fake news in India.

1. PROBLEM STATEMENT

Nowadays, it is extremely difficult to decide whether the news we come across is real or not. There are very few options to check the authenticity and all of them are sophisticated and not accessible to the average person. There is an acute need for a web-based fact-checking platform that harnesses the power of Machine Learning to provide us with that opportunity.

1. PROBLEM FORMULATION

We will be training and testing the data, when we use supervised learning, it means we are labelling the data. By getting the testing and training data and labels we can perform different machine learning algorithms but before performing the predictions and accuracies, the data is need to be pre-processing i.e., the null values which are not readable are required to be removed or filled from the data set and the data is required to be converted into vectors by normalizing and tokening the data so that it could be understood by the machine. Next step is by using this data, getting the visual reports, which we will get by using the Matplotlib Library of Python and Scikit Learn. This library helps us in getting the results in the form of histograms, pie charts or bar charts.

1. OBJECTIVE

The objective of this project is to examine the problems and possible significances related with the spread of fake news. We will be working on different fake news data set in which we will apply different machine learning algorithms to train the data and test it to find which news is the real news or which one is the fake news. As the fake news is a problem that is heavily affecting society and our perception of not only the media but also facts and opinions themselves. By using the artificial intelligence and the machine learning, the problem can be solved as we will be able to mine the patterns from the data to maximize well defined objectives. So, our focus is to find which machine learning algorithm is best suitable for what kind of text dataset. Also, which dataset is better for finding the accuracies as the accuracies directly depends on the type of data and the amount of data. The more the data, more are your chances of getting correct accuracy as you can test and train more data to find out your results.

1. OVERVIEW OF PROJECT

With the advancement of technology, digital news is more widely exposed to users globally and contributes to the increment of spreading and disinformation online. Fake news can be found through popular platforms such as social media and the Internet. There have been multiple solutions and efforts in the detection of fake news where it even works with tools. However, fake news intends to convince the reader to believe false information which deems these articles difficult to perceive. The rate of producing digital news is large and quick, running daily at every second, thus it is challenging for machine learning to effectively detect fake news.

1. EXPERIMENT SETUP

There exists a large body of research on the topic of machine learning methods for deception detection, most of it has been focusing on classifying online reviews and publicly available social media posts. Particularly since late 2016 during the American Presidential election, the question of determining 'fake news' has also been the subject of particular attention within the literature.

Project tries to outline several approaches that seem promising towards the aim of perfectly classify the misleading articles. They note that simple content-related n-grams and shallow parts-of-speech tagging have proven insufficient for the classification task, often failing to account for important context information. Rather, these methods have been shown useful only in tandem with more complex methods of analysis. Deep Syntax analysis using Probabilistic Context Free Grammars have been shown to be particularly valuable in combination with n-gram methods. Model can achieve 85%-91% accuracy in deception related classification tasks using online review corpora.

1. METHODOLOGY

In this paper a model is build based on the count vectorizer or a tf-idf matrix (i.e.) word tallies relatives to how often they are used in other articles in your dataset) can help. Since this problem is a kind of text classification, implementing a Naive Bayes classifier will be best as this is standard for text-based processing. The actual goal is in developing a model which was the text transformation (count vectorizer vs tf-idf vectorizer) and choosing which type of text to use (title vs full text). Now the next step is to extract the most optimal features for count vectorizer or tf-idf-vectorizer, this is done by using a n-number of the most used words, and/or phrases, lower casing or not, mainly removing the stop words which are common words such as “the”, “when”, and “there” and only using those words that appear at least a given number of times in a given text dataset.

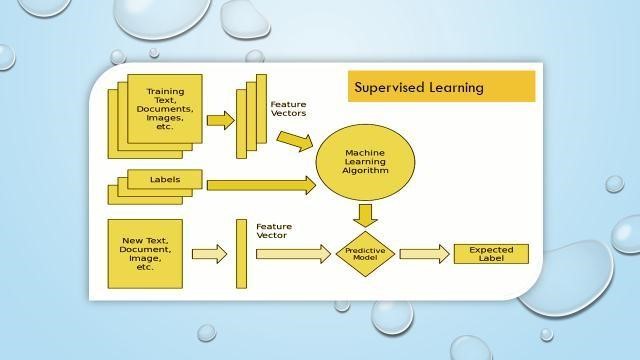


Figure 2 Machine Learning Methodology

1. DATASET AND FEATURES

The dataset for fake news detection, collected from Kaggle, consists of 20,000 rows and following features,

* **Title**: This column contains the titles or headlines of the news articles. The title often provides a summary or description of the news content.
* **Author**: This column contains the names or identities of the authors who wrote the news articles. It provides information about the individuals responsible for producing the news content.
* **Text**: This column contains the main body of the news articles. It includes the detailed information, facts, and statements presented in the news.
* **Label**: This column is used for labelling the news articles as real or fake. It assigns a binary value (0 or 1) to indicate whether the news is considered genuine or fraudulent. A label of 0 typically represents real news, while a label of 1 represents fake news.

The dataset aims to provide a comprehensive collection of news articles along with relevant information that can be used to train and evaluate machine learning models for fake news detection. By analysing the title, author, and content of the news articles, the goal is to develop a predictive model that can accurately classify news articles as real or fake.

This dataset can be utilized for various purposes, such as, Training and evaluating machine learning algorithms: The dataset can be used to train and evaluate models, such as Naive Bayes, SVM, or Random Forest, to predict the authenticity of news articles based on their features. Feature engineering and text analysis: The dataset allows for exploring different features and performing text analysis techniques, such as text pre-processing, TF-IDF vectorization, or word embeddings, to extract meaningful information and patterns from the news content. Comparative analysis: Researchers or practitioners can compare the performance of different models or techniques for fake news detection by using this dataset as a benchmark. Insights and research: The dataset can be used to gain insights into the characteristics of fake news articles, investigate trends, patterns, or factors that contribute to the spread of misinformation, and conduct research in the field of fake news detection and media literacy.

Overall, this dataset provides a valuable resource for studying, analysing, and developing methods to tackle the challenge of identifying fake news and promoting information credibility in the digital age.

1. MODULE DESCRIPTION

* **NumPy**: NumPy is a fundamental library for scientific computing in Python. It provides support for large, multi-dimensional arrays and matrices, along with a collection of mathematical functions to operate on these arrays efficiently. NumPy is widely used in data analysis, numerical computations, and machine learning tasks.
* **Pandas**: Pandas is a powerful library for data manipulation and analysis in Python. It provides data structures, such as Data Frames, that allow easy handling of structured data. Pandas offers functionalities for reading and writing data in various formats, cleaning and pre-processing data, performing data aggregation and summarization, and conducting exploratory data analysis.
* **NLTK (Natural Language Toolkit)**: NLTK is a library for natural language processing (NLP) in Python. It provides tools and resources for various NLP tasks, including tokenization, stemming, lemmatization, part-of-speech tagging, sentiment analysis, and more. NLTK also offers corpora, lexical resources, and pre-trained models for text analysis and NLP research. The Porter Stemmer is an algorithm for stemming words in the English language. Stemming is the process of reducing words to their base or root form. The Porter Stemmer in NLTK is widely used for stemming words by removing suffixes.
* **Scikit-learn**: scikit-learn is a popular machine learning library in Python. It provides a wide range of machine learning algorithms and tools for tasks such as classification, regression, clustering, dimensionality reduction, and model evaluation. Scikit-learn offers a unified interface and consistent API for implementing and working with various machine learning models. The sklearn.metrics module in scikit-learn provides various evaluation metrics for assessing the performance of machine learning models. It includes metrics such as accuracy, precision, recall, F1-score, ROC curve, confusion matrix, and more. The sklearn.metrics module is commonly used to evaluate and compare the performance of classification and regression models.
* **re (Regular Expressions)**: The re module in Python provides support for regular expressions. Regular expressions are patterns used to match and manipulate strings. The re module allows you to search, match, and manipulate text based on specified patterns.
* **Word cloud**: The word cloud module is a library for creating word clouds in Python. Word clouds visually represent the frequency or importance of words in a text. The word cloud module provides functionalities to generate attractive and customizable word clouds from textual data.
* **Matplotlib**: Matplotlib is a popular plotting library in Python. It provides a comprehensive set of functions for creating various types of static, animated, and interactive visualizations. Matplotlib can be used to generate line plots, scatter plots, bar plots, histograms, pie charts, and more.
* **Seaborn**: Seaborn is a data visualization library built on top of matplotlib. It provides a high-level interface for creating attractive and informative statistical graphics. Seaborn simplifies the process of creating complex visualizations, such as heatmaps, cluster maps, violin plots, box plots, and more.

These modules play a crucial role in data analysis, machine learning, natural language processing, and visualization tasks, providing efficient and convenient tools for working with data, text, and models in Python.

1. LOADING DATA

So, in this project we are using different packages and to load and read the data set we are using pandas. By using pandas, we can read the .csv file and then we can display the shape of the dataset with that we can also display the dataset in the correct form. We will be training and testing the data, when we use supervised learning, it means we are labelling the data. By getting the testing and training data and labels we can perform different machine learning algorithms but before performing the predictions and accuracies, the data is need to be pre-processing i.e., the null values which are not readable are required to be removed from the data set and the data is required to be converted into vectors by normalizing and tokening the data so that it could be understood by the machine. Next step is by using this data, getting the visual reports, which we will get by using the Matplot library of Python and Scikit Learn. This library helps us in getting the results in the form of histograms, pie charts or bar charts.

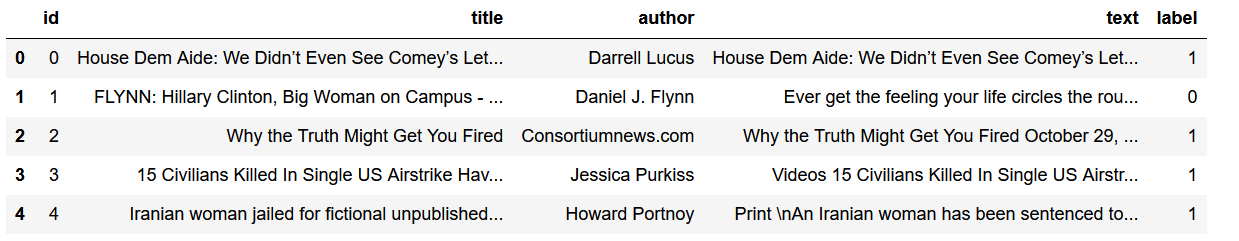


Figure 3 Reading CSV file.

1. DATA PRE-PROCESSING AND FEATURE ENGINEERING

* First, the presence of null values was checked in the data. Only a few of the rows had null values which were then replaced with empty string.
* Feature extraction is the process of selecting a subset of relevant features for use in model construction. Feature extraction methods helps in to create an accurate predictive model. They help in selecting features that will give better accuracy. When the input data to an algorithm is too large to be handled and it is supposed to be redundant then the input data will be transformed into a reduced illustration set of features also named feature vectors. Altering the input data to perform the desired task using this reduced representation instead of the full-size input. Feature extraction is performed on raw data prior to applying any machine learning algorithm, on the transformed data in feature space. Here we have chosen features ‘Author’ and ‘Title’ to work with.
* We have combined the 'Author' and 'Title' columns together to create a single text field that contains both the author's name and the news title. By concatenating these columns, we are feature engineering and creating a consolidated textual representation that captures both the source of the news and a concise summary of its content.
* After concatenating the 'Author' and 'Title' columns, we have applied regular expressions (regex) to remove special characters from the text. Special characters such as punctuation marks, symbols, and other non-alphabetic characters are eliminated from the text. Removing special characters helps in cleaning the text data and preparing it for further analysis.
* Once the special characters have been removed, we have applied the Porter Stemmer algorithm to the pre-processed text. The Porter Stemmer is a widely used stemming algorithm in natural language processing (NLP). Stemming is the process of reducing words to their base or root form. By stemming the text, we are normalizing the words to their core meaning, regardless of their inflections or tense variations. This step helps to reduce the dimensionality of the text data and improve the efficiency of subsequent analysis.
* After stemming the text, we have utilized the TF-IDF technique on the pre-processed data. TF-IDF is a numerical statistic that reflects the importance of a word in a document or corpus. It calculates a weight for each word by considering both its frequency in the document and its rarity across the entire corpus. TF-IDF is commonly used in text mining and information retrieval to highlight important terms and downplay common or irrelevant words. By applying TF-IDF, we are transforming the stemmed text into a numerical representation that captures the relative importance of words in the context of the dataset.

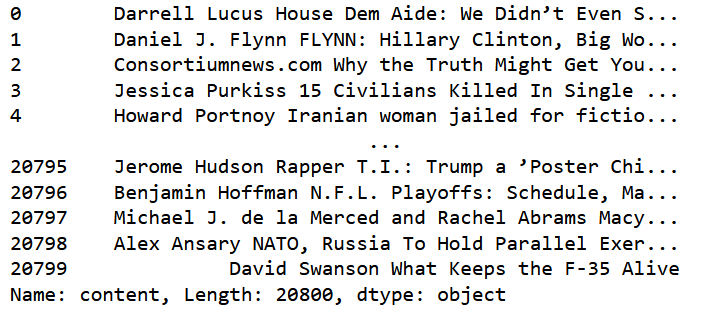


Figure 4 Feature Engineering

Overall, our pre-processing pipeline involves concatenating the 'Author' and 'Title' columns, removing special characters using regex, stemming the text using the Porter Stemmer, and finally applying the TF-IDF technique. These steps collectively help in preparing the textual data for further analysis, such as classification or clustering, by reducing noise, normalizing words, and capturing the semantic significance of the text features.

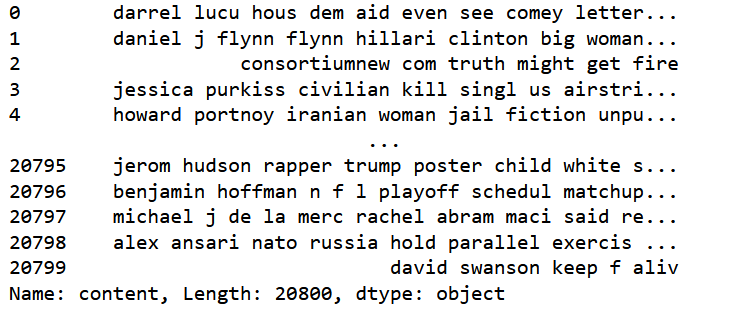


Figure 5 Stemmed feature.

1. DATA VISUALIZATION

We next visualised the data to acquire deeper insights and comprehend the patterns and trends evident after doing data manipulation and exploring the dataset. Data visualisation is a potent approach that enables us to graphically display the data, making it simpler to understand and spot patterns that may not be immediately obvious in raw data.

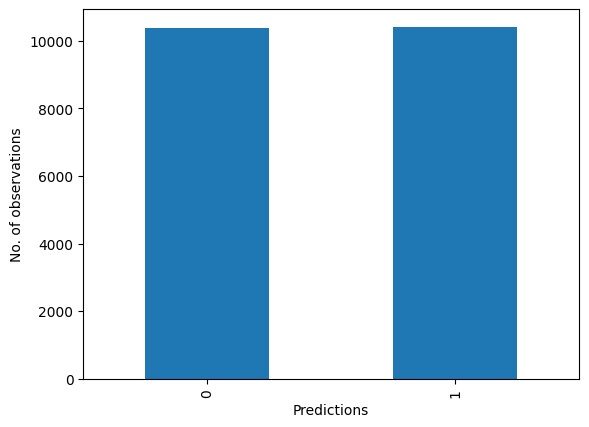


Figure 6 Dataset is balanced.

1. ALGORITHMS USED

As In this project We are using Scikit-Learn Machine learning library for implementing the architecture. Scikit Learn is an open-source python Machine Learning library. This just needs importing the packages and you can compile the command as soon as you write it. If the command does not run, we can get the error at the same time. We are using 4 different algorithms and we have trained these 4 models i.e., Naïve Bayes, Support Vector Machine, Random Forest, and Logistic Regression which are very popular methods for document classification problem. Once the classifiers are trained, we can check the performance of the models on test-set. We can extract the word count vector for each mail in test-set and predict it class with the trained models.

* **Logistic Regression**:

Logistic regression is another supervised learning algorithm that is appropriate to conduct when the dependent variable binary. It is commonly used to obtain odds ratio in the presence of more than one explanatory variable. The procedure is quite like linear regression, but its response variable is binomial. The result is the impact of each variable on the odds ratio of the observed event of interest. Below is the general proof of the logistic regression: In this section, we will fit the logistic regression into the news data. Also, we will try to determine the optimal parameter for logistic regression. The method is like the method in the last section. In this case, we would use ridge regression here because it enforces the β coefficients to be lower, but it does not enforce them to be zero. That is, it will not get rid of irrelevant features but rather minimize their impact on the trained model. By which, the model would tend to have more prediction power. Below is the Cost Function for Logistic Regression with Ridge Penalty:

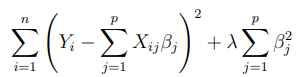


Figure 7 Logistic Regression: Cost Function

In ridge regression, Lambda (λ) controls the trade-off between bias and variance. In the other words, if λ is 0 or close to 0, the model will have enough power to increase its complexity by assigning big values to the weights for each parameter which will lead to overfitting problem. if we increase the value of λ, the model will tend to underfit, as the model will become too simple. In this case, we use parameter C as our regularization parameter. (Where C = 1/λ).

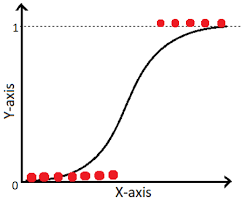


Figure 8 Logistic Regression

* **Random Forest:**

Random Forest is an ensemble learning method that combines multiple decision trees to make predictions. It leverages the power of averaging predictions from multiple trees to improve accuracy and reduce overfitting. Random Forest's superior accuracy makes it the most effective model in this context. Its ability to handle complex relationships and mitigate overfitting makes it a reliable choice for news prediction tasks. Random Forest is a supervised learning algorithm. It is like an ensemble of decision trees with bagging method. The general idea of the bagging method is that a combination of learning models improves the overall result. The Random Forest algorithm randomly selects observations and features to build several decision trees and then averages the results.

The key formulas used include:

Gini impurity: Used as a criterion to measure the impurity or diversity of a set of data points in classification tasks. It is calculated as: Gini(p) = 1 - Σ(p\_i)^2, where p\_i is the probability of class i in the set.

Mean squared error: Used as a criterion to measure the error or deviation from the actual values in regression tasks. It is calculated as: MSE = Σ(y\_i - ŷ\_i)^2 / n, where y\_i is the actual value, ŷi is the predicted value, and n is the number of data points.

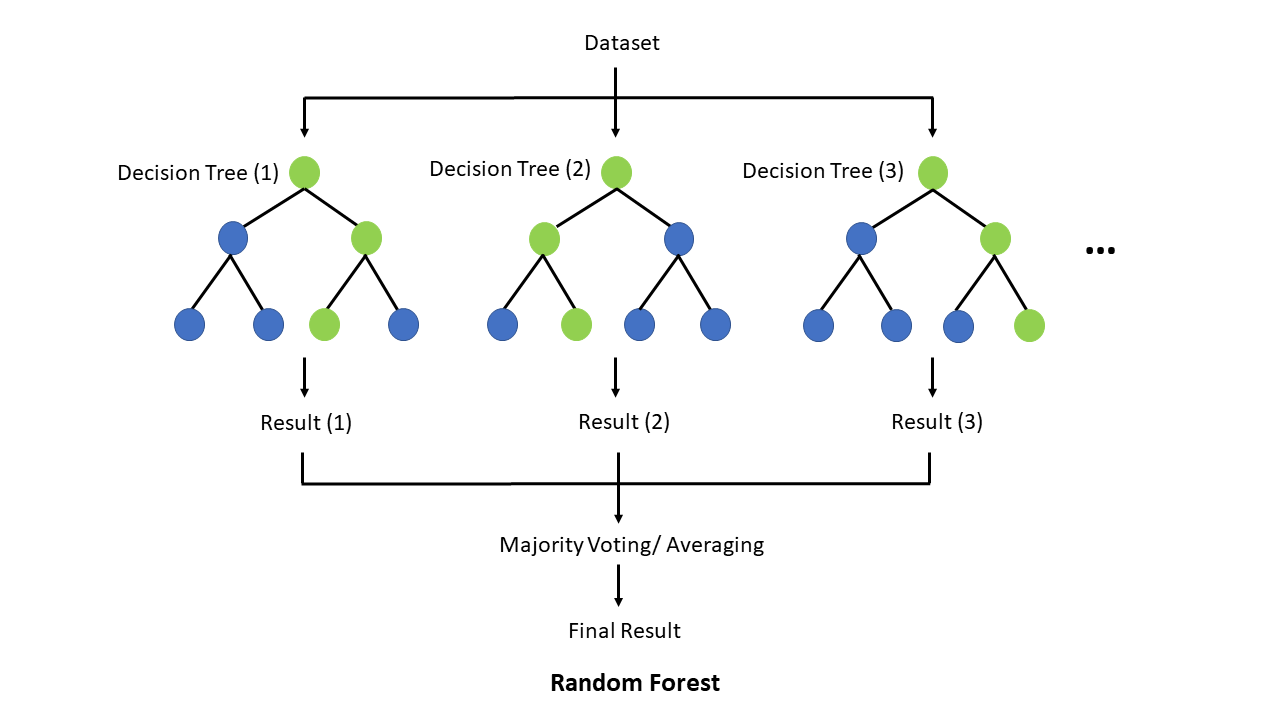


Figure 9 Random Forest

* **Support Vector Machines (SVM):**

A Support Vector Machine (SVM) is also a supervised learning algorithm which used to separating hyperplane. In other words, given labelled training data, the algorithm outputs an optimal hyperplane which classifies new examples. In two-dimensional space this hyperplane is a line dividing a plane in two parts where in each class lay in either side. In multidimensional space, the separation of the class is a hyperplane.

The formula for SVM can be expressed as follows:

Given a training dataset consisting of N data points {(x₁, y₁), (x₂, y₂), ..., (xₙ, yₙ)}, where xᵢ represents the input features and yᵢ represents the corresponding class labels (1 or -1 for binary classification), SVM aims to find the optimal hyperplane in the form of: w^T \* x + b = 0,

where w is a vector perpendicular to the hyperplane, and b is a bias term that determines the offset of the hyperplane from the origin.

To find the optimal hyperplane, SVM maximizes the margin, which is the perpendicular distance between the hyperplane and the nearest data points of different classes. The margin is denoted as 2/||w||, where ||w|| represents the Euclidean norm of the weight vector w.

The SVM optimization problem can be formulated as follows:

minimize: 1/2 \* ||w||² + C \* Σξᵢ,

subject to: yᵢ (w^T \* xᵢ + b) ≥ 1 - ξᵢ, for all i.

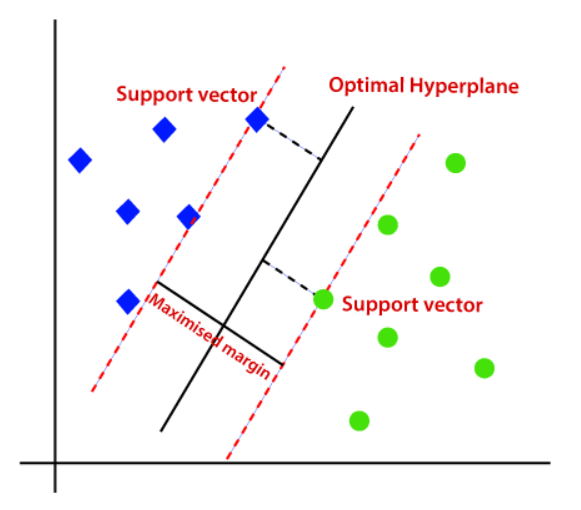


Figure 10 Support Vector Machine

* **Naïve Bayes:**

Bayes’ rule provides us with the formula for the probability of Y given some feature X. In real-world problems, we hardly find any case where there is only one feature.

When the features are independent, we can extend Bayes’ rule to what is called Naive Bayes which assumes that the features are independent that means changing the value of one feature does not influence the values of other variables and this is why we call this algorithm “*NAIVE*”. Naive Bayes can be used for various things like face recognition, weather prediction, Medical Diagnosis, News classification, Sentiment Analysis, and a lot more.

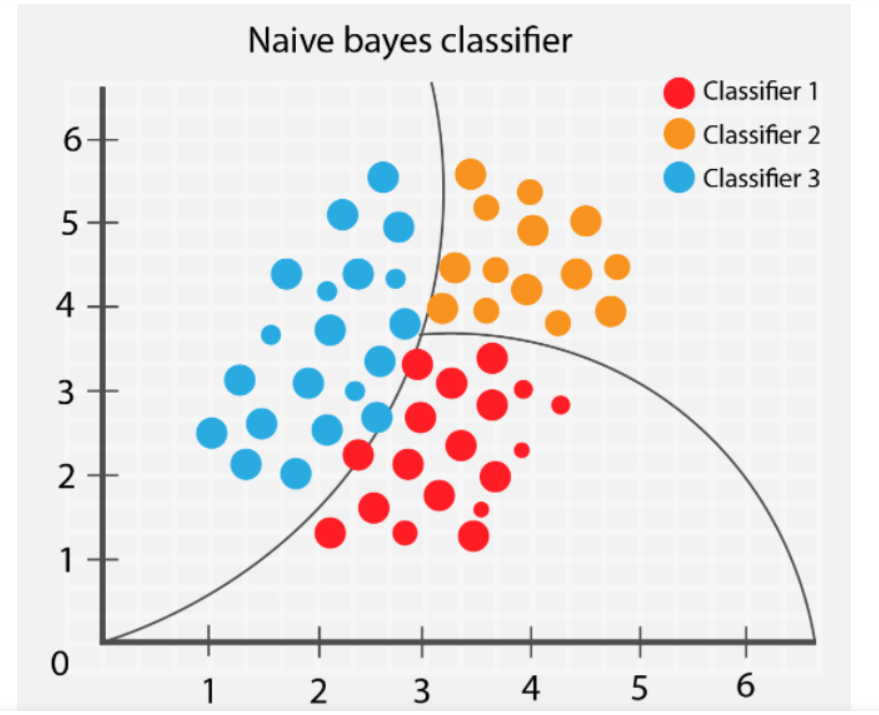


Figure 11 Naive Bayes Algorithm

1. MODEL PREPARATION

Numerous machines learning techniques, including Logistic Regression, Random Forest, and Support Vector Machine (SVM) and Naïve Bayes were chosen and put into practise. For analysing the models' effectiveness, the dataset was divided into training and testing sets.

* **Logistic Regression:** Logistic regression was employed on the news dataset to predict whether news is fake or real. The model achieved an accuracy of 97.3%, indicating its ability to correctly predict the news status for a significant portion of the dataset. However, accuracy alone is not a comprehensive evaluation measure, and further metrics such as precision, recall, confusion matrix and F1-score should be considered. The model's performance can be enhanced through parameter tuning and the use of more advanced algorithms.

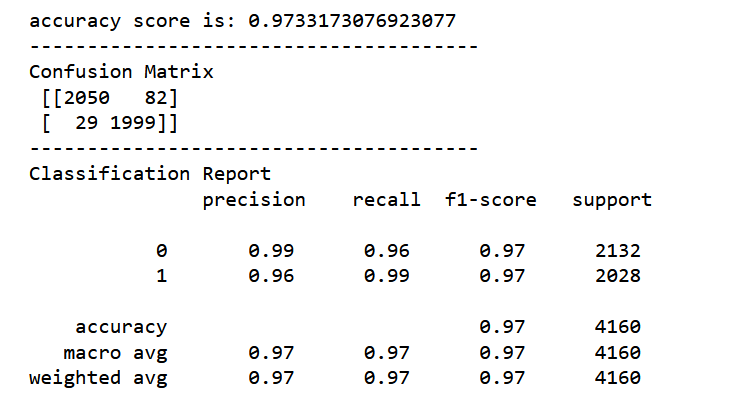


Figure 12 Logistic Regression Report

* **Support Vector Machine**: The Support Vector Machine (SVM) algorithm was implemented on the news dataset, achieving an accuracy of 98.8%. SVM is a powerful supervised learning algorithm used for classification tasks. It finds an optimal hyperplane that separates the data points belonging to different classes. SVM's ability to handle complex decision boundaries and capture non-linear relationships contributed to its strong performance. With its high accuracy, SVM proved to be a robust model for fake news prediction.

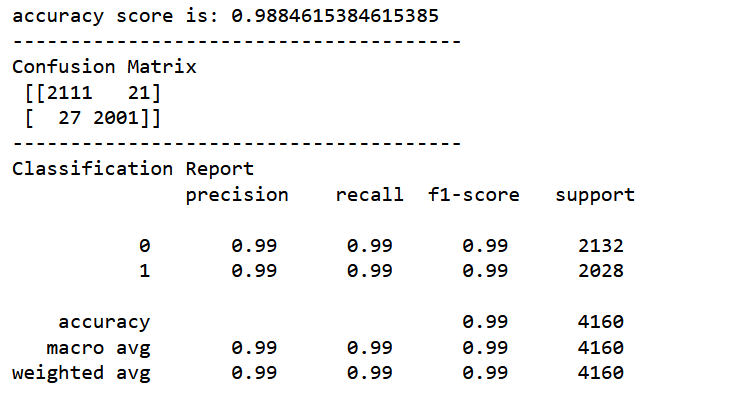


Figure 13 Support Vector Machine Report

* **Random Forest:** The Random Forest algorithm was implemented on the news dataset, achieving the highest accuracy of 99.2% among all models. Random Forest is an ensemble learning method that combines multiple decision trees to make predictions. It leverages the power of averaging predictions from multiple trees to improve accuracy and reduce overfitting. Random Forest's superior accuracy makes it the most effective model in this context. It outperformed Logistic Regression, and Support Vector Machine (SVM), and Naïve Bayes models. Its ability to handle complex relationships and mitigate overfitting makes it a reliable choice for news prediction tasks.

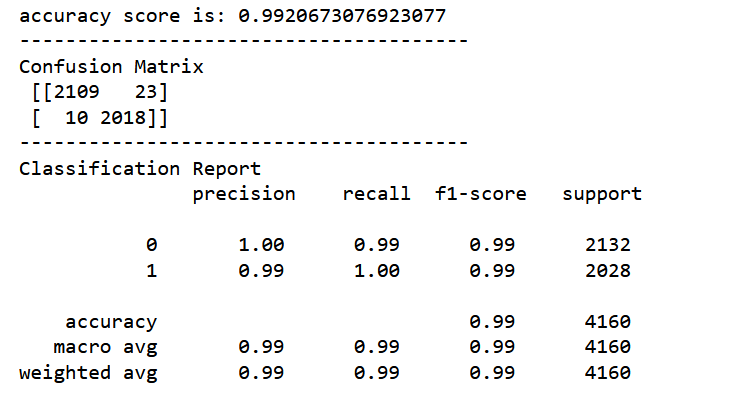


Figure 14 Random Forest Report

* **Naïve Bayes:** The Naïve Bayes algorithm was implemented on the news dataset, achieving an accuracy of 95.1%. The Naïve Bayes classifier is a supervised machine learning algorithm, which is used for classification tasks, like text classification. It is also part of a family of generative learning algorithms, meaning that it seeks to model the distribution of inputs of a given class or category. With its high accuracy, Naïve Bayes proved to be one of the great models for fake news prediction.

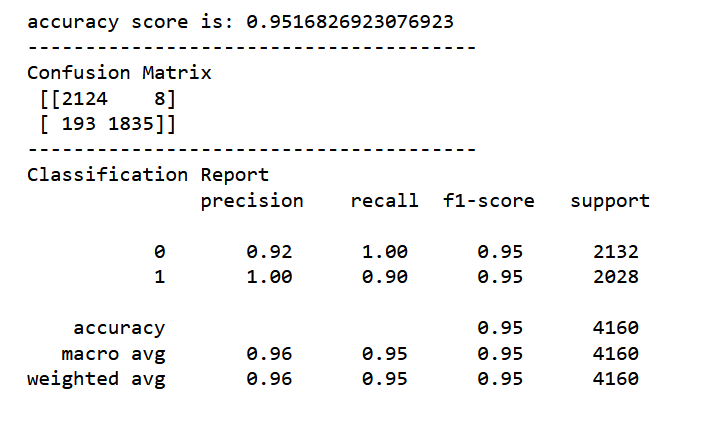


Figure 15 Naive Bayes Report

1. EVALUATION

To evaluate the performance of the models, evaluation metrics like F1 Score, Confusion matrix are used.

* **F1 Score**: F1 score is a machine learning evaluation metric that measures a model’s accuracy. It combines the precision and recall scores of a model. The accuracy metric computes how many times a model made a correct prediction across the entire dataset. This can be a reliable metric only if the dataset is class-balanced; that is, each class of the dataset has the same number of samples. Nevertheless, real-world datasets are heavily class-imbalanced, often making this metric unviable. For example, if a binary class dataset has 90 and 10 samples in class-1 and class-2, respectively, a model that only predicts “class-1,” regardless of the sample, will still be 90% accurate. Accuracy computes how many times a model made a correct prediction across the entire dataset. However, can this model be called a good predictor? This is where the F1 score comes into play. We will investigate the mathematical explanation behind the metric in the next section but let us first understand the precision and recall in relation to a binary class dataset with classes labelled “positive” and “negative.” Precision measures how many of the “positive” predictions made by the model were correct. Recall measures how many of the positive class samples present in the dataset were correctly identified by the model.
* **Confusion Matrix**: The confusion matrix is a matrix used to determine the performance of the classification models for a given set of test data. It can only be determined if the true values for test data are known. The matrix itself can be easily understood, but the related terminologies may be confusing. Since it shows the errors in the model performance in the form of a matrix, hence also known as an error matrix.

Here are all the model evaluations,

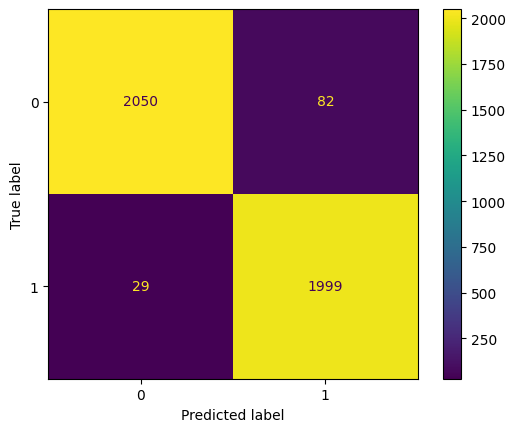


Figure 16 Logistic Regression Confusion Matrix

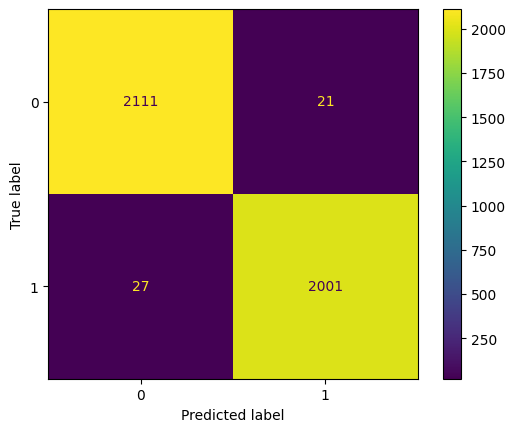


Figure 17 Support Vector Machine Confusion Matrix

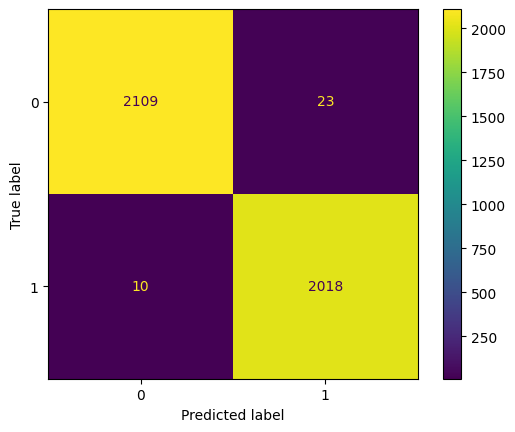
****

Figure 18 Random Forest Confusion Matrix

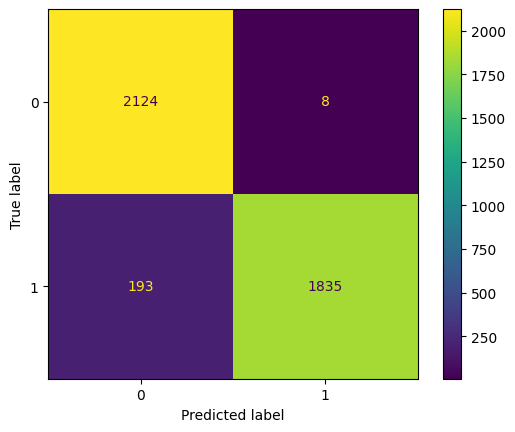
****

Figure 19 Naive Bayes Confusion Matrix

1. RESULT

|  |  |
| --- | --- |
| Models’ Accuracy | |
| Logistic Regression | 97.3% |
| Support Vector Machine | 98.8% |
| Random Forest | 99.2% |
| Naïve Bayes Algorithm | 95.1% |

*Chart: Models’ accuracies & F1 Scores*

Algorithm’s accuracy depends on the type and size of your dataset. More the data, more chances of getting correct accuracy. Machine learning depends on the variations and relations. Understanding what is predictable is as important as trying to predict it. While making algorithm choice, speed should be a consideration factor. Overall, Random Forest stood out as the top-performing algorithm, demonstrating its suitability for the given classification task. Therefore, for its accuracy, and high F1 score Random Forest Model is chosen.

1. FLOWCHART

The Data flowchart is also called as bubble chart. It is a simple graphical formalism that can be used to represent a system in terms of input data to the system, various processing carried out on this data, and the output data is generated by this system.

The data flow chart is one of the most important modelling tools. It is used to model the system components. These components are the system process, the data used by the process, an external entity that interacts with the system and the information flows in the system.

**TESTING DATA**

**TRAINING DATA**

**CATEGORICAL OUTPUT**

**SPLITTING THE DATA**

**VECTORIZE**

**STEMMING**

**FEATURE ENGINEERING**

**DATA PREPROCESSING**

**MISSING DATA**

LOADING DATASET

**FINAL CLASSIFICATION MODEL**

**MODEL SELECTION**

**PERFOMANCE METRICS**

**POST PROCESSING**

**TRAINING ALGORITHM**

1. MODEL DEPLOYMENT

In this project, we used Flask, a Python module created exclusively for building python web applications. Flask is a popular Python web framework that simplifies the process of building web applications. It provides a lightweight and flexible approach to developing web services and APIs. Flask is often referred to as a microframework because it focuses on simplicity and minimalism. It provides only the essential tools and features needed to build web applications, allowing developers to have more control and flexibility over their project structure and dependencies. Flask uses a routing mechanism that maps URLs to specific functions or view handlers. This allows you to define different routes for different parts of your application and handle various HTTP methods (GET, POST, etc.) accordingly. Routing in Flask is easy to configure and can handle dynamic parameters in URLs. Flask includes a powerful template engine called Jinja2, which allows you to separate the presentation layer from the application logic. With templates, you can dynamically generate HTML pages by embedding variables, loops, conditionals, and other control structures within your HTML code. Flask provides simple and intuitive ways to handle HTTP requests. You can access request data such as form input, query parameters, cookies, and more using the request object. Flask also supports file uploads, allowing you to process files sent by the client. Flask is a lightweight and flexible web framework that empowers developers to build web applications efficiently. It offers essential features for routing, request handling, response generation, template rendering, URL building, and more. Flask's simplicity, modularity, and extensibility make it a popular choice for developing web services and APIs.

The snapshots of the deployed model are displayed in the accompanying images:

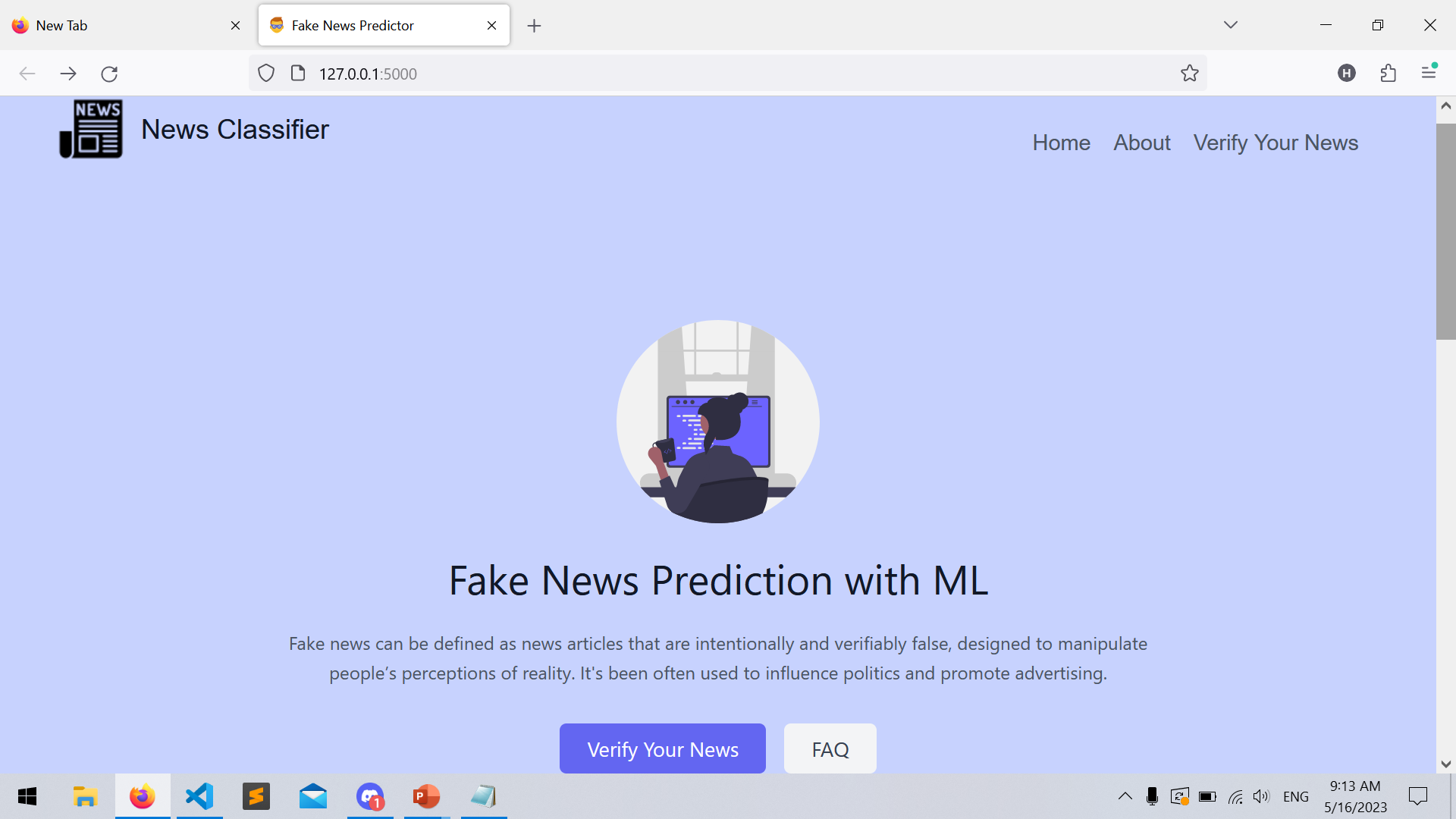


Figure 20 Model deployment: Homepage

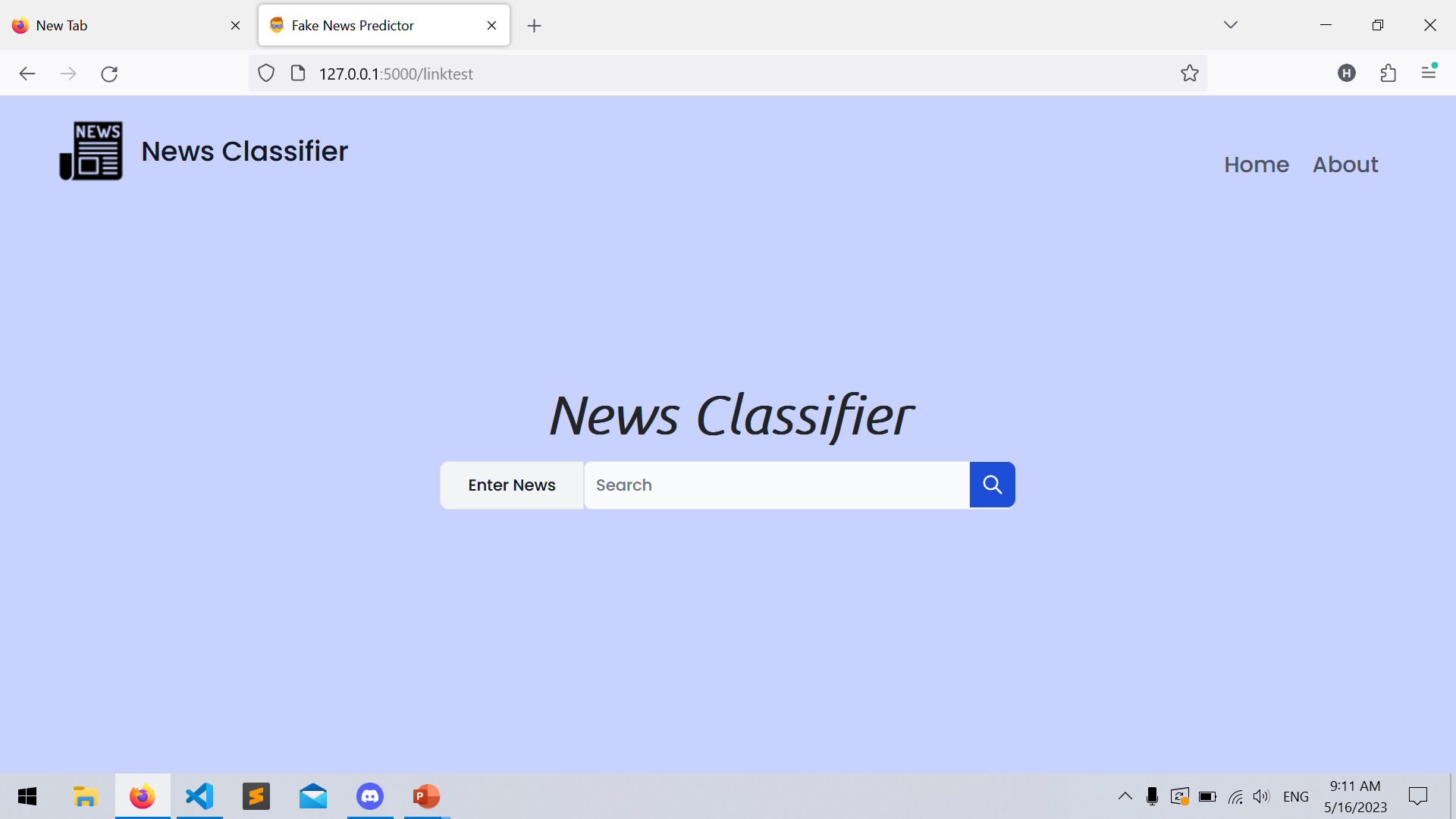
****

Figure 21 Model deployment: Check page

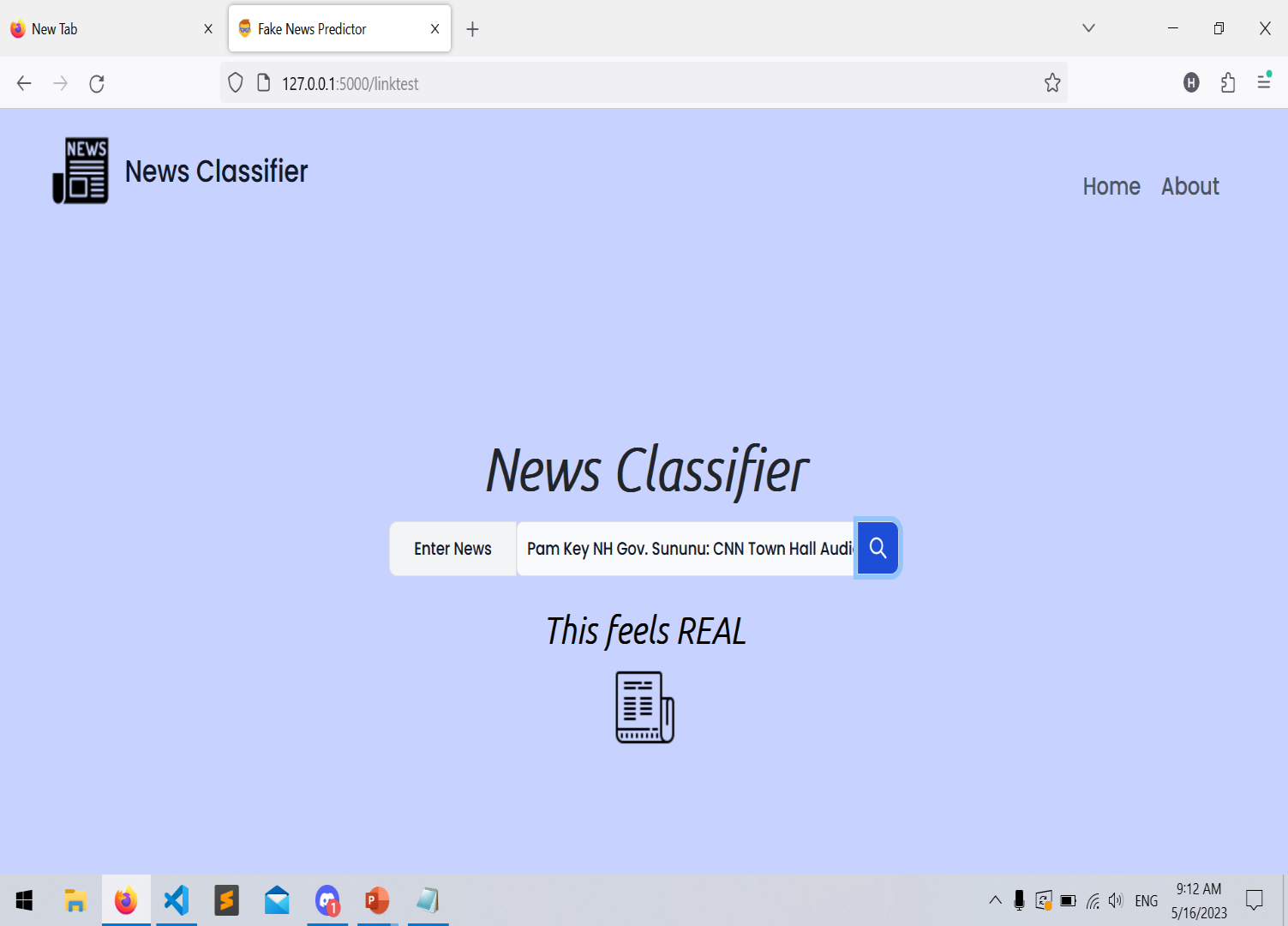
****

Figure 22 Model deployment: Check page working.

1. CONCLUSION

Many people consume news from social media instead of traditional news media. However, social media has also been used to spread fake news, which has negative impacts on individual people and society. In this paper, an innovative model for fake news detection using machine learning algorithms has been presented. This model takes news events as an input and based on twitter reviews and classification algorithms it predicts the percentage of news being fake or real.

1. FUTURE WORK

For analysis, some understanding of the major requirements for the system is essential. This study is carried out to check the impact of news on the world. The aim is to further increase the working efficiency of the model and extend its range to the farther news of the world. More complex algorithms could be considered for better understanding and future predictions

1. REFERENCES

[**https://www.analyticsvidhya.com/blog/2021/07/getting-started-with-nlp-using-nltk-library/**](https://www.analyticsvidhya.com/blog/2021/07/getting-started-with-nlp-using-nltk-library/)

[**https://towardsdatascience.com/how-tf-idf-works-3dbf35e568f0**](https://towardsdatascience.com/how-tf-idf-works-3dbf35e568f0)

[**https://www.w3schools.com/python/python\_ml\_confusion\_matrix.asp**](https://www.w3schools.com/python/python_ml_confusion_matrix.asp)

[**https://www.javatpoint.com/types-of-machine-learning**](https://www.javatpoint.com/types-of-machine-learning)