**FAKE NEWS DETECTION USING NLP**

**Table of contents**

* [Introduction](https://github.com/abiek12/Fake-News-Detection-using-MachineLearning#introduction)
* [Problem Definition](https://github.com/abiek12/Fake-News-Detection-using-MachineLearning#problem-definition)
* [Project Structure](https://github.com/abiek12/Fake-News-Detection-using-MachineLearning#project-structure)
* [Datasets](https://github.com/abiek12/Fake-News-Detection-using-MachineLearning#datasets)
* [Model Name](https://github.com/abiek12/Fake-News-Detection-using-MachineLearning#model-name)
* [Images](https://github.com/abiek12/Fake-News-Detection-using-MachineLearning#images)
* [Prerequisites](https://github.com/abiek12/Fake-News-Detection-using-MachineLearning#prerequisites)
* [Getting Started](https://github.com/abiek12/Fake-News-Detection-using-MachineLearning#getting-started)

**Introduction**

This repository contains a comprehensive project for detecting fake news using machine learning techniques and various natural language processing techniques. The project includes data analysis, model training, and a web application for real-time fake news detection. The machine learning model is designed to classify news articles as either real or fake based on their content.

**Problem Definition**

We aim to develop a machine learning program to identify when a news source may be producing fake news. The model will focus on identifying fake news sources, based on multiple articles originating from a source. Once a source is labeled as a producer of fake news, we can predict with high confidence that any future articles from that source will also be fake news. Focusing on sources widens our article misclassification tolerance, because we will have multiple data points coming from each source.

The intended application of the project is for use in applying visibility weights in social media. Using weights produced by this model, social networks can make stories that are highly likely to be fake news less visible.

**Project Structure**

The repository is organized into the following directories and files:

* **Images**: Contains important project images, such as block diagrams, classification reports, confusion matrices, and screenshots.
* **dataset**: Includes the dataset, consisting of train and test data from Kaggle, which is used to train and test the model.
* **static**: Houses static assets for the web application, including CSS, JavaScript, etc.
* **templates**: Includes HTML templates for the web application, such as Landingpage.html and prediction page.html.
* **Fake\_News\_Detector-PA.ipynb**: Jupyter Notebook file for data analysis and model training.
* **app.py**: Flask web application for real-time fake news detection.
* **model.pkl**: Pre-trained machine learning model for fake news detection.
* **vector.pkl**: Pre-trained vectorizer for text data.

**Datasets**

**Train.csv**

A full training dataset with the following attributes:

* id: unique id for a news article
* title: the title of a news article
* author: author of the news article
* text: the text of the article; could be incomplete
* label: a label that marks the article as potentially unreliable
  + 1: unreliable
  + 0: reliable

**test.csv**

A testing training dataset with all the same attributes as train.csv without the label.

**Model Name**

The machine learning model used for fake news detection in this project is the **Passive Aggressive Classifier**.

**Model Description**

The Passive Aggressive Classifier (PAC) is a type of online learning algorithm for binary classification tasks. It is well-suited for applications like fake news detection. The PAC algorithm updates its model continuously as new data arrives, making it efficient for real-time classification.

**Model Accuracy**

The Passive Aggressive Classifier achieved an impressive accuracy of **96%** during evaluation. This high accuracy indicates its effectiveness in classifying news articles as reliable or unreliable.

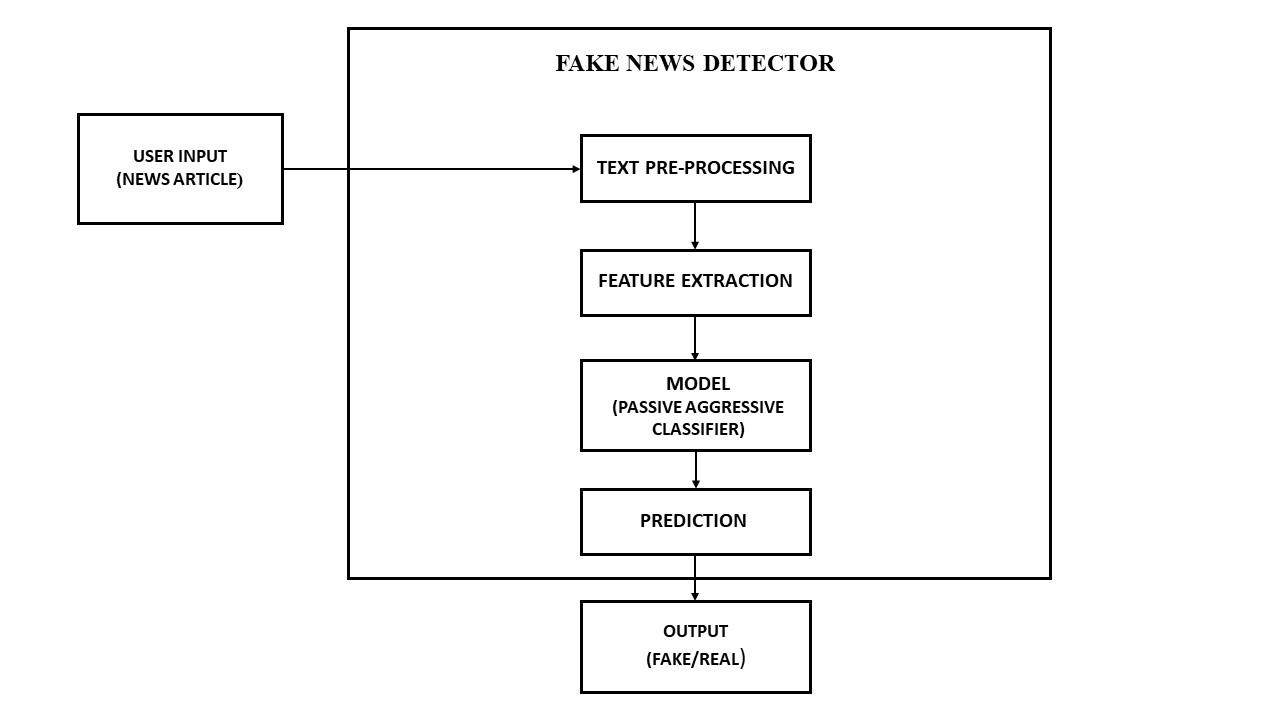
The model is pre-trained and available as model.pkl in this repository, allowing you to use it for making predictions.

Feel free to explore the Jupyter Notebook (Fake\_News\_Detector-PA.ipynb) for more details about the model's training and performance.

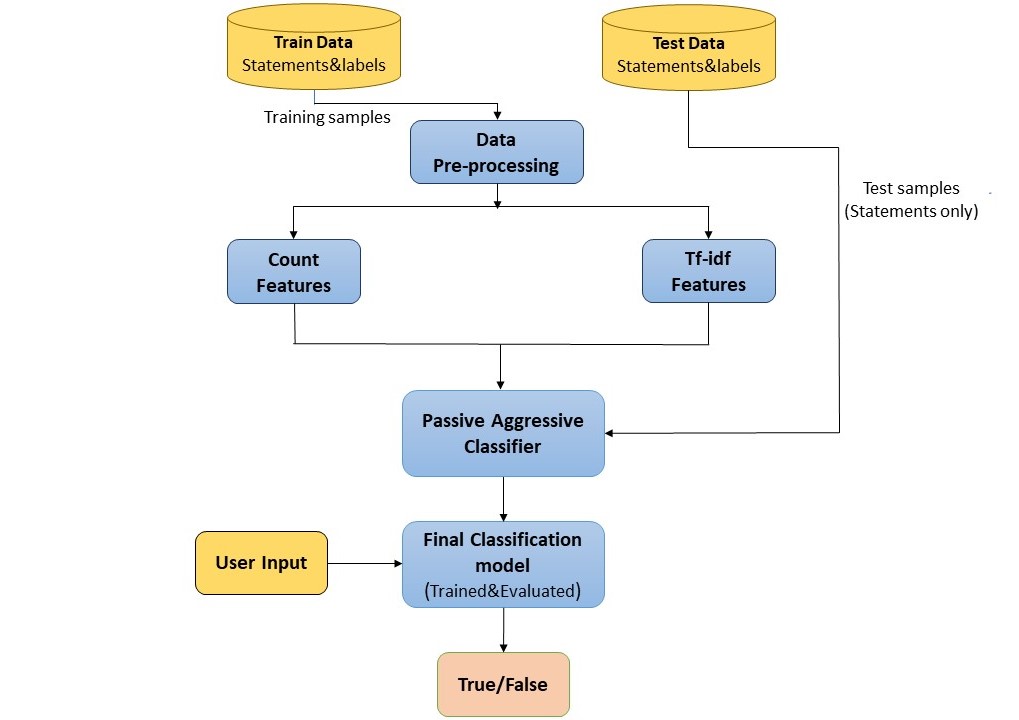
**Images**

This section provides visuals and diagrams used in the project:

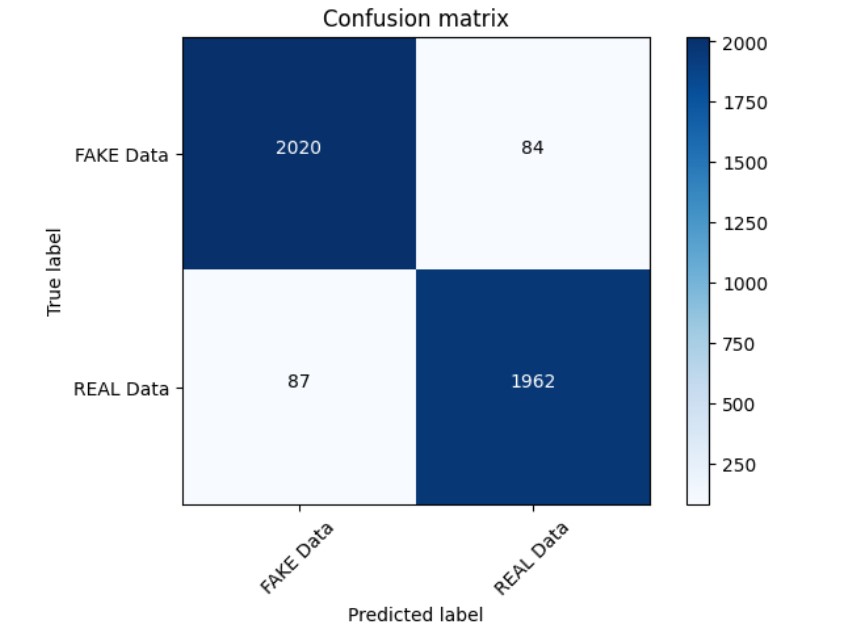
* Block Diagram

[](https://github.com/abiek12/Fake-News-Detection-using-MachineLearning/blob/main/Images/BlockDiagram.jpg)

* Process Flow Diagram

[](https://github.com/abiek12/Fake-News-Detection-using-MachineLearning/blob/main/Images/Processflow.jpg)

* Confusion Matrix

[](https://github.com/abiek12/Fake-News-Detection-using-MachineLearning/blob/main/Images/ConfusionMatrix.jpg)

**Prerequisites**

Before you begin, ensure you have met the following requirements:

* Python 3.7 or higher
* Install all dependencies from the requirements.txt file.

**Getting Started**

To get started with this project, follow these steps:

1. Clone the repository to your local machine:

git clone https://github.com/abiek12/Fake-News-Detection-using-MachineLearning.git

1. Create a virtual environment (optional but recommended):

python -m venv my\_env

1. Activate the virtual environment:
2. # On Windows
3. .\my\_env\Scripts\Activate.ps1
4. # On macOS and Linux

source my\_env/bin/activate

1. Install project dependencies:

pip install -r requirements.txt

1. Run the web application:

python app.py

Access the application in your web browser by navigating to http://localhost:5000.