FAKE NEWS DETECTION USING PYTHON

A project report submitted in partial fulfillment of the requirements
for the award of credits to
Data Pre-Processing and Data Visualization
skill - oriented course of

Bachelor of Technology

In

ELECTRONICS & COMMUNICATION ENGINEERING

By

GOPALAM DIVYA SREE
22BQ1A0434

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING
VASIREDDY VENKATADRI INSTITUTE OF TECHNOLOGY
(Approved by AICTE and permanently affiliated to JNTUK)
Accredited by NBA and NAAC with 'A' Grade
NAMBUR (V), PEDAKAKANI (M), GUNTUR-522 508
DECEMBER 2023

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

VASIREDDY VENKATADRI INSTITUTE OF TECHNOLOGY: NAMBUR

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY KAKINADA

CERTIFICATE

This is to certify that the project titled “**FAKE NEWS DETECTION USING PYTHON**” is a bonafide record of work done by **GOPALAM DIVYA SREE** under the guidance **of Dr.M.Venkatesh, Mr.B.V.Sathish Kumar, Sk.Mastanbi , ASSISTANT PROFESSORS** in partial fulfillment of the requirement for the award of credits to **Data Preprocessing and Data Visualization** - a skill-oriented course of Bachelor of Technology in Electronics and Communication Engineering, JNTUK during the academic year 2023–24.

Dr.M.Venkatesh, Assistant Professor
Course Instructor

Dr. M.Y. Bhanu Murthy
Head of the Department

DECLARATION

I **GOPALAM DIVYA SREE (22BQ1A0434)** hereby declare that the Project Report entitled “**FAKE NEWS DETECTION”** done by me under the guidance of **Dr.M.Venkatesh, Mr.B.V.Sathish Kumar, Sk.Mastanbi, Assistant Professors, Department of ECE** is submitted in partial fulfillment of the requirements for the award of degree **of BACHELOR OF TECHNOLOGY IN ELECTRONICS AND COMMUNICATION ENGINEERING**

DATE : **SIGNATURE OF THE CANDIDATE**

PLACE : VVIT, NAMBUR. **GOPALAM DIVYA SREE**

ACKNOWLEDGEMENT

I express my sincere thanks wherever it is due I express my sincere thanks to the Chairman, Vasireddy Venkatadri Institute of Technology, Sri Vasireddy Vidya Sagar for providing me well equipped infrastructure and environment.

I thank Dr.Y.Mallikarjuna Reddy, Principal, Vasireddy Venkatadri Institute of Technology, Nambur, for providing me the resources for carrying out the project.

I express my sincere thanks to Dr. K. Giribabu, Dean of Studies for providing support and stimulating environment for developing the project.

My sincere thanks to Dr. M. Y.Bhanumurthy, Head of the Department, Department of ECE, for his co-operation and guidance which help us to make our project successful and complete in all aspects.

I also express my sincere thanks and are grateful to our guide Dr.M.Venkatesh, Mr.B.V.Sathish Kumar, Sk.Mastanbi, Assistant Professor, Department of ECE, for motivating me to make our project successful and fully complete. I am grateful for her preciousguidance and suggestions.

I also place our floral gratitude to all other teaching staff and lab technicians for their constant support and advice throughout the project.

NAME OF THE CANDIDATE
GOPALAM DIVYA SREE
(22BQ1A0434)

CONTENTS

LIST OF FIGURES

ABSTRACT

Indian politics suffered from a great set back due to fake news. Fake news is intentionally written to mislead the audience to believe the false propaganda, which makes it difficult to detect based on news content. The fake news has hindered the mindset of the common people. Due to this widespread of the fake news online it is the need of the hour to check the authenticity of the news. The spread of fake news has the potential for extremely negative impact on society. The proposed approach is to use machine learning to detect fake news. Using vectorisation of the news title and then analysing the tokens of words with our dataset. The dataset we are using is a predefined curated list of news with their property of being a fake news or not. Our goal is to develop a model that classifies a given article as either true or fake.

The libraries I have used in the project are Sklearn and pandas. Sci-kit learn is a library in python that provides many unsupervised and supervised learning algorithm.

CHAPTER 1
INTRODUCTION

1.1 What are fake news?

“Fake news” is a term that has come to mean different things to different people. At itscore, we are defining “fake news” as those news stories that are false: the story itself is fabricated, with no verifiable facts, sources or quotes. Sometimes these stories may be propaganda that is intentionally designed to mislead the reader, or may be designed as “clickbait” written for economic incentives (the writer profits on the number of people who click on the story). In recent years, fake news stories have proliferated via social media, in part because they are so easily and quickly shared online.

In this paper I experiment the possibility to detect fake news based only on textual information by applying traditional machine learning techniques.

In order to work on fake news detection, it is important to understand what is fake news and how they are characterized. The first is characterization or what is fake news and the second is detection. In order to build detection models, it is need to start by characterization, indeed, it is need to understand what is fake news before trying to detect them.

1.2 Fake News Characterization

Fake news definition is made of two parts: authenticity and intent. Authenticity means that fake news content false information that can be verified as such, which means that conspiracy theory is not included in fake news as there are difficult to be proven true or false in most cases. The second part, intent, means that the false information has been written with the goal of misleading the reader.

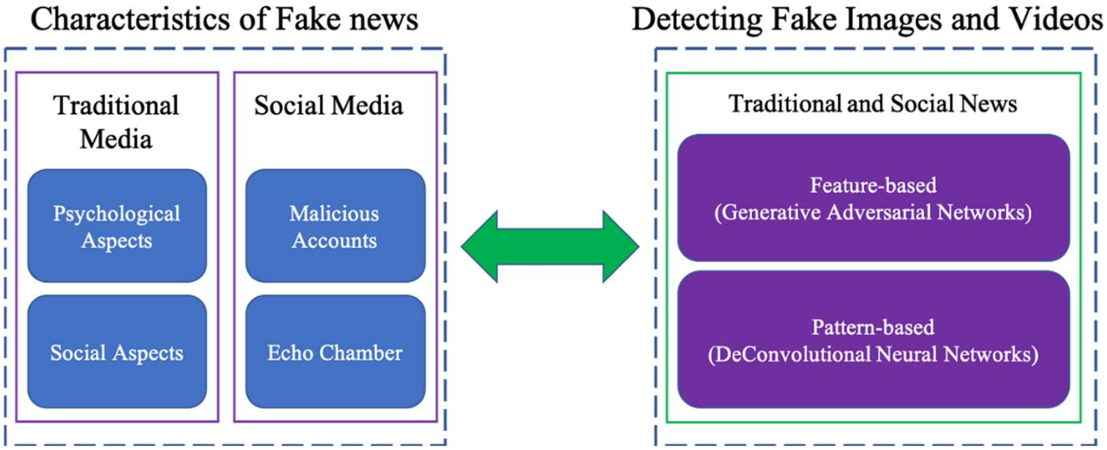

Characteristics of Fake news
Traditional Media
Social Media
Psychological Aspects
Social Aspects
Malicious Accounts
Echo Chamber
Detecting Fake Images and Videos
Traditional and Social News
Feature-based (Generative Adversarial Networks)
Pattern-based (DeConvolutional Neural Networks)

Fig 1.2.1 Fake news characterization

1.3 Fundamental Theories

Fundamental human cognition and behaviour theories developed across various disciplines, such as social sciences and economics, provide invaluable insights for fake news analysis. These theories can introduce new opportunities for qualitative and quantitative studies of big fake news data. These theories can also facilitate building well-justified and explainable models for fake news detection and intervention, which, to date, have been rarely available. We have conducted a comprehensive literature survey across various disciplines and have identified well-known theories that can be potentially used to study fake news. These theories are provided in Table 2 along with short descriptions, which are related to either (I) the news itself or (II) its spreaders.

I. News-related theories. For instance, theories have implied that fake news potentially differs from the truth in terms of, e.g., writing style and quality, quantity such as word counts, and sentiments expressed. It should be noted that these theories, developed by forensic psychology, target deceptive statements or testimonies but not fake news, though these are similar concepts. Thus, one research opportunity is to verify whether these attributes are statistically distinguishable among disinformation, fake news, and the truth, in particular, using big fake news data.

II. User-related theories. User-related theories investigate the characteristics of users involved in fake news activities Fake news, unlike information such as fake reviews, can "attract" both malicious and normal users. Malicious users spread fake news often

III. intentionally and are driven by benefits. Some normal users (which we denote as vulnerable normal users) can frequently and unintentionally spread fake news without recognizing the falsehood. Such vulnerability psychologically stems from (i) social impacts and (ii) self-impact.

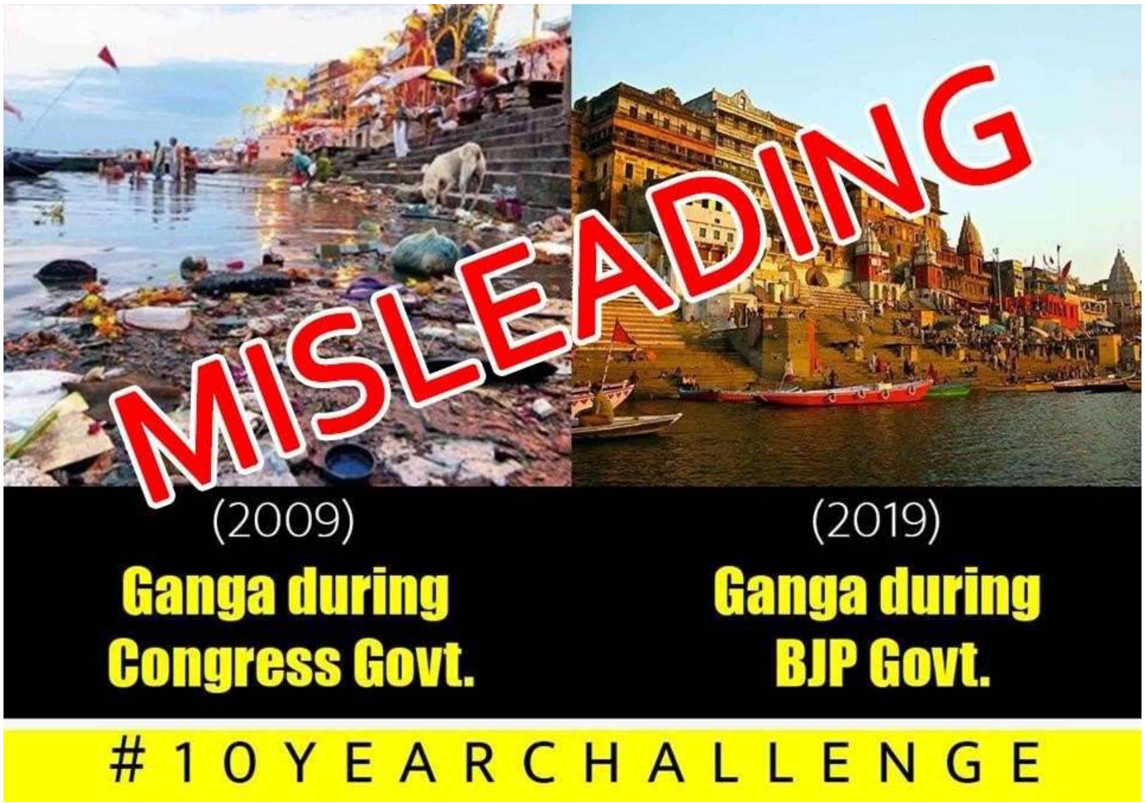
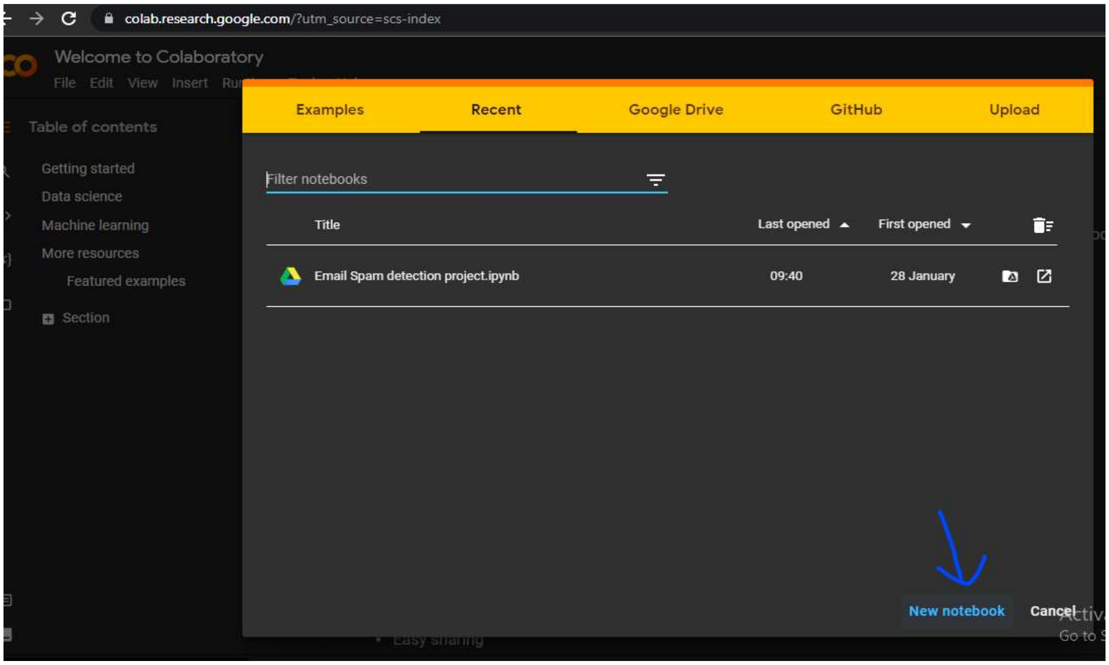


Fig 1.3.1 Misleading by fake news

1.4 STEPS TO USE GOOGLE COLAB

1. Type Google Colab or Google colaboratory in chrome or any web browser and click on its website.



Title	Last opened	First opened
Email Spam detection project.ipynb	09:40	28 January

Fig1.4.1:Opening Google colab

Then we will get a screen like this.click on the new notebook.

2.A new Colab notebook will be opened.It is in .ipynb form.

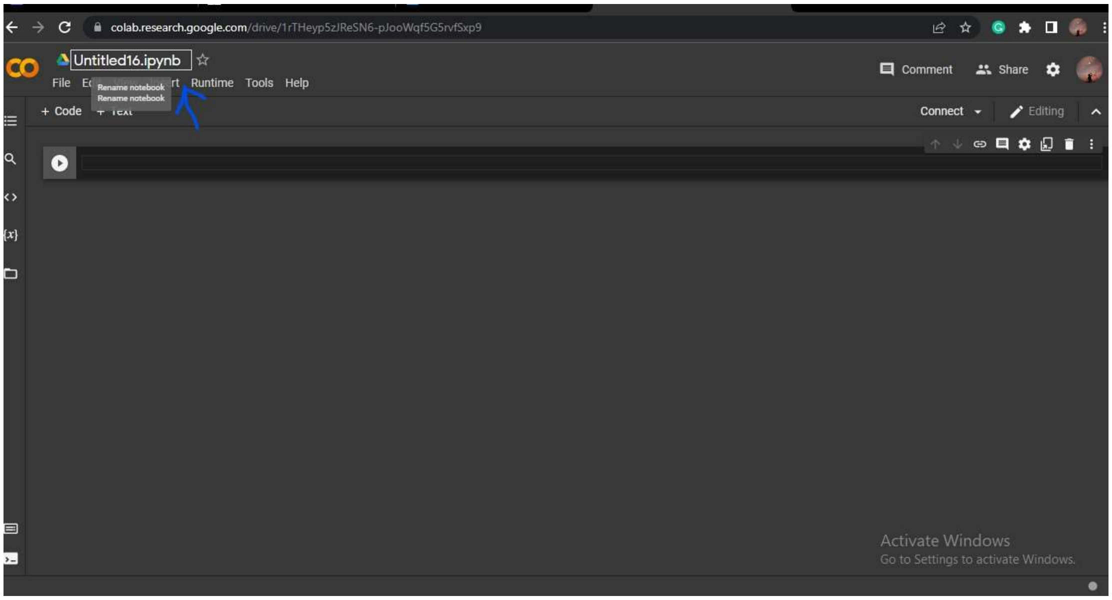
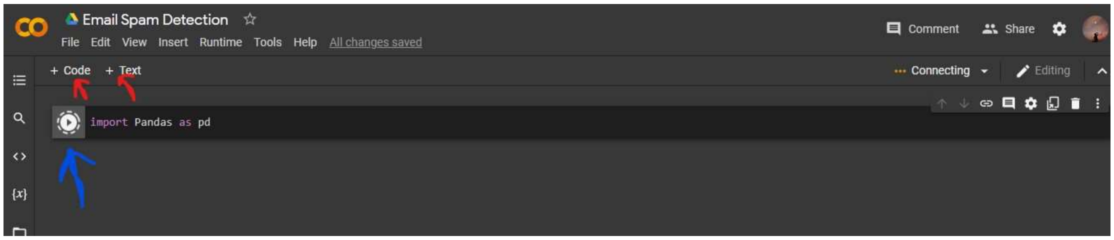


Fig 1.4.2:New notebook in Google Colab

We can simply change the name of the folder by clicking at untitled.ipynb

3.We can write code by clicking on code and we can also write by clicking on the text option.

4.we can run the code by clicking on the button which is at right of the code snippet.



```
import Pandas as pd
```

Fig 1.4.3:Writing and running the code in colab

1.5 Proposed Framework

In my proposed framework, I am expanding on the current literature by introducing ensemble techniques with various linguistic feature sets to classify news articles from multiple domains as true or fake. The ensemble techniques along with Linguistic Inquiry and Word Count (LIWC) feature set used in this research are the novelty of our proposed approach.

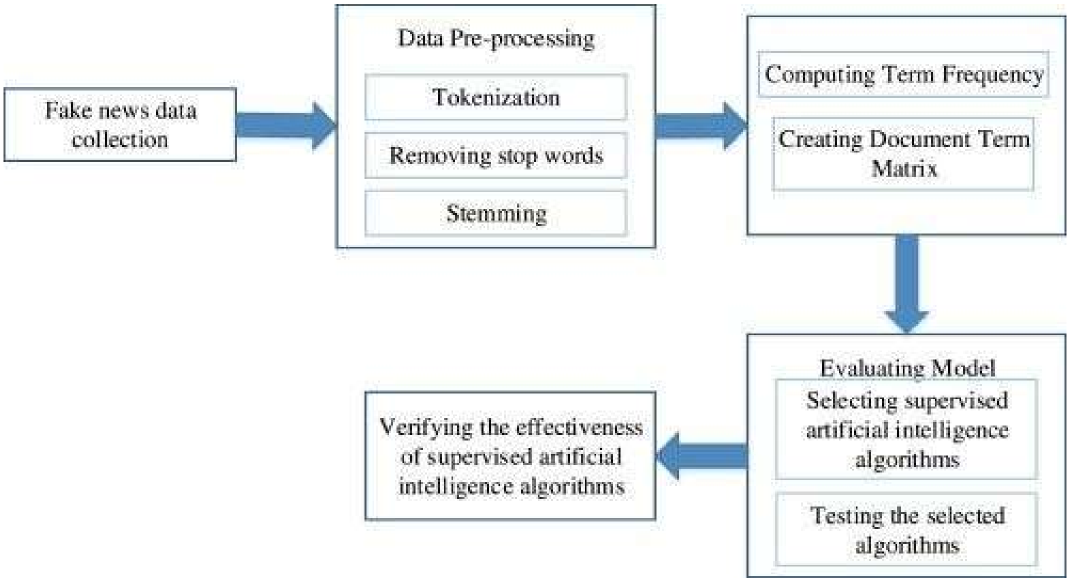

Fake news data collection
Data Pre-processing
Tokenization
Removing stop words
Stemming
Computing Term Frequency
Creating Document Term Matrix
Evaluating Model
Selecting supervised artificial intelligence algorithms
Testing the selected algorithms
Verifying the effectiveness of supervised artificial intelligence algorithms

Fig 1.5.1 Proposed framework

There are numerous reputed websites that post legitimate news contents which are used for fact checking. However, we selected three datasets for our experiments which contain news from multiple domain and contain a mix of both truthful and fake articles, and merged the three datasets into large dataset. The datasets are available online and are extracted from Kaggle.

CHAPTER 2
LIBRARIES DESCRIPTION

The libraries used in this project are Pandas,Sklearn,numpy,TfidfVectorizer, PassiveAggressiveClassifier

2.1 PANDAS

Pandas is an open-source python package that is mostly used for data science/data analysis and machine learning tasks. It is built on top of another project named Numpy, which provides support for multidimensional arrays. It provides a variety of tools for data manipulation and analysis. Pandas allows importing data from various file formats such as Comma-Separated Values(CSV files), JSON, SQL database tables or queries and Microsoft Excel. Pandas allow various data manipulation operations such as merging, reshaping, selecting as well as data cleaning and data wrangling features. It is flexible, powerful, fast and easy to use.

2.1.1 VITAL FEATURES OF PANDAS

- A fast and efficient Data Frame object for data manipulation with integrated indexing.
- Intelligence data alignment and integrated handling of missing data: gains automatic label-based alignment in computations and easily manipulates messy data into an orderly form.
- Intelligent label-based slicing, fancy indexing, and subsetting of large data sets
- High-performance merging and joining of data sets.
- Hierarchical axis indexing provides an intuitive way of working with high-dimensional data in a low-dimensional data structure.
- Python with pandas is used in a variety of academic and commercial domains including finance, Neuroscience, Economics, Statistics, Advertising, Web analytics and more.

2.2 SKLEARN

Scikit-learn in short sklearn is a machine learning library used in python which provides many unsupervised and supervised machine learning algorithms.it is one of the most useful libraries in python for machine learning. It is one of the simple and efficient tools for predictive data analysis. It is accessible to everybody. Sklearn is built on NumPy, SciPy and matplotlib.

The sklearn library contains many essential tools for machine learning and statistical modelling including classification, regression, clustering and dimensionality reduction. It also provides various tools for model fitting, data preprocessing, model selection, model evaluation and many other utilities.

2.2.1 IMPORTANT FEATURES OF SKLEARN

It provides diverse algorithms for

- Classification
  - Identifying which category an object belongs to
    Eg: Spam detection, Image recognition.
- Regression
  - Predicting a continuous-valued attribute associated with an object.
    Eg: Drug response, Stock prices
- Clustering
  - Automatic grouping of similar objects into sets.
    Eg: Customer segmentation, Grouping experiment outcomes.
- Dimensionality reduction
  - Reducing the number of random variables to consider.
    Eg: Visualisation, Increased efficiency.
- Model selection
  - Comparing, validating and choosing parameters and models.
    Eg: Improved accuracy via parameter tuning.
- Pre-processing
  - Feature extraction and normalisation

Eg: Transforming input data such as text for use with machine learning algorithms

2.3 NUMPY

NumPy is a Python library used for working with arrays.It also has functions for working in domain of linear algebra, fourier transform, and matrices.NumPy was created in 2005 by Travis Oliphant. It is an open source project and you can use it freely.

NumPy stands for Numerical.

2.3.1. Why Use NumPy?

In Python we have lists that serve the purpose of arrays, but they are slow to process.NumPy aims to provide an array object that is up to 50x faster than traditional Python lists.The array object in NumPy is called ndarray, it provides a lot of supporting functions that make working with ndarray very easy.Arrays are very frequently used in data science, where speed and resources are very important.

2.4 TfidfVectorizer

TF (Term Frequency): The number of times a word appears in a document is its Term Frequency. A higher value means a term appears more often than others, and so, the document is a good match when the term is part of the search terms.

IDF (Inverse Document Frequency): Words that occur many times a document, but also occur many times in many others, may be irrelevant. IDF is a measure of how significant a term is in the entire corpus.
The TfidfVectorizer converts a collection of raw document s into a matrix of TF-IDF features.

2.4.1 Terminologies:

Term Frequency: In document d, the frequency represents the number of instances of a given word t. Therefore, we can see that it becomes more relevant when a word appears in the text, which is rational. Since the ordering of terms is not significant, we can use a vector to describe the text in the bag of term models. For each specific term in the paper, there is an entry with the value being the term frequency.
The weight of a term that occurs in a document is simply proportional to the term frequency.

tf(t,d) = count of t in d / number of words in d

2.5 PassiveAgressiveClassifier

Passive Aggressive algorithms are online learning algorithms. Such an algorithm remains passive for a correct classification outcome, and turns aggressive in the event of a miscalculation, updating and adjusting. Unlike most other algorithms, it does not converge. Its purpose is to make updates that correct the loss, causing very little change in the norm of the weight vector.

The Passive-Aggressive algorithms are a family of Machine learning algorithms that are not very well known by beginners and even intermediate Machine Learning enthusiasts. However, they can be very useful and efficient for certain applications.

CHAPTER 3

IMPLEMENTATION

There are few steps involved in building any machine learning model.
For this project the steps involved in building Fake News Detection is:

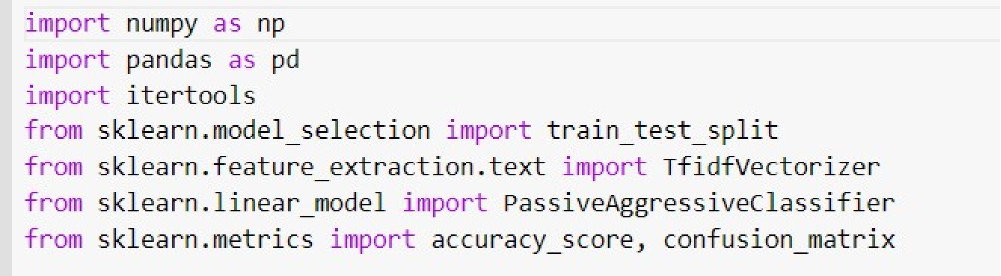
3.1 MAIN STEPS INVOLVED IN BUILDING FAKE NEWS DETECTION MODEL

1. Collecting dataset
2. Loading the data
3. Read the data into a DataFrame,and get the shape of the data
4. Split the dataset into training and testing sets
5. Read the data into a DataFrame,and get the shape of the data
6. Splitting training and testing data
7. Initializing Tfidvectorizer
8. Initializing PassiveAgressiveClassifier
9. Building Confusion Matrix

First of all, we have to import the required libraries

3.2 LOADING DEPENDENCIES

As I am using Google Colab for my project there is no need to install any libraries, we just need to import them.



```
import numpy as np
import pandas as pd
import itertools
from sklearn.model_selection import train_test_split
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.linear_model import PassiveAggressiveClassifier
from sklearn.metrics import accuracy_score, confusion_matrix
```

Fig 3.2.1: Importing libraries

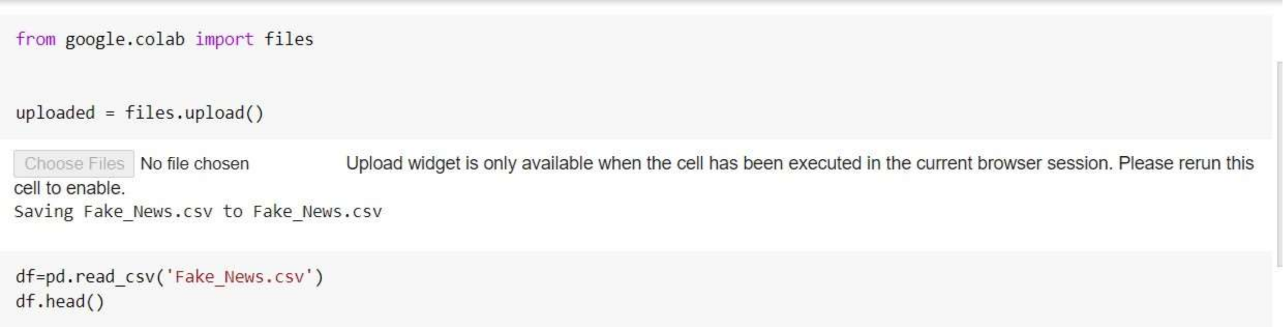
We have already discussed about numpy,pandas,Tfidvectorizer,PassiveAgressiveClassifier

3.3 DATA COLLECTION

Data set is vital for any machine learning program. Without a data set we cannot train the model.The output of the model depends on the dataset we feed. Based on the data set the model will get trained then only it will produce expected outcomes.Uploaded data set which is in CSV format contains a huge list of fake and true news.Data set contains two columns in which one column contain the message of the data another column contains the information whether that news is fake or true.

3.3.1 LOADING DATA

Now we will just load our data into a pandas dataframe uploaded=files.upload() method and read the file as 'df' using its read_csv() method.



```
from google.colab import files


uploaded = files.upload()
```

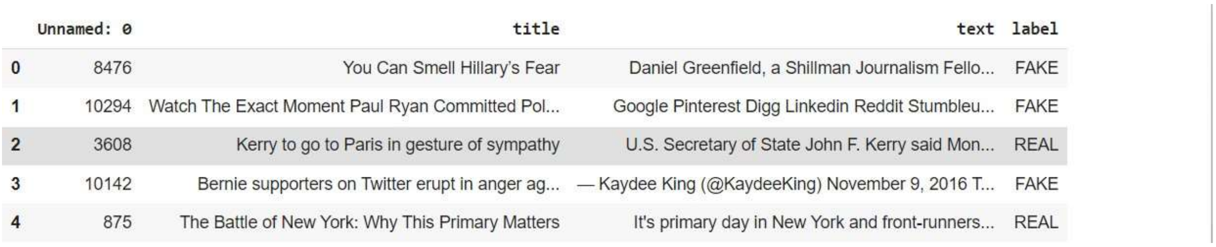
Choose Files No file chosen Upload widget is only available when the cell has been executed in the current browser session. Please rerun this cell to enable.

```
Saving Fake_News.csv to Fake_News.csv
```

```
df=pd.read_csv('Fake_News.csv')
df.head()
```

Fig 3.3.1 *Loading data*

Executing the above code returns a data frame



	Unnamed: 0	title	text	label
0	8476	You Can Smell Hillary's Fear	Daniel Greenfield, a Shillman Journalism Fello...	FAKE
1	10294	Watch The Exact Moment Paul Ryan Committed Pol...	Google Pinterest Digg Linkedin Reddit Stumbleu...	FAKE
2	3608	Kerry to go to Paris in gesture of sympathy	U.S. Secretary of State John F. Kerry said Mon...	REAL
3	10142	Bernie supporters on Twitter erupt in anger ag...	— Kaydee King (@KaydeeKing) November 9, 2016 T...	FAKE
4	875	The Battle of New York: Why This Primary Matters	It's primary day in New York and front-runners...	REAL

Fig 3.3.2 loaded data

3.4 Get the shape and head of the data

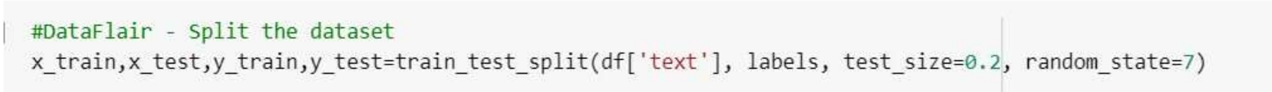
Now we will just read the data into a dataframe and get shape of the data which we have provided. After running the data we get the shape and head.

3.5 Getting the labels from the dataframe

Data labelling is an important part of data preprocessing for ML, particularly for supervised learning, in which both input and output data are labelled for classification to provide a learning basis for future data processing.Labeled datasets help to train machine learning model to identify and understand the recurring patterns in the input fed into them for delivering accurate output.

3. 6 Split the dataset into training and testing sets

Train/Test is a method to measure the accuracy of your model.It is called Train/Test because you split the data set into two sets: a training set and a testing set.Typically, when you separate a data set into a training set and testing set, most of the data is used for training, and a smaller portion of the data is used for testing.The main reason to keep train data large is if we train with more data then the accuracy of the model will increase.



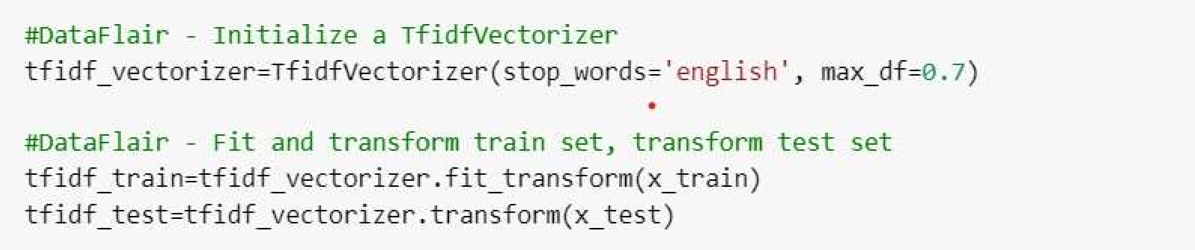
```
#DataFlair - Split the dataset
x_train,x_test,y_train,y_test=train_test_split(df['text'], labels, test_size=0.2, random_state=7)
```

Fig 3.6.1 split the dataset

3.7 Initialize Tfidvectorizer

Let’s initialize a TfidfVectorizer with stop words from the English language and a maximum document frequency of 0.7 (terms with a higher document frequency will be discarded). Stop words are the most common words in a language that are to be filtered out before processing the natural language data. And a TfidfVectorizer turns a collection of raw documents into a matrix of TF-IDF features.

Now, fit and transform the vectorizer on the train set, and transform the vectorizer on the test set.



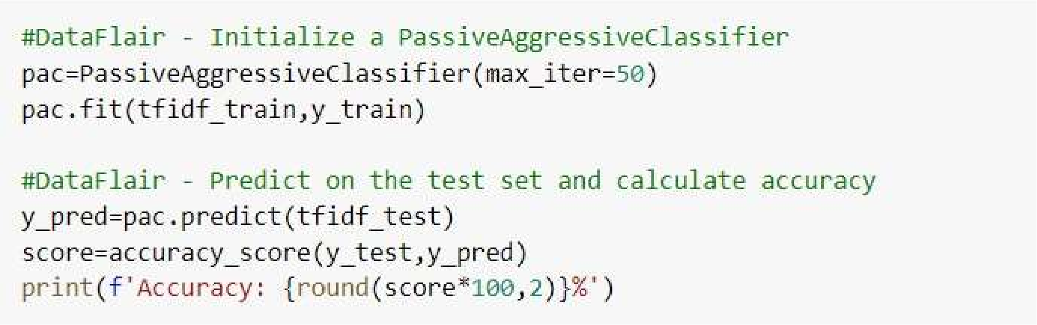
```
#DataFlair - Initialize a TfidfVectorizer
tfidf_vectorizer=TfidfVectorizer(stop_words='english', max_df=0.7)

#DataFlair - Fit and transform train set, transform test set
tfidf_train=tfidf_vectorizer.fit_transform(x_train)
tfidf_test=tfidf_vectorizer.transform(x_test)
```

Fig 3.7.1 initializing Tfidvectorizer

3.8 Initialize PassiveAgressiveClassifier

Next, we'll initialize a PassiveAggressiveClassifier. This is. We'll fit this on tfidf_train and y_train.

Then, we'll predict on the test set from the TfidfVectorizer and calculate the accuracy with accuracy_score() from sklearn.metrics.

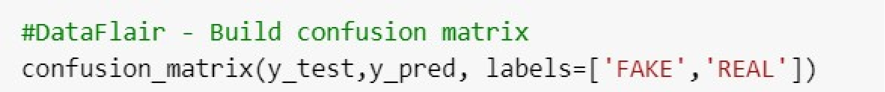


```
#DataFlair - Initialize a PassiveAggressiveClassifier
pac=PassiveAggressiveClassifier(max_iter=50)
pac.fit(tfidf_train,y_train)

#DataFlair - Predict on the test set and calculate accuracy
y_pred=pac.predict(tfidf_test)
score=accuracy_score(y_test,y_pred)
print(f'Accuracy: {round(score*100,2)}%')
```

Fig 3.8.1 initializing PassiveAgressiveClassifier

3.9 Build Confusion Matrix

Finally, let's print out a confusion matrix to gain insight into the number of false and true negatives and positives.



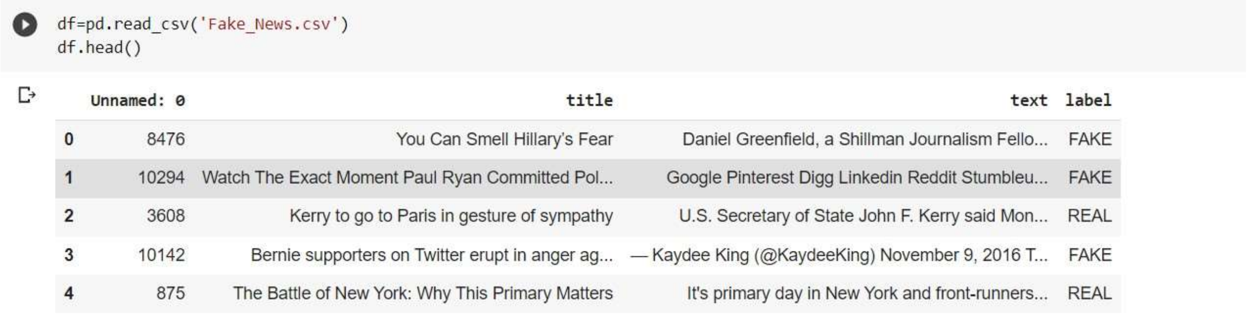
```
#DataFlair - Build confusion matrix
confusion_matrix(y_test,y_pred, labels=['FAKE','REAL'])
```

Fig 3.9.1 building confusion matrix

CHAPTER 4

RESULTS AND CONCLUSIONS

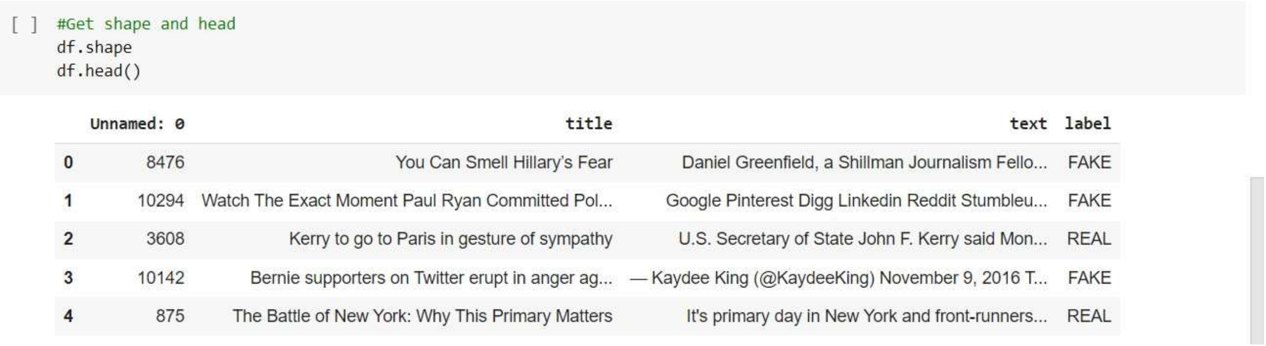
RESULTS



```
df=pd.read_csv('Fake_News.csv')
df.head()
```

	Unnamed: 0	title	text	label
0	8476	You Can Smell Hillary's Fear	Daniel Greenfield, a Shillman Journalism Fello...	FAKE
1	10294	Watch The Exact Moment Paul Ryan Committed Pol...	Google Pinterest Digg Linkedin Reddit Stumbleu...	FAKE
2	3608	Kerry to go to Paris in gesture of sympathy	U.S. Secretary of State John F. Kerry said Mon...	REAL
3	10142	Bernie supporters on Twitter erupt in anger ag...	— Kaydee King (@KaydeeKing) November 9, 2016 T...	FAKE
4	875	The Battle of New York: Why This Primary Matters	It's primary day in New York and front-runners...	REAL

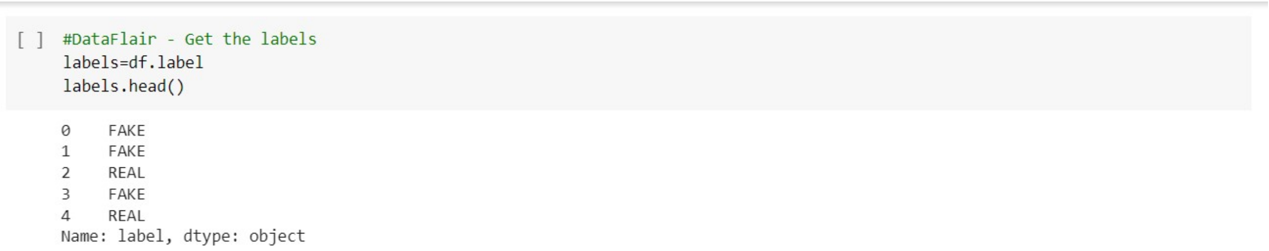
Fig 4.1



```
[ ] #Get shape and head
    df.shape
    df.head()
```

	Unnamed: 0	title	text	label
0	8476	You Can Smell Hillary's Fear	Daniel Greenfield, a Shillman Journalism Fello...	FAKE
1	10294	Watch The Exact Moment Paul Ryan Committed Pol...	Google Pinterest Digg Linkedin Reddit Stumbleu...	FAKE
2	3608	Kerry to go to Paris in gesture of sympathy	U.S. Secretary of State John F. Kerry said Mon...	REAL
3	10142	Bernie supporters on Twitter erupt in anger ag...	— Kaydee King (@KaydeeKing) November 9, 2016 T...	FAKE
4	875	The Battle of New York: Why This Primary Matters	It's primary day in New York and front-runners...	REAL

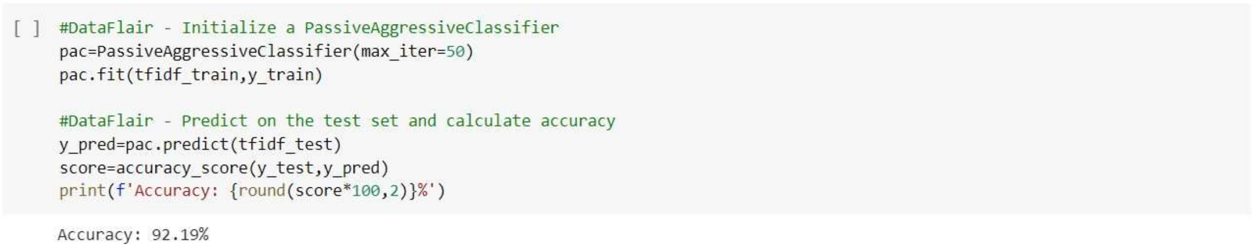
Fig 4.2



```
[ ] #DataFlair - Get the labels
    labels=df.label
    labels.head()
```

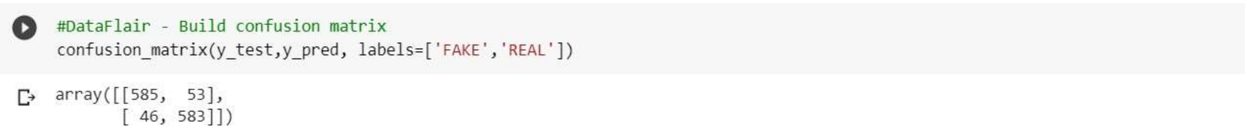
```
0    FAKE
1    FAKE
2    REAL
3    FAKE
4    REAL
Name: label, dtype: object
```

Fig 4.3



```
[ ]  #DataFlair - Initialize a PassiveAggressiveClassifier
     pac=PassiveAggressiveClassifier(max_iter=50)
     pac.fit(tfidf_train,y_train)

     #DataFlair - Predict on the test set and calculate accuracy
     y_pred=pac.predict(tfidf_test)
     score=accuracy_score(y_test,y_pred)
     print(f'Accuracy: {round(score*100,2)}%')

     Accuracy: 92.19%
```

Fig 4.4



```
#DataFlair - Build confusion matrix
confusion_matrix(y_test,y_pred, labels=['FAKE','REAL'])

array([[585,  53],
       [ 46, 583]])
```

Fig 4.5

CONCLUSION

So with this model, we have 589 true positives, 587 true negatives, 42 false positives, and 49 false negatives.

The task of classifying news manually requires in-depth knowledge of the domain and expertise to identify anomalies in the text. In this research, we discussed the problem of classifying fake news articles using machine learning models and ensemble techniques. The data we used in our work is collected from the KAGGLE and contains news articles from various domains to cover most of the news rather than specifically classifying political news. The primary aim of the research is to identify patterns in text that differentiate fake articles from true news.

The learning models were trained and parameter-tuned to obtain optimal accuracy. Some models have achieved comparatively higher accuracy than others. We used multiple performance metrics to compare the results for each algorithm. The ensemble learners have shown an overall better score on all performance metrics as compared to the individual learners.

Fake news detection has many open issues that require attention of researchers. For instance, in order to reduce the spread of fake news, identifying key elements involved in the spread of news is an important step. Graph theory and machine learning techniques can be employed to identify the key sources involved in spread of fake news. Likewise, real time fake news identification in videos can be another possible future direction.

Finally, this application is only one that would be necessary in a larger toolbox that could function as a highly accurate fake news classifier. Other tools that would need to be built may include a fact detector and a stance detector. In order to combine all of these “routines,” there would need to be some type of model that combines all of the tools and learns how to weight each of them in its final decision.

REFERENCES

[1] Natali Ruchansky, Sungyong Seo, and Yan Liu. Csi: A hybrid deep model for fake news detection. In Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, pages 797–806. ACM, 2017.

[2] James Thorne, Mingjie Chen, Giorgos Myrianthous, Jiashu Pu, Xiaoxuan Wang, and Andreas Vlachos. Fake news stance detection using stacked ensemble of classifiers. In Proceedings of the 2017 EMNLP Workshop: Natural Language Processing meets Journalism, pages 80–83, 2017.

[3] Mykhailo Granik and Volodymyr Mesyura. Fake news detection using naive bayes classifier. In 2017 IEEE First Ukraine Conference on Electrical and Computer Engineering (UKRCON), pages 900–903. IEEE, 2017

[4] Yaqing Wang, Fenglong Ma, Zhiwei Jin, Ye Yuan, Guangxu Xun, Kishlay Jha, Lu Su, and Jing Gao. Eann: Event adversarial neural networks for multi-modal fake news detection. In Proceedings of the 24th acm sigkdd international conference on knowledge discovery & data mining, pages 849–857. ACM, 2018.

[5] W. Y. Wang, “” liar, liar pants on fire”: A new benchmark dataset for fake news detection,” arXiv preprint arXiv:1705.00648, 2017.

[6] T. Ahmad, H. Akhtar, A. Chopra, and M. Waris Akhtar, “Satire detection from web documents using
machine learning methods,” pp. 102–105, 09 2014.

[7] J. Soll, T. Rosenstiel, A. D. Miller, R. Sokolsky, and J. Shafer. (2016, Dec) The long and brutal history of fake news. [Online]. Available: https://www.politico.com/magazine/story/2016/12/fake-news-history-long-violent-214535

[8] C. Wardle. (2017, May) Fake news. it’s complicated. [Online]. Available: https://firstdraftnews.com/
fake-news-complicated/

[9] C. Kang and A. Goldman. (2016, Dec) In washington pizzeria attack, fake news brought real guns. [Online]. Available: https://www.nytimes.com/2016/12/05/business/media/comet-ping-pong-pizza-shooting-fake-news-consequences.html

[10] C. Domonoske. (2016, Nov) Students have ’dismaying’ inability to tell fake news from real, study finds.[Online]. Available: https://www.npr.org/sections/thetwo-way/2016/11/23/503129818/study-finds-students-have-dismaying-inability-to-tell-fake-news-from-real

APPENDIX

```
import numpy as np
import pandas as pd
import itertools
from sklearn.model_selection import train_test_split
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.linear_model import PassiveAggressiveClassifier
from sklearn.metrics import accuracy_score, confusion_matrix
from google.colab import files
uploaded = files.upload()
df=pd.read_csv('Fake_News.csv')
df.head()
df.shape
df.head()
labels=df.label
labels.head()
x_train,x_test,y_train,y_test=train_test_split(df['text'], labels, test_size=0.2, random_state=7)
tfidf_vectorizer=TfidfVectorizer(stop_words='english', max_df=0.7)
tfidf_train=tfidf_vectorizer.fit_transform(x_train)
tfidf_test=tfidf_vectorizer.transform(x_test)
pac=PassiveAggressiveClassifier(max_iter=50)
pac.fit(tfidf_train,y_train)
y_pred=pac.predict(tfidf_test)
score=accuracy_score(y_test,y_pred)
print(f'Accuracy: {round(score*100,2)}%')
confusion_matrix(y_test,y_pred, labels=['FAKE','REAL'])
```