A Project Report on

Fake News Detection

Held at

NPTEL/ Spoken Tutorial Project at IIT Bombay

Submitted in partial fulfillment of award of

BACHELOR OF TECHNOLOGY

degree
in
Computer Science and Engineering

By
Shaili Rajput (1900820100134)
Shivani (1900820100138)
Shubham (1900820100143)
Shubham Sharma (1900820100145)
Tabish Absar (1900820100153)

Under the Guidance of
Ms. Kanchan
Assistant Professor

Department of Computer Science and Engineering

Moradabad Institute of Technology
Moradabad (U.P.)

SESSION: 2021-22

CERTIFICATE

Certified that the Project Report entitled **"FAKE NEWS DETECTION"** submitted by **Shaili Rajput(1900820100134)**, **Shivani(1900820100138)**, **Shubham(1900820100143)**, **Shubham Sharma(1900820100045)**, **Tabish Absar(1900820100153)** is their own work and has been carried out in our supervision. It is recommended that the candidates may now be evaluated for their project work by the university.

Date:

<div align="right">

MS. KANCHAN
Assistant Professor
Department of Computer Science and Engineering

</div>

ABSTRACT

In our modern era where the internet is ubiquitous, everyone relies on various online resources for news. Along with the increase in the use of social media platforms like Facebook, Twitter, etc. news spread rapidly among millions of users within a very short span of time.

The spread of fake news has far-reaching consequences like the creation of biased opinions to swaying election outcomes for the benefit of certain candidates. Moreover, spammers use appealing news headlines to generate revenue using advertisements via click-baits.

In this paper, we aim to perform binary classification of various news articles available online with the help of concepts pertaining to Artificial Intelligence, Natural Language Processing and Machine Learning. We aim to provide the user with the ability to classify the news as fake or real and also check the authenticity of the website publishing the news.

ACKNOWLEDGEMENT

SHAILI RAJPUT (1900820100134)

SHIVANI (1900820100138)

SHUBHAM (1900820100143)

SHUBHAM SHARMA (1900820100145)

TABISH ABSAR (1900820100153)

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Introduction

As an increasing amount of our lives is spent interacting online through social media platforms, more and more people tend to hunt out and consume news from social media instead of traditional news organizations. The explanations for this alteration in consumption behaviors are inherent within the nature of those social media platforms: it's often more timely and fewer expensive to consume news on social media compared with traditional journalism, like newspapers or television; and (ii) it's easier to further share, discuss, and discuss the news with friends or other readers on social media. For instance, 62 percent of U.S. adults get news on social media in 2016, while in 2012; only 49 percent reported seeing news on social media.

It had been also found that social media now outperforms television because the major news source. Despite the benefits provided by social media, the standard of stories on social media is less than traditional news organizations. However, because it's inexpensive to supply news online and far faster and easier to propagate through social media, large volumes of faux news, i.e., those news articles with intentionally false information, are produced online for a spread of purposes, like financial and political gain. it had been estimated that over 1 million tweets are associated with fake news "Pizzagate" by the top of the presidential election. Given the prevalence of this new phenomenon, "Fake news" was even named the word of the year by the Macquarie dictionary in 2016.

The extensive spread of faux news can have a significant negative impact on individuals and society. First, fake news can shatter the authenticity equilibrium of the news ecosystem for instance; it's evident that the most popular fake news was even more outspread on Facebook than the most accepted genuine mainstream news during the U.S. 2016 presidential election. Second, fake news intentionally

persuades consumers to simply accept biased or false beliefs. Fake news is typically manipulated by propagandists to convey political messages or influence for instance, some report shows that Russia has created fake accounts and social bots to spread false stories. Third, fake news changes the way people interpret and answer real news, for instance, some fake news was just created to trigger people's distrust and make them confused; impeding their abilities to differentiate what's true from what's not. To assist mitigate the negative effects caused by fake news (both to profit the general public and therefore the news ecosystem). It's crucial that we build up methods to automatically detect fake news broadcast on social media.

Internet and social media have made the access to the news information much easier and comfortable. Often Internet users can pursue the events of their concern in online form, and increased number of the mobile devices makes this process even easier. But with great possibilities come great challenges. Mass media have an enormous influence on the society, and because it often happens, there's someone who wants to require advantage of this fact.

1.2 Aim and Objectives

The main objective behind the development and upgradation of existing projects are the following smart approaches:

- Be Aware of such article while forwarding to others
- Reveal True stories
- Prevent from false crisis events
- Be Informative

1.3 Motivation

Machine learning (ML) is a type of artificial intelligence (AI) that allows software applications to become more accurate at predicting outcomes without being explicitly programmed to do so. Machine learning algorithms use historical data as input to predict new output values.

The extensive spread of faux news can have a significant negative impact on individuals and society. First, fake news can shatter the authenticity equilibrium of the news ecosystem for instance.

Understanding the truth of new and message with news detection can create positive impact on the society.

1.4 Scope

The usage of this system greatly reduces the time required to search for a place leading to quicker decision making with respect to places to visit. Used to view the location view (the user can even zoom in and zoom out to get a better view) as well as 360 degree image embedded in the application. The System makes use of weather underground API for fetching the details of weather at accuracy.

The user can also find the paths to follow to reach the final destination in map which gives a better view to the users. It becomes convenient for users to book their tour via website instead of visiting agency ultimately saves time and money.

CHAPTER 2

TECHNOLOGY DESCRIPTION

2.1 Introduction

Our project is an web application which gives you the guidance of the day to day routine of fake news, spam message in daily news channel, Facebook, Twitter, Instagram and other social media. We have shown some data analysis from our dataset which have retrieve from many online social media and display the main source till now fake news and true news are engaged.

Our project is tangled with multiple model trained by our own and also some pre trained model extracted from Felipe Adachi. The accuracy of the model is around 95% for all the self made model and 97% for this pre trained model. This model can detect all news and message which are related to covid-19, political news, geology ,etc.

2.2 Existing System

We can get online news from different sources like social media websites, search engine, homepage of news agency websites or the factchecking websites. On the Internet, there are a few publicly available datasets for Fake news classification like Buzzfeed News, LIAR, BS Detector etc. These datasets have been widely used in different research papers for determining the veracity of news. In the following sections, I have discussed in brief about the sources of the dataset used in this work.This Existing system can help us to trained our model using machine learning technique.

2.3 Need of New System

Currently, many people are using the internet as a central platform to find the information about reality in world and need to be continue.Hence I has mention above we will create fake news and message detection model which detect the reality of the news and message. Also whose use our website can see the upto date about main source or keyword are getting most fake news and message and mapoed up with chart. After and all everyone want to know how to prevent this hence we are giving some important tips to avoid this fake news of spreading rumor in the world.

2.4 Problems Definition

The system is an Web application which help user to detect the fake news. We have given the text box where the user has the option to paste the message or paste the url link of the news and other message link and after that it gives the reality of it. All the user gives data to detector may save for further use in order to update the statue of model, data analysis in future. We also help user by giving some guidance of how to prevent from such false event and how to stop with such event from spreading it.

2.5 Language

2.5.1 Python

Python is an interpreted, high-level and general-purpose programming language. Python's design philosophy emphasizes code readability with its notable use of significant whitespace. Its language constructs and object-oriented approach aim to help programmers write clear, logical code for small and large-scale projects.

Python was created in the late 1980s, and first released in 1991, by Guido van Rossum as a successor to the ABC programming language. Python 2.0, released in 2000, introduced new features, such as list comprehensions, and a garbage collection system with reference counting, and was discontinued with version 2.7 in 2020. Python 3.0, released in 2008, was a major revision of the language that is not completely backward-compatible and much Python 2 code does not run unmodified on Python 3. With Python 2's end-of-life (and pip having dropped support in 2021), only Python 3.6.x and later are supported, with older versions still supporting e.g. Windows 7 (and old installers not restricted to 64-bit Windows).

Python is a multi-paradigm programming language. Object-oriented programming and structured programming are fully supported, and many of its features support functional programming and aspect-oriented programming (including by metaprogramming and metaobjects (magic methods)). Many other paradigms are supported via extensions, including design by contract and logic programming.

Python uses dynamic typing and a combination of reference counting and a cycle-detecting garbage collector for memory management. It also features dynamic name resolution (late binding), which binds method and variable names during program execution.

Python's design offers some support for functional programming in the Lisp tradition. It has filter, map, and reduce functions; list comprehensions, dictionaries, sets, and generator expressions. The standard library has two modules (itertools and functools) that implement functional tools borrowed from Haskell and Standard ML.

The language's core philosophy is summarized in the document The Zen of Python (PEP 20), which includes aphorisms such as:

- Beautiful is better than ugly.
- Explicit is better than implicit.
- Simple is better than complex.
- Complex is better than complicated.
- Readability counts.

Rather than having all of its functionality built into its core, Python was designed to be highly extensible. This compact modularity has made it particularly popular as a means of adding programmable interfaces to existing applications. Van Rossum's vision of a small core language with a large standard library and easily extensible interpreter stemmed from his frustrations with ABC, which espoused the opposite approach.

An important goal of Python's developers is keeping it fun to use. This is reflected in the language's name—a tribute to the British comedy group Monty Python—and in occasionally playful approaches to tutorials and reference materials, such as examples that refer to spam and eggs (from a famous Monty Python sketch) instead of the standard foo and bar.

Python's large standard library, commonly cited as one of its greatest strengths,[109] provides tools suited to many tasks. For Internet-facing applications, many standard formats and protocols such as MIME and HTTP are supported. It includes modules for creating graphical user interfaces, connecting to relational databases, generating pseudorandom numbers, arithmetic with arbitrary-precision decimals,[110] manipulating regular expressions, and unit testing.

As of January 2020, the Python Package Index (PyPI), the official repository for third-party Python software, contains over 287,000[112] packages with a wide range of functionality, including:

- Automation

- Data analytics
- Databases
- Documentation
- Graphical user interfaces
- Image processing
- Machine learning
- Mobile App
- Multimedia
- Computer Networking
- Scientific computing
- System administration
- Test frameworks
- Text processing
- Web frameworks
- Web scraping

2.6 Libraries

2.6.1 NumPy

NumPy is also a python package which stands for Numerical python.NumPy is an open-source numerical Python library.

NumPy contains a multi-dimensional array and matrix data structures.

It can be utilized to perform a number of mathematical operations on arrays such as trigonometric, statistical, and algebraic routines. Therefore, the library contains a large number of mathematical, algebraic, and transformation functions.

NumPy is an extension of Numeric and Numarray.

Numpy also contains random number generators.

2.6.2 Pandas

Pandas is a Python library that is used for faster data analysis, data cleaning, and data pre-processing. Pandas is built on top of the numerical library of Python, called numpy.

Before you install pandas, make sure you have numpy installed in your system. If numpy is not much familiar to you, then you need to have a look at this article. Brush up your numpy skills and then learn pandas.

You might have heard about data-frames, which is a common term in machine learning. This word comes from pandas. Pandas library helps us to make data-frames easily. Later in this tutorial, we will talk about data frames in detail.

Pandas library is often compared to excel sheets. A lot of features in excel sheets are available in the pandas as well.

2.6.3 Matplotlib

Matplotlib is an amazing visualization library in Python for 2D plots of arrays. Matplotlib is a multi-platform data visualization library built on NumPy arrays and designed to work with the broader SciPy stack. It was introduced by John Hunter in the year 2002.

One of the greatest benefits of visualization is that it allows us visual access to huge amounts of data in easily digestible visuals. Matplotlib consists of several plots like line, bar, scatter, histogram etc.

2.6.4 Seaborn

Seaborn is a Python **data visualization library** based on matplotlib. It provides a high-level interface for drawing attractive and informative statistical graphics. For a brief introduction to the ideas behind the library, you can read the introductory notes or the paper.

2.6.5 Sklearn

Scikit-learn (Sklearn) is the most useful and robust library for machine learning in Python. It provides a selection of efficient tools for machine learning and statistical modeling including classification, regression, clustering and dimensionality reduction via a consistence interface in Python. This library, which is largely written in Python, is built upon **NumPy, SciPy** and **Matplotlib**.

2.7 Design tool and Environment

2.7.1 Jupyter Notebook

The Jupyter Notebook is an open source web application that you can use to create and share documents that contain live code, equations, visualizations, and text. Jupyter Notebook is maintained by the people at Project Jupyter.

Jupyter Notebooks are a spin-off project from the IPython project, which used to have an IPython Notebook project itself. The name, Jupyter, comes from the core supported programming languages that it supports: Julia, Python, and R. Jupyter ships with the IPython kernel, which allows you to write your programs in Python, but there are currently over 100 other kernels that you can also use.

Jupyter notebooks have three particularly strong benefits:

- They're great for showcasing your work. You can see both the code and the results. The notebooks at Kaggle is a particularly great example of this.
- It's easy to use other people's work as a starting point. You can run cell by cell to better get an understanding of what the code does.
- Very easy to host server side, which is useful for security purposes. A lot of data is sensitive and should be protected, and one of the steps toward that is no data is stored on local machines. A server-side Jupyter Notebook setup gives you that for free.

CHAPTER 3

ANALYSIS AND OUTPUT

3.1 Proposed System

The system is an Web application which help user to detect the fake news. We have given the text box where the user has the option to paste the message or paste the url link of the news and other message link and after that it gives the reality of it. All the user gives data to detector may save for further use in order to update the statue of model, data analysis in future. We also help user by giving some guidance of how to prevent from such false event and how to stop with such event from spreading it.
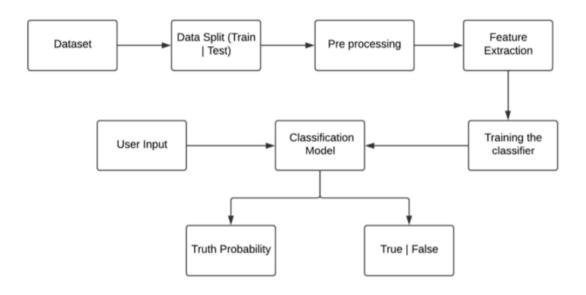
3.2 System Architecture Design

Fig 3.2.1 System Architecture Design

3.3 System Design Diagram

FAKE NEWS DETECTOR

Text Message or Text News → VECTOR MODEL → CLASSIFICATION MODEL → RESULT
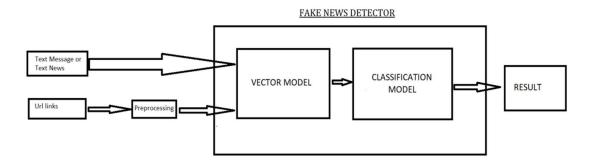
Url links → Preprocessing →

Fig 3.3.1 System Design Diagram

3.4.1 Logistic Regression

The logistic function, also called the sigmoid function was developed by statisticians to describe properties of population growth in ecology, rising quickly and maxing out at the carrying capacity of the environment. It's an S-shaped curve that can take any real-valued number and map it into a value between 0 and 1, but never exactly at those limits.

$$\text{sigmoid } (Z) = 1 / (1 + e^{-z})$$
$$\text{Hypothesis} \Rightarrow Z = WX + B$$
$$h\Theta(x) = \text{sigmoid } (Z)$$

3.4.2 Decision Tree Classification

Decision Tree is a Supervised learning technique that can be used for both classification and Regression problems, but mostly it is preferred for solving Classification problems. It is a tree-structured classifier, where internal nodes represent the features of a dataset, branches represent the decision rules and each leaf node represents the outcome.

23

3.4.3 Gradient Boosting Classifier

Gradient Boosting is a popular boosting algorithm. In gradient boosting, each predictor corrects its predecessor's error. In contrast to Adaboost, the weights of the training instances are not tweaked, instead, each predictor is trained using the residual errors of predecessor as labels.

3.4.4 Random Foresst Classifier

Random Forest is a trademark term for an ensemble of decision trees. In Random Forest, we've collection of decision trees (so known as "Forest"). To classify a new object based on attributes, each tree gives a classification and we say the tree "votes" for that class. The forest chooses the classification having the most votes (over all the trees in the forest). The random forest is a classification algorithm consisting of many decisions trees. It uses bagging and feature randomness when building each individual tree to try to create an uncorrelated forest of trees whose prediction by committee is more accurate than that of any individual tree. Random forest, like its name implies, consists of a large number of individual decision trees that operate as an ensemble. Each individual tree in the random forest spits out a class prediction and the class with the most votes becomes our model's prediction. The reason that the random forest model works so well is: A large number of relatively uncorrelated models (trees) operating as a committee will outperform any of the individual constituent models.

CHAPTER 4

SOURCE CODE

```python
import pandas as pd

import numpy as np

import seaborn as sns

import matplotlib.pyplot as plt

from sklearn.model_selection import train_test_split

from sklearn.metrics import accuracy_score

from sklearn.metrics import classification_report

import re

import string

df_fake = pd.read_csv("Fake.csv")

df_true = pd.read_csv("True.csv")

df_fake.head(5)

df_true.head(5)

df_fake["class"] = 0

df_true["class"] = 1

df_fake.shape, df_true.shape

df_fake.shape, df_true.shape

df_fake_manual_testing["class"] = 0

df_true_manual_testing["class"] = 1

df_fake_manual_testing.head(10)

df_true_manual_testing.head(10)

df_manual_testing = pd.concat([df_fake_manual_testing,df_true_manual_testing], axis = 0)

df_manual_testing.to_csv("manual_testing.csv")
```

```python
df_marge = pd.concat([df_fake, df_true], axis =0 )

df_marge.head(10)

df_marge.columns

df = df_marge.drop(["title", "subject","date"], axis = 1)

df.isnull().sum()

df = df.sample(frac = 1)

df.head()

df.reset_index(inplace = True)

df.drop(["index"], axis = 1, inplace = True)

df.columns

df.head()

def wordopt(text):

    text = text.lower()

    text = re.sub('\[.*?\]', '', text)

    text = re.sub("\\W"," ",text)

    text = re.sub('https?://\S+|www\.\S+', '', text)

    text = re.sub('<.*?>+', '', text)

    text = re.sub('[%s]' % re.escape(string.punctuation), '', text)

    text = re.sub('\n', '', text)

    text = re.sub('\w*\d\w*', '', text)

    return text

df["text"] = df["text"].apply(wordopt)

x = df["text"]

y = df["class"]

x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.25)

from sklearn.feature_extraction.text import TfidfVectorizer

vectorization = TfidfVectorizer()

xv_train = vectorization.fit_transform(x_train)
```

```python
xv_test = vectorization.transform(x_test)

from sklearn.linear_model import LogisticRegression

LR = LogisticRegression()

LR.fit(xv_train,y_train)

pred_lr=LR.predict(xv_test)

LR.score(xv_test, y_test)

print(classification_report(y_test, pred_lr))

from sklearn.tree import DecisionTreeClassifier

DT = DecisionTreeClassifier()

DT.fit(xv_train, y_train)

pred_dt = DT.predict(xv_test)

DT.score(xv_test, y_test)

print(classification_report(y_test, pred_dt))

from sklearn.ensemble import GradientBoostingClassifier

GBC = GradientBoostingClassifier(random_state=0)

GBC.fit(xv_train, y_train)

pred_gbc = GBC.predict(xv_test)

GBC.score(xv_test, y_test)

print(classification_report(y_test, pred_gbc))

from sklearn.ensemble import RandomForestClassifier

RFC = RandomForestClassifier(random_state=0)

RFC.fit(xv_train, y_train)

pred_rfc = RFC.predict(xv_test)

RFC.score(xv_test, y_test)

print(classification_report(y_test, pred_rfc))

def output_lable(n):

    if n == 0:

        return "Fake News"
```

```python
    elif n == 1:
        return "Not A Fake News"


def manual_testing(news):
    testing_news = {"text":[news]}
    new_def_test = pd.DataFrame(testing_news)
    new_def_test["text"] = new_def_test["text"].apply(wordopt)
    new_x_test = new_def_test["text"]
    new_xv_test = vectorization.transform(new_x_test)
    pred_LR = LR.predict(new_xv_test)
    pred_DT = DT.predict(new_xv_test)
    pred_GBC = GBC.predict(new_xv_test)
    pred_RFC = RFC.predict(new_xv_test)


    return print("\n\nLR Prediction: {} \nDT Prediction: {} \nGBC Prediction: {}
\nRFC Prediction: {}".format(output_lable(pred_LR[0]),

output_lable(pred_DT[0]),

output_lable(pred_GBC[0]),

output_lable(pred_RFC[0])))
news = str(input())
manual_testing(news)
```

CHAPTER 5

OUTPUT SCREEN
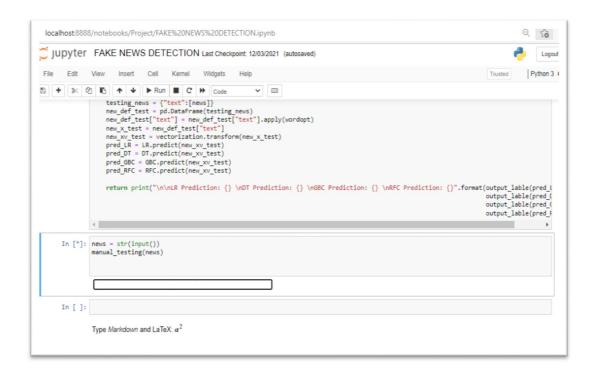
5.1 INPUT STAGE

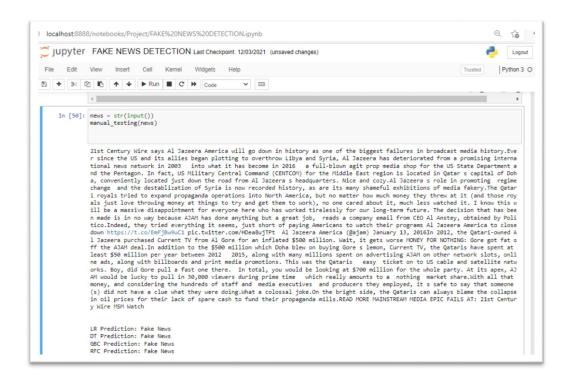Fig 5.1.1 Input Stage

5.2 OUTPUT STAGE

Fig 5.2.1 Output Stage

CHAPTER 6

CONCLUSION

- With the help of Machine Learning we have create 5 prediction model which gives the accuracy above 90% and it cover all lastest political covid 19 news. Also with some pretrained model we have cover news related to history and sport.

- We intent to build our own dataset which will be kept up to data according to the lastest news in furture.

CHAPTER 7

FUTURE SCOPE

This project can be further enhanced to provide greater flexibility and performance with certain modification whenever necessary. Deep fake learning which can be help to detect fake image. Deep learning machine learning to get more accurate result.

Findings from this research can be used for various activities:

- Researchers can further analyze and compare several language generation models with the human writing style and inter-model styles. It will increase their scope and help with better fake news detection.
- Even though TF-IDF classifiers worked well, there are possibilities of exploring other features to improve the model and make it a generic fit.
- While the project focused on text-based news articles and language models, AI algorithms can also analyze other features such as images, videos, date and time, sources, website, and domain for valuable information.
- Teaching the detection model to trace the source of a machine-generated fake news article to the language model from which it originated will be a huge step forward. This development will ensure mitigating and blocking the spread of misinformation.

REFERENCES

- Kai Shu, Amy Sliva, Suhang Wang, Jiliang Tang, and Huan Liu, "Fake News Detection on Social Media: A Data Mining Perspective" arXiv:1708.01967v3 [cs.SI], 3 Sep 2017

- M. Granik and V. Mesyura, "Fake news detection using naive Bayes classifier," 2017 IEEE First Ukraine Conference on Electrical and Computer Engineering (UKRCON), Kiev, 2017, pp. 900-903.

- Fake news websites. (n.d.) Wikipedia. [Online].
  Available: https://en.wikipedia.org/wiki/Fake_news_website. Accessed Feb. 6, 2017

- Cade Metz. (2016, Dec. 16). The bittersweet sweepstakes to build an AI that destroys fake news.

- Conroy, N., Rubin, V. and Chen, Y. (2015). "Automatic deception detection: Methods for finding fake news" at Proceedings of the Association for Information Science and Technology, 52(1), pp.1-4