FAKE NEWS DETECTION

**A DESIGN PROJECT REPORT SUBMITTED
IN PARTIAL FULFILMENT FOR THE AWARD OF
THE DEGREE OF
BACHELOR OF TECHNOLOGY
IN
COMPUTER SCIENCE AND ENGINEERING &
INFORMATION TECHNOLOGY**

Submitted by

IFFA TAQUI JOWHER	18071A0522
KOLASANI AARTI CHOWDARY	18071A0583
KAMMARI KRISHNA VAMSHI	18071A1283
GUNDA SWETHA	18071A12E3
PARAVASTU SATWIK	19075A1218

TAMASOMA JYOTIRGAMAYA
Estd. 1995

**VALLURUPALLI NAGESWARA RAO VIGNANA JYOTHI
INSTITUTE OF ENGINEERING AND TECHNOLOGY**

An Autonomous Institute, NAAC Accredited with 'A++' Grade (CGPA: 3.73/4.0)
NBA Accredited for CE, EEE, ME, ECE, CSE, EIE, IT B.Tech. Programmes
Approved by AICTE, New Delhi, Affiliated to JNTU-H, Recognised as "College with Potential for Excellence" by UGC
Vignana Jyothi Nagar, Pragathi Nagar, Nizampet (S.O), Hyderabad TS 500 090 India

2020-2021

CENTRE FOR PRESENCING AND DESIGN THINKING

VALLURUPALLI NAGESWARA RAO VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

An Autonomous Institute, NAAC Accredited with 'A++' Grade (CGPA: 3.73/4.0)
NBA Accredited for CE, EEE, ME, ECE, CSE, EIE, IT B.Tech. Programmes
Approved by AICTE, New Delhi, Affiliated to JNTU-H, Recognised as "College with Potential for Excellence" by UGC
Vignana Jyothi Nagar, Pragathi Nagar, Nizampet (S.O), Hyderabad TS 500 090 India

TAMASOMA JYOTIRGAMAYA
Estd. 1995

CERTIFICATE

This is to certify that the project titled **"Fake News Detection"** is being submitted, by **Iffa Taqui Jowher(18071A0522)**, **Kolasani Aarti Chowdary(18071A083), Kammari Krishna Vamshi(18071A1283),Gunda Swetha(18071A12E3)** and **Paravastu Satwik (19075A1218),** in partial fulfilment of the requirement for the award of degree of **Bachelor of Technology in Computer Science Engineering**, to the Centre for Presenting and Design Thinking at the **Vallurupalli Nageswara Rao Vignana Jyothi Institute of Engineering and Technology** is a record of *bona fide* work carried out by them under our pedagogy. The results embodied in this Project have not been submitted to any other University or Institute for the award of any degree.

Dr. N. Sandhya
Project Mentor
Member, CPADT
VNRVJIET
Hyderabad

Internal Examiners **External Examiners**

ABSTRACT

Social media for news consumption is a double-edged sword. On one hand, its low cost, easy access and rapid dissemination of information lead people to seek out and consume news from social media. On the other, it enables the wide spread of “fake news”. Recent political events have led to an increase in the popularity and spread of fake news. The extensive spread of fake news has the potential for extremely negative impacts on individuals and society. Therefore, fake news detection on has recently become an emerging research that is attracting tremendous attention. As demonstrated by the widespread effects of the large onset of fake news, humans are inconsistent if not outright poor detectors of fake news. With this, efforts have been made to automate the process of fake news detection. The results of this project demonstrate the ability for machine learning to be useful in this task.

LITERATURE SURVEY

Fake news has been defined and the target has been set, it is now needed to analyze what features can be used in order to classify fake news. Starting by looking at news content, it can be seen that it is made of four principal raw components:

- Source: Where does the news come from, who wrote it, is this source reliable or not.
- Headline: Short summary of the news content that try to attract the reader.
- Body Text: The actual text content of the news.
- Image/Video: Usually, textual information is in agreement with visual information such as images, videos or audio. Features will be extracted from these four basic components, with the mains features being linguistic-based and visual-based. As explained before, fake news is used to influence the consumer, and in order to do that, they often use a specific language in order to attract the readers.

On the other hand, non-fake news will mostly stick to a different language register, being more formal. This is linguistic-based features, to which can be added lexical features such as the total number of words, frequency of large words or unique words.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Objective

Fake news may be a relatively new term but it is not necessarily a new phenomenon. Fake news has technically been around at least since the 19th century. As an increasing amount of our lives is spent interacting online through social media platforms, more and more people tend to seek out and consume news from social media rather than traditional news organizations. As such, the effects of fake news have increased exponentially in the recent past and something must be done to prevent this from continuing in the future..

1.2 Introduction

It is also found that social media now outperforms television as the major news source. Despite the advantages provided by the social media, the quality of news on social media is lower than traditional news organizations. However, because it is cheap to provide news online and much faster and easier to disseminate through social media, large volumes of fake news I.e. those news articles with intentionally false information, are produced online for a variety of purposes, such as financial and political gain. Given the prevalence of this new phenomenon, “Fake news” was even named the word of the year by the Macquarie dictionary in 2016.

1.3 Motivation

To influence public opinion. Fake news on social media has been occurring for several years; however, there is no agreed upon definition of the term “fake news”. To better guide the future directions of fake news detection research, appropriate clarifications are necessary. Social media has proved to be a powerful source for fake news dissemination. There are some emerging patterns that can be utilized for fake news detection in social media. A review on existing fake news detection methods under various social media scenarios can provide a basic understanding on the sate-of-the-art fake news detection methods.

1.4 Scope for the Work

This type of solution is not intended to be an end-to-end solution for fake news classification. There are cases in which it fails and some for it succeeds. Instead of being an end-to-end solution, this project is intended to be one tool that could be used to aid humans who are trying to classify fake news. Alternatively, it could be one tool used in future applications that intelligently combine multiple tools to create an end-to-end solution to automating the process of fake news classification.

CHAPTER 2

DISCOVER AND DEFINE

2.1 Empathy

- Fake news and disinformation can covertly modify the behavior of individuals. It can do this by manipulating implicit attitudes and emotions.
- Current mitigation methods do not prevent behavior modification. It is urgent to address this threat to democracy and individual autonomy.
- The fact that if a lie is repeated enough times, you'll begin to believe it's true.
- Though people's initial belief in false information can be difficult to change, some evidence suggests that warnings about false information can reduce belief in false claims or prevent the uptake of misinformation.

2.2 Empathy Tool

2.2.1 Empathy map

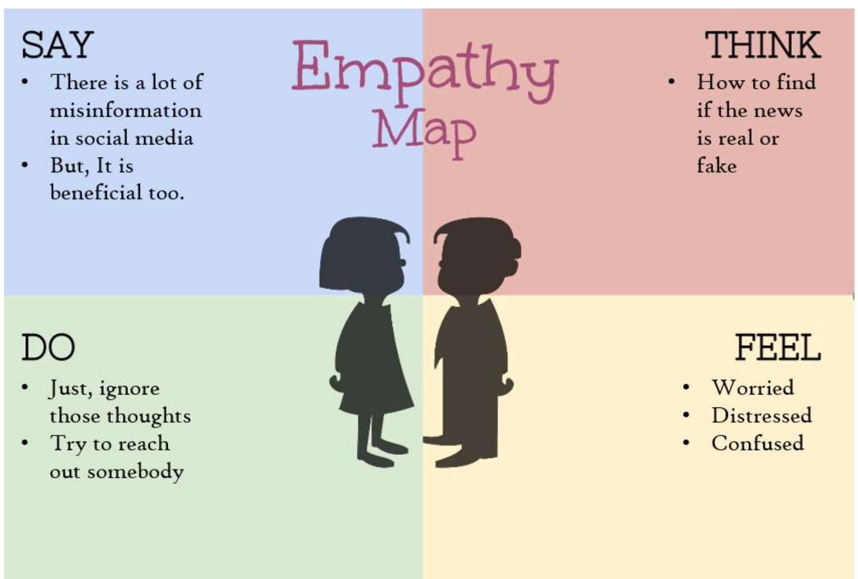

Figure 2.2: Empathy map

- Purposely misleading a story is 'fake news'. Now a days, fake news is a major problem in the society.
- People started losing faith in news on social media. As most of our lives are spent interacting online through social media platforms, more and more people tend to seek

news from social media rather than news organizations or newspapers and further it is easy to share and discuss the news with friends or other readers on social media.

- Though there are many advantages provided by social media the quality of news on social media is lower when compared to traditional news organization.
- The extensively spread of fake news can have a serious negative impact on individual and society
- After analyzing the survey results, we understand that the people are concerned and worried by fake news but are really not trying to verify whether the news is real or fake.
- And most of them believed that it is social media which is playing a key role in this phenomenon

A Survey has been conducted for the students.

Google Form: https://forms.gle/DEz3bQPc2tnBofVz6

2.2.2 Survey

- What are students news reading habits like?
- What are some of the news resources/platforms most commonly utilized by students?
- How confident are students in identifying fake news?
- Can they actually identify fake news from the real?

2.2.3 Interview

- What are students views on fake news and its impact?
- How do high students actually read the news?
- What kind of news literacy skills do they utilize and lack of?

2.2.4 Questionnaire:

- What are some of the existing solutions out there? (to avoid repeating similar designs)
- Are existing solutions good enough to solve the problem?
- What are some of the strengths and weaknesses of the existing solutions?

2.3 User Needs

User needs are requirements that add value to a product, service or environment for a user. Capturing user needs is a process of engaging users to understand their problems, processes, goals and preferences.

- With the data we obtained through the research phase, we first analyzed the current **characteristics** of our target user group to identify what are some of the opportunities we have with the user group. The 4 main characteristics of students are summarized as the following:

    1. Social media as main source of news
    2. Snacking on news instead of reading
    3. Limited news literacy skills
    4. High level of self-confidence.

2.3.1 Primary Needs

- Finding the spam websites
- Misleading content
- Fabricated content
- False context
- Imposter content

2.3.2 Secondary Needs

- Helps users to spam filters on system.
- Will be able to get information about clicks which they have clicked.

2.3.3 Latent Needs

We need users to be smart enough to understand what is spam and what is not . Also we need them not to blindly believe what is sent over internet.

CHAPTER 3
CUSTOMER SERVICE EXPERIENCE

3.1 Task Flow

The task flows is tend to be cyclic, showing the high-level steps that a person would take to get to a specific goal or end point. Task flows tend not to branch out with options or decision points, tend to be cyclic and sequential, and are generally meant to be simple iterative, rather than complex.

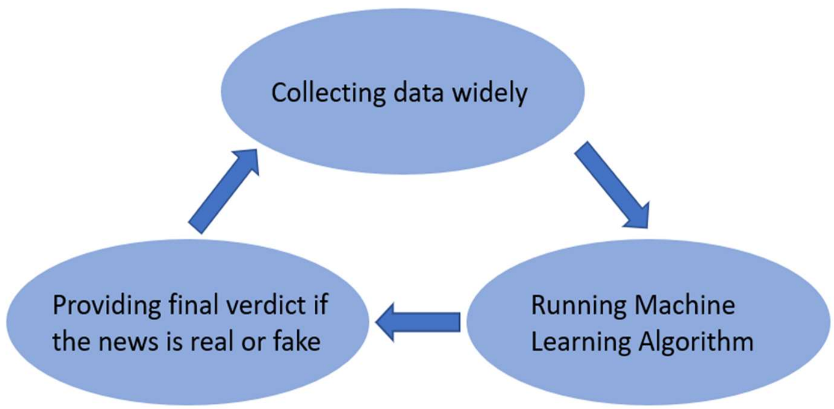

Figure 3.1: Service Experience Cycle

3.2 Pain Points

- Misinformation can lead to something which is really dangerous sometimes.
- They don’t have time to verify the news articles.

3.3 Gain Points

- This platform can get clarification to them.
- It will cut off the time spent on reaching out to somebody else.
- Real information is always worthy.

CHAPTER 4

IDEATION

4.1 Ideation Tool- Mind map

Before we started a design sprint, we all sat down as a team to discuss design considerations we need to be aware of as we ideate. A main goal identified in the research phase of the project is to help users to progress from simply aware of some basic news literacy skills to successfully applying the skills when they read news on different platforms in their daily lives. In order to achieve the goal, tools that only provide passive detection based on keywords from articles would not satisfy our users' needs. We also need to ensure users are learning skills and applying them in context according their level. Therefore, Mind mapping is used to represent how ideas or other items are linked to a central idea and to each other. Mind maps are used to generate, visualize, structure and classify ideas to look for patterns and insights that provide key design criteria.

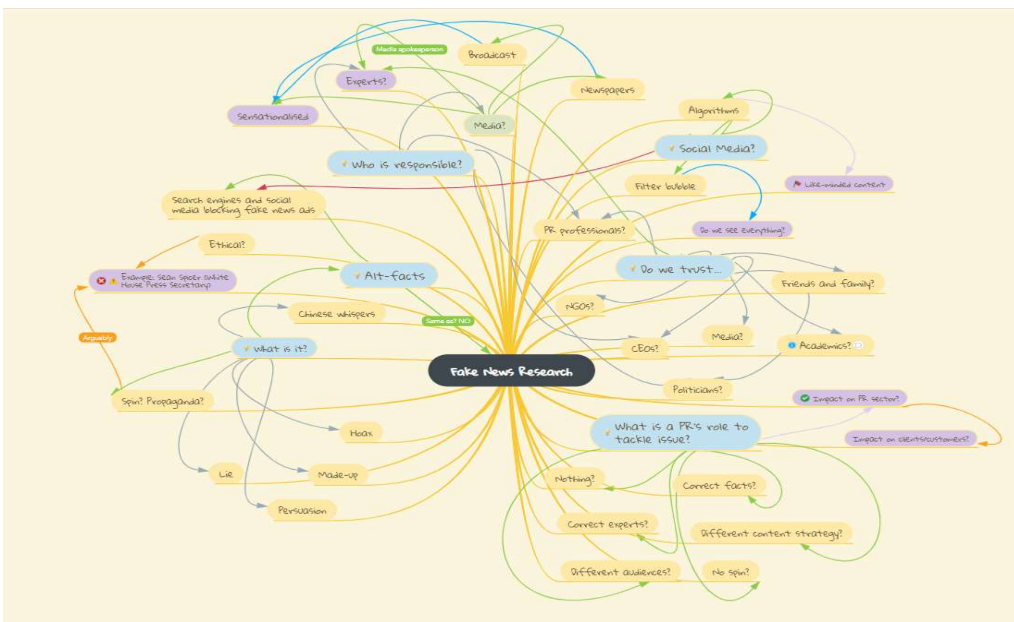

Figure 4.1: Mind map

4.2 Ideation Tool- Brainstorming

- After understanding the problems and brainstorming we have discussed about
    - What is it?
    - Who is responsible?
    - Do we trust?
- Finally, came up with a solution to use Machine Learning algorithms to classify real and fake news
- These machine learning algorithms can be run against like-minded content

CHAPTER 5

PROTOTYPE MODEL

5.1 Solution Prototype

- The solution gives us if the news is real or fake after running it on five algorithms.
- Algorithms used are
    1. Logistic Regression
    2. Decision Tree Classifier
    3. Gradient Boosting Classifier
    4. Random Forest Classifier
    5. Linear Super Vector Classifier
- Colab Notebook: https://colab.research.google.com/drive/1RAiWivFbFT0Ouj85l8C7Nbx5PV8VsLkv

5.2 Code:

5.2.1 Import library

```
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
import matplotlib.pyplot as plt
from collections import Counter
#from sklearn import naive_bayes, metrics, svm
from IPython.display import Image
import warnings
warnings.filterwarnings("ignore")
%matplotlib inline
import seaborn as sns
from sklearn.metrics import classification_report
import re
import string
```

Figure 5.2.1: Import library Code

5.2.2 Inserting dataset

```
df = pd.read_csv('news.csv')
X=df['text']
y=df['label']
```

Figure 5.2.2: Inserting dataset Code

5.2.3 Data visualization

```
count_Class=pd.value_counts(df["label"], sort= True)
count_Class.plot(kind= 'bar', color= ["blue", "orange"])
plt.title('Bar chart')
plt.show()
```

Figure 5.2.3: Data visualization Code -1

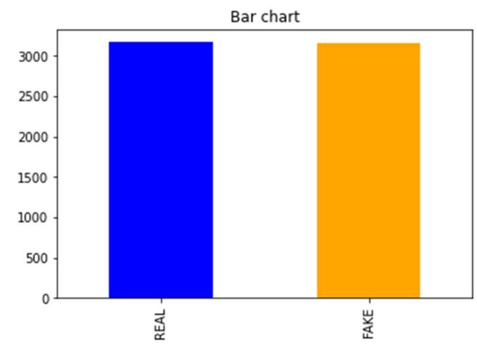

Figure 5.2.3: Data visualization Bar Chart-1

```
count_Class.plot(kind = 'pie',  autopct='%1.0f%%')
plt.title('Pie chart')
plt.ylabel('')
plt.show()
```

Figure 5.2.3: Data visualization Code -2

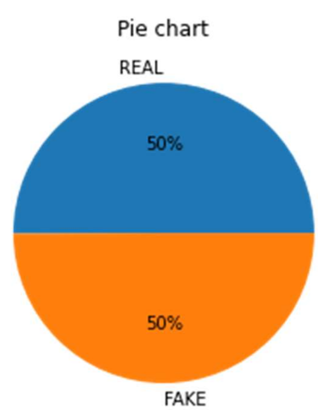

Figure 5.2.3: Data visualization Bar Chart-2

```
count1 = Counter(" ".join(df[df['label']=='REAL']["text"]).split()).most_common(20)
df1 = pd.DataFrame.from_dict(count1)
df1 = df1.rename(columns={0: "words in REAL", 1 : "count"})
count2 = Counter(" ".join(df[df['label']=='FAKE']["text"]).split()).most_common(20)
df2 = pd.DataFrame.from_dict(count2)
df2 = df2.rename(columns={0: "words in FAKE", 1 : "count_"})
```

```
df1.plot.bar(legend = False)
y_pos = np.arange(len(df1["words in REAL"]))
plt.xticks(y_pos, df1["words in REAL"])
plt.title('More frequent words in non-spam messages')
plt.xlabel('words')
plt.ylabel('number')
plt.show()
```

Figure 5.2.3: Data visualization Code -3

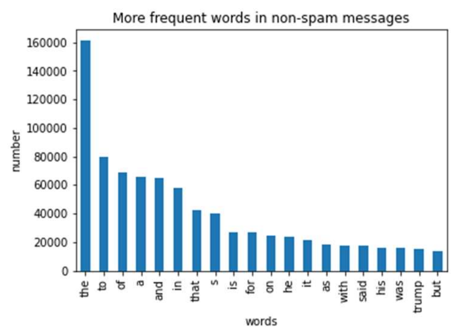

Figure 5.2.3: Data visualization Bar Chart-3

```
df2.plot.bar(legend = False, color = 'orange')
y_pos = np.arange(len(df2["words in FAKE"]))
plt.xticks(y_pos, df2["words in FAKE"])
plt.title('More frequent words in spam messages')
plt.xlabel('words')
plt.ylabel('number')
plt.show()
```

Figure 5.2.3: Data visualization Code-4

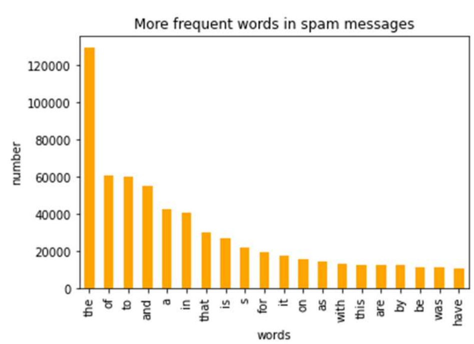

Figure 5.2.3: Data visualization Bar Chart-4

5.2.4 Data preprocessing

```python
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
```

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

```python
x = df["text"]
y = df["label"]
```

Figure 5.2.4: Data Preprocessing Code

5.2.5 Data splitting

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

Figure 5.2.5: Data Splitting Code

5.2.6 Convert text to vectors

```python
from sklearn.feature_extraction.text import TfidfVectorizer
```

```python
vectorization = TfidfVectorizer()
xv_train = vectorization.fit_transform(x_train)
xv_test = vectorization.transform(x_test)
```

Figure 5.2.6 Convert text to vectors Code

5.2.7 Logistic Regression

```python
from sklearn.linear_model import LogisticRegression
```

```python
LR = LogisticRegression()
LR.fit(xv_train,y_train)
```

```
LogisticRegression(C=1.0, class_weight=None, dual=False, fit_intercept=True,
                   intercept_scaling=1, l1_ratio=None, max_iter=100,
                   multi_class='auto', n_jobs=None, penalty='l2',
                   random_state=None, solver='lbfgs', tol=0.0001, verbose=0,
                   warm_start=False)
```

```python
pred_lr=LR.predict(xv_test)
```

```python
LR.score(xv_test, y_test)
```

```
0.913510101010101
```

```python
print(classification_report(y_test, pred_lr))
```

```
              precision    recall  f1-score   support

        FAKE       0.90      0.93      0.92       798
        REAL       0.93      0.89      0.91       786

    accuracy                           0.91      1584
   macro avg       0.91      0.91      0.91      1584
weighted avg       0.91      0.91      0.91      1584
```

Figure 5.2.7 Logistic Regression Code

5.2.8 Decision Tree Classifier

```
from sklearn.tree import DecisionTreeClassifier
```

```
DT = DecisionTreeClassifier()
DT.fit(xv_train, y_train)
```

```
DecisionTreeClassifier(ccp_alpha=0.0, class_weight=None, criterion='gini',
                       max_depth=None, max_features=None, max_leaf_nodes=None,
                       min_impurity_decrease=0.0, min_impurity_split=None,
                       min_samples_leaf=1, min_samples_split=2,
                       min_weight_fraction_leaf=0.0, presort='deprecated',
                       random_state=None, splitter='best')
```

```
pred_dt = DT.predict(xv_test)
```

```
DT.score(xv_test, y_test)
```

```
0.8055555555555556
```

```
print(classification_report(y_test, pred_dt))
```

```
              precision    recall  f1-score   support

        FAKE       0.81      0.81      0.81       798
        REAL       0.81      0.80      0.80       786

    accuracy                           0.81      1584
   macro avg       0.81      0.81      0.81      1584
weighted avg       0.81      0.81      0.81      1584
```

Figure 5.2.8 Decision Tree Classifier Code

5.2.9 Gradient Boosting Classifier

```
from sklearn.ensemble import GradientBoostingClassifier
```

```
GBC = GradientBoostingClassifier(random_state=0)
GBC.fit(xv_train, y_train)
```

```
GradientBoostingClassifier(ccp_alpha=0.0, criterion='friedman_mse', init=None,
                           learning_rate=0.1, loss='deviance', max_depth=3,
                           max_features=None, max_leaf_nodes=None,
                           min_impurity_decrease=0.0, min_impurity_split=None,
                           min_samples_leaf=1, min_samples_split=2,
                           min_weight_fraction_leaf=0.0, n_estimators=100,
                           n_iter_no_change=None, presort='deprecated',
                           random_state=0, subsample=1.0, tol=0.0001,
                           validation_fraction=0.1, verbose=0,
                           warm_start=False)
```

```
pred_gbc = GBC.predict(xv_test)
```

```
GBC.score(xv_test, y_test)
```

```
0.8977272727272727
```

```
print(classification_report(y_test, pred_gbc))
```

```
              precision    recall  f1-score   support

        FAKE       0.89      0.91      0.90       798
        REAL       0.91      0.88      0.90       786

    accuracy                           0.90      1584
   macro avg       0.90      0.90      0.90      1584
weighted avg       0.90      0.90      0.90      1584
```

Figure 5.2.9 Gradient Boosting Classifier Code

5.2.10 Random Forest Classifier

```
from sklearn.ensemble import RandomForestClassifier
```

```
RFC = RandomForestClassifier(random_state=0)
RFC.fit(xv_train, y_train)
```

```
RandomForestClassifier(bootstrap=True, ccp_alpha=0.0, class_weight=None,
                       criterion='gini', max_depth=None, max_features='auto',
                       max_leaf_nodes=None, max_samples=None,
                       min_impurity_decrease=0.0, min_impurity_split=None,
                       min_samples_leaf=1, min_samples_split=2,
                       min_weight_fraction_leaf=0.0, n_estimators=100,
                       n_jobs=None, oob_score=False, random_state=0, verbose=0,
                       warm_start=False)
```

```
pred_rfc = RFC.predict(xv_test)
```

```
RFC.score(xv_test, y_test)
```

```
0.8800505050505051
```

```
print(classification_report(y_test, pred_rfc))
```

```
              precision    recall  f1-score   support

        FAKE       0.87      0.90      0.88       798
        REAL       0.89      0.86      0.88       786

    accuracy                           0.88      1584
   macro avg       0.88      0.88      0.88      1584
weighted avg       0.88      0.88      0.88      1584
```

Figure 5.2.10 Random Forest Classifier Code

5.2.11 Linear Super Vector Classifier

```
from sklearn.svm import LinearSVC
```

```
SVC = LinearSVC()
SVC.fit(xv_train, y_train)
```

```
LinearSVC(C=1.0, class_weight=None, dual=True, fit_intercept=True,
          intercept_scaling=1, loss='squared_hinge', max_iter=1000,
          multi_class='ovr', penalty='l2', random_state=None, tol=0.0001,
          verbose=0)
```

```
pred_SVC = SVC.predict(xv_test)
```

```
SVC.score(xv_test, y_test)
```

```
0.9393939393939394
```

```
print(classification_report(y_test, pred_SVC))
```

```
              precision    recall  f1-score   support

        FAKE       0.94      0.94      0.94       798
        REAL       0.94      0.93      0.94       786

    accuracy                           0.94      1584
   macro avg       0.94      0.94      0.94      1584
weighted avg       0.94      0.94      0.94      1584
```

Figure 5.2.11 Linear Super Vector Classifier Code

5.2.12 Model testing with manual Entry

```
def output_lable(n):
    if n == 'FAKE':
        return "Fake News"
    elif n == 'REAL':
        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)
    pred_SVC = SVC.predict(new_xv_test)

    return print("\n\nLR Prediction: {} \nDT Prediction: {} \nGBC Prediction: {} \nRFC Prediction: {} \nSVC Prediction: {}".format(output_lable(pred_LR[0]),
                                                                                                              output_lable(pred_DT[0]),
                                                                                                              output_lable(pred_GBC[0]),
                                                                                                              output_lable(pred_RFC[0]),
                                                                                                              output_lable(pred_SVC[0])))
```

Figure: 5.2.12 Model testing with manual Entry Code

5.2.13 Input / Output

```
news = str(input())
manual_testing(news)
```

```
your email was selected to claim to sum of  $ 5,000,000 in the 2011 european lottery


LR Prediction: Fake News
DT Prediction: Fake News
GBC Prediction: Fake News
RFC Prediction: Fake News
SVC Prediction: Fake News
```

Figure: 5.2.13 Input / Output Code

5.3 Real Win Worth

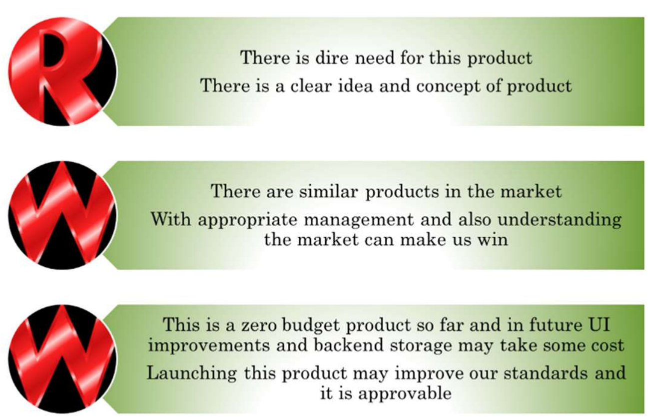

Figure: 5.3 Real Win Worth

CHAPTER 6
CONCLUSIONS AND FUTURE SCOPE

6.1 Conclusions

Social media has been used to spread fake news, which has strong negative impacts on individual users and broader society. The main contribution of this project is support for the idea that machine learning could be useful in a novel way for the task of classifying fake news. Many patterns are intuitively useful in a humans manner of classifying fake news. This application could be a tool for humans trying to classify fake news, to get indications of which words might cut them into the correct classification. It could also be useful I researchers trying to develop improved models through the use of improved and enlarged datasets, different parameters, etc.

6.2 Future Scope

Through this work done in this project, we have shown that machine learning certainly does have the capacity to pick up on sometimes subtle language patterns that may be difficult for humans to pick up. An aspect in which this project could be expanded is by comparing it to humans performing the same task. Comparing the accuracies would be beneficial in deciding whether or not the dataset is representative of how difficult the task of separating fake from real news is. If humans are more accurate than the model, it may mean that we need to choose more deceptive fake news examples. As we have mentioned throughout, this application is only one that would be necessary in a larger toolbox that could function as a highly accurate fake-news classifier. In order to combine all tools, there would need to be some type of model that combines all the tools and learns how to weight each of them in its final decision.

We can also develop User Interface in future and use it widely to verify news of different subjects

References

- https://www.hindawi.com/journals/complexity/2020/8885861/
- https://data-flair.training/blogs/advanced-python-project-detecting-fake-news/
- https://ieeexplore.ieee.org/abstract/document/8862770
- https://www.interaction-design.org/literature/topics/design-thinking#:~:text=Design%20thinking%20is%20a%20non,are%20ill%2Ddefined%20or%20unknown.

Appendix A: User Surveys

A.1 Questionnaire for Users

- Have you ever been a victim of "fake news".
- If yes, how many times and from which medium?
- From which medium do you usually get informed ?
- Do you take any measures to verify the news you consume?
- If yes, please specify the measures.
- Do you think that it is important to fight "fake news"?
- Do you think that on the internet and/or social media the chances for people to fall for "fake news" are higher?
- Has "fake news" affected your trust towards the news organisations?
- If yes, in which way? (please specify)
- Who do you think should be responsible for identifying "fake news" on social media? (please select all applicable answers)

A.2 Text Transcripts of User Responses

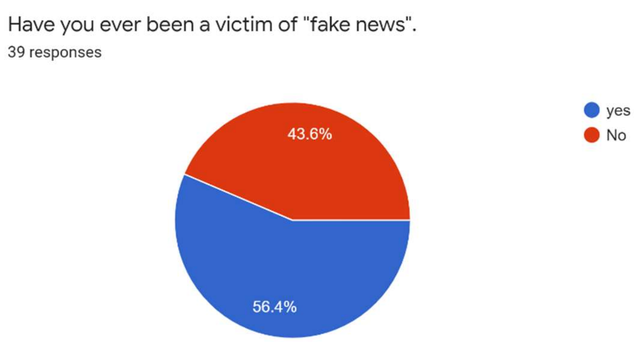

Figure A.2: Survey result -1

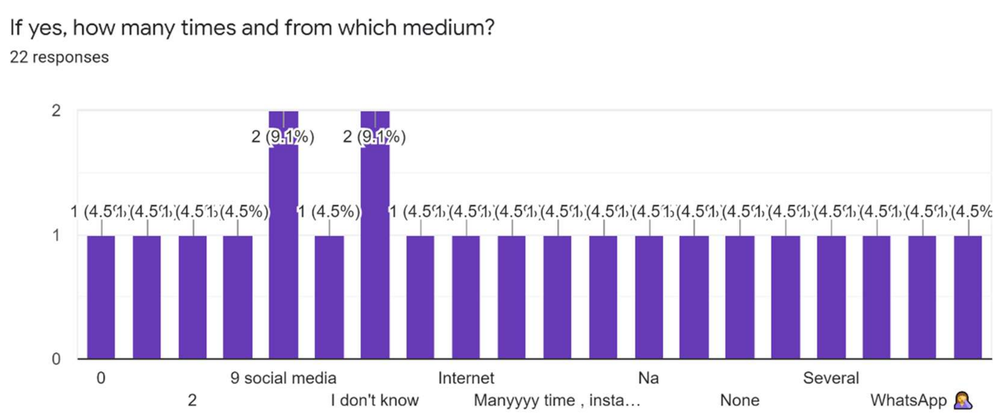

Figure A.2: Survey result -2

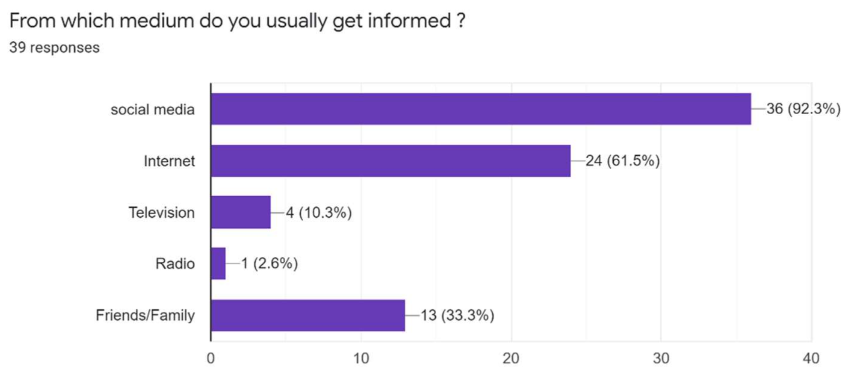

Figure A.2: Survey result -3

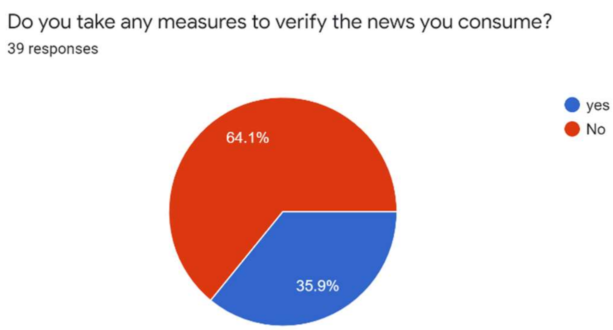

Figure A.2: Survey result -4

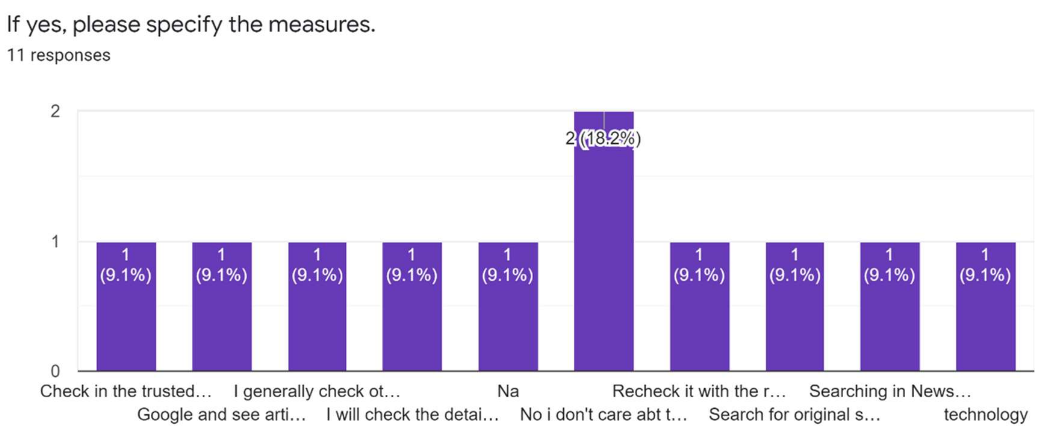

Figure A.2: Survey result -5

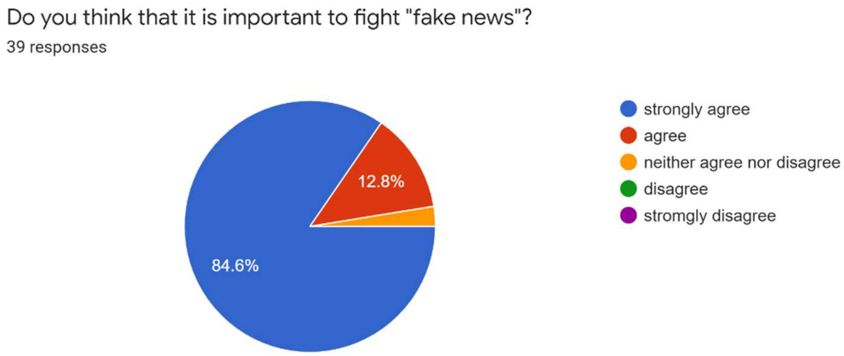

Figure A.2: Survey result -6

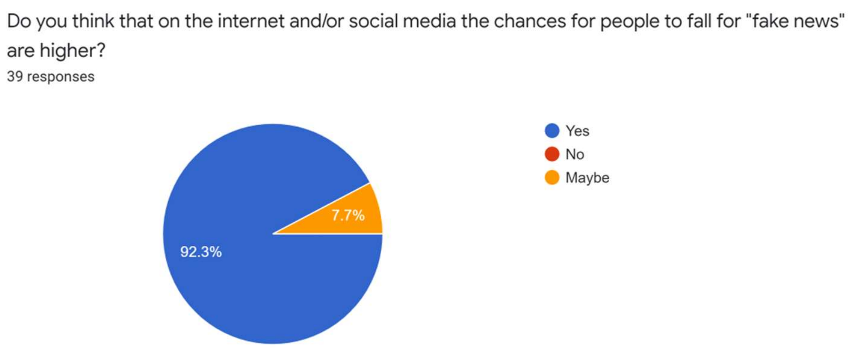

Figure A.2: Survey result -7

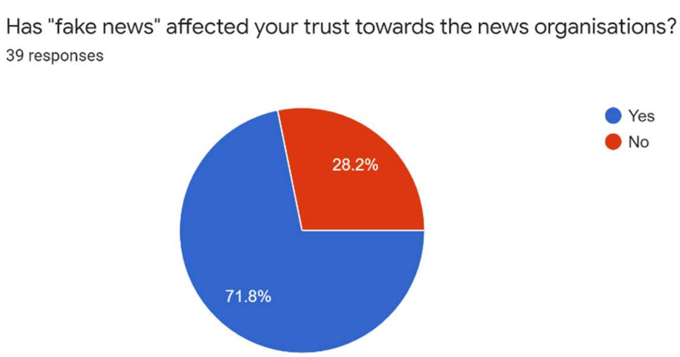

Figure A.2: Survey result -8

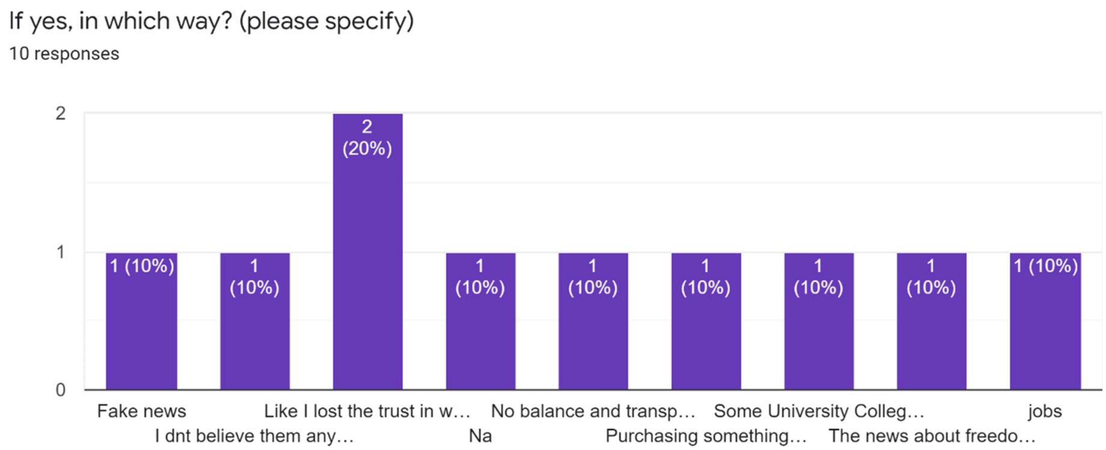

Figure A.2: Survey result -9

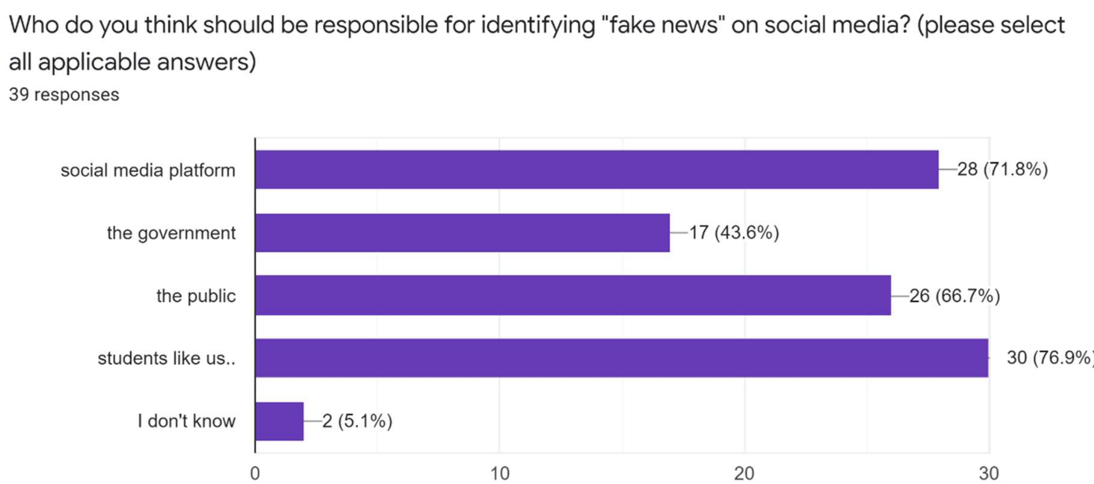

Figure A.2: Survey result -10