

TABLE OF CONTENT

CHAPTER NO	TITLE	PAGE NO
	ABSTRACT	
	LIST OF FIGURES	
I.	INTRODUCTION	
	1.1 MOTIVATION	
	1.2 OBJECTIVES	
	1.3 OVERVIEW OF THE PROJECT	
II	LITERATURE OF SURVEY	
	2.1 MEDIA RICH FAKE NEWS DETECTION	
	2.1.1 WEAKLY SUPERVISED LEARNING FOR FAKE NEWS	
	2.2 FAKE NEWS DETECTION IN SOCIAL MEDIA	
	2.3 THE SPREA OF FAKE NEWS BY SOCIAL BOTS	
	2.4 MISLEADING ONLINE CONTENT	
III	METHODOLOGY	
	3.1 EXISTING SYSTEM	
	3.2 PROPOSED SYSTEM	
	3.3 SOFTWARE ENVIRONMENT	
	3.3.1 PYTHON	
	3.3.2 PYTHON FEATURES	
	3.3.3 INTERACTIVE MODE PROGRAMMING	
	3.3.4 SCRIPT MODE PROGRAMMING	
	3.4 FLASK FRAMEWORK	
	3.5 MODULES	
	3.6 ALGORITHMS	
IV	RESULTS AND DISCUSSION	
	4.1 REQUIREMENT ANALYSIS	
	4.2 FUNCTIONAL REQUIREMENTS	
	4.3 NON FUNCTIONAL REQUIREMENTS	
	4.4 SYSTEM DESIGN AND TESTING PLAN	
	4.5 DATA FLOW DIAGRAM	
	4.6 TEST PROCESS	
V	CONCLUSION AND FUTURE WORK	
	REFERENCES	
	A. SOURCE CODE	
	B. SCREENSHOTS	

ABSTRACT

This paper investigates the intricate process of detecting fake news articles, creators, and subjects within online social networks. Recognizing the imperative of fostering trustworthiness in online information dissemination, our study aims to unveil the underlying principles, methodologies, and algorithms essential for timely identification of deceptive content. The vast expanse of web-scale data presents formidable challenges in discerning, evaluating, and rectifying misinformation, commonly termed as "fake news." In response, our proposed method focuses on detecting fake news specifically within the context of Facebook, one of the most prominent social media platforms. Leveraging a Naive Bayes classification model, we outline a systematic approach to predict whether a post on Facebook is genuine or fabricated. Furthermore, we explore various techniques aimed at refining the efficacy of our detection method. These techniques encompass a spectrum of strategies, including feature engineering, ensemble learning, and sentiment analysis, among others. By integrating these methodologies, we aim to enhance the accuracy and reliability of our fake news detection system. Our research findings underscore the feasibility of employing machine learning methods to address the pervasive issue of fake news. By elucidating the intricacies of detection mechanisms and evaluating their performance, we contribute to the ongoing discourse on combating misinformation and fostering trust in online information dissemination.

CHAPTER 1

INTRODUCTION

These days“ fake news is creating different issues from sarcastic articles to a fabricated news and plan government propaganda in some outlets. Fake news and lack of trust in the media are growing problems with huge ramifications in our society. Obviously, a purposely misleading story is “fake news “ but lately blathering social media”s discourse is changing its definition. Some of them now use the term to dismiss the facts counter to their preferred viewpoints. The importance of disinformation within American political discourse was the subject of weighty attention , particularly following the American president election . The term 'fake news' became common parlance for the issue, particularly to describe factually incorrect and misleading articles published mostly for the purpose of making money through page views. In this paper,it is sought to produce a model that can accurately predict the likelihood that a given article is fake news.Facebook has been at the epicenter of much critique following media attention. They have already implemented a feature to flag fake news on the site when a user sees“s it ; they have also said publicly they are working on to to distinguish these articles in an automated way. Certainly, it is not an easy task. A given algorithm must be politically unbiased — since fake news exists on both ends of the spectrum — and also give equal balance to legitimate news sources on either end of the spectrum. In addition, the question of legitimacy is a difficult one.However, in order to solve this problem, it is necessary to have an understanding on what Fake News. In this paper, we endeavor to contribute to the discourse on fake news detection by proposing a model capable of accurately predicting the likelihood of an article being fake news. Drawing upon insights from interdisciplinary research, we aim to develop a nuanced understanding of fake news and its detection, informed by empirical evidence and theoretical frameworks. Through rigorous analysis and experimentation, we seek to elucidate effective strategies for combatting the spread of misinformation in online social networks.

OBJECTIVE

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

OVERVIEW OF PROJECT

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

CHAPTER 2

LITERATURE OF SURVEY

The available literature has described many automatic detection techniques of fake news and deception posts. Since there are multidimensional aspects of fake news detection ranging from using chatbots for spread of misinformation to use of clickbaits for the rumor spreading . There are many clickbaits available in social media networks including facebook which enhance sharing and liking Proceedings of posts which in turn spreads falsified information. Lot of work has been done to detect falsified information.

MEDIA RICH FAKE NEWS DETECTION: A SURVEY

In general, the goal is profiting through clickbaits. Clickbaits lure users and entice curiosity with flashy headlines or designs to click links to increase advertisements revenues. This exposition analyzes the prevalence of fake news in light of the advances in communication made possible by the emergence of social networking sites. The purpose of the work is to come up with a solution that can be utilized by users to detect and filter out sites containing false and misleading information. We use simple and carefully selected features of the title and post to accurately identify fake posts. The experimental results show a 99.4% accuracy using logistic classifier.

WEAKLY SUPERVISED LEARNING FOR FAKE NEWS DETECTION ON TWITTER

The problem of automatic detection of fake news in social media, e.g., on Twitter, has recently drawn some attention. Although, from a technical perspective, it can be regarded as a straight-forward, binary classification problem, the major challenge is the collection of large enough training corpora, since manual annotation of tweets as fake or non-fake news is an expensive and tedious endeavor. In this paper, we discuss a weakly supervised approach, which automatically collects a large-scale, but very noisy training dataset comprising hundreds of thousands of tweets. During collection, we automatically label tweets by their source, i.e., trustworthy or untrustworthy source, and train a classifier on this dataset. We then use that classifier for a different

classification target, i.e., the classification of fake and non-fake tweets. Although the labels are not accurate according to the new classification target (not all tweets by an untrustworthy source need to be fake news, and vice versa), we show that despite this unclean inaccurate dataset, it is possible to detect fake news with an F1 score of up to 0.9.

FAKE NEWS DETECTION IN SOCIAL MEDIA

Fake news and hoaxes have been there since before the advent of the Internet. The widely accepted definition of Internet fake news is: “fictitious articles deliberately fabricated to deceive readers”. Social media and news outlets publish fake news to increase readership or as part of psychological warfare. In general, the goal is profiting through clickbaits. Clickbaits lure users and entice curiosity with flashy headlines or designs to click links to increase advertisements revenues. This exposition analyzes the prevalence of fake news in light of the advances in communication made possible by the emergence of social networking sites. The purpose of the work is to come up with a solution that can be utilized by users to detect and filter out sites containing false and misleading information. We use simple and carefully selected features of the title and post to accurately identify fake posts. The experimental results show a 99.4% accuracy using logistic classifier.

Automatic Online Fake News Detection Combining Content and Social Signals

The proliferation and rapid diffusion of fake news on the Internet highlight the need of automatic hoax detection systems. In the context of social networks, machine learning (ML) methods can be used for this purpose. Fake news detection strategies are traditionally either based on content analysis (i.e. analyzing the content of the news) or - more recently - on social context models, such as mapping the news’ diffusion pattern. In this paper, we first propose a novel ML fake news detection method which, by combining news content and social context features, outperforms existing methods in the literature, increasing their already high accuracy by up to 4.8%. Second, we implement our method within a Facebook Messenger chatbot and validate it with a real-world application, obtaining a fake news detection accuracy of 81.7%. In recent years, the reliability of information on the Internet has emerged as a crucial issue of modern society. Social network sites (SNSs) have revolutionized the

way in which information is spread by allowing users to freely share content. As a consequence, SNSs are also increasingly used as vectors for the diffusion of misinformation and hoaxes. The amount of disseminated information and the rapidity of its diffusion make it practically impossible to assess reliability in a timely manner, highlighting the need for automatic hoax detection systems. As a contribution towards this objective, we show that Facebook posts can be classified with high accuracy as hoaxes or non-hoaxes on the basis of the users who "liked" them. We present two classification techniques, one based on logistic regression, the other on a novel adaptation of boolean crowdsourcing algorithms. On a dataset consisting of 15,500 Facebook posts and 909,236 users, we obtain classification accuracies exceeding 99% even when the training set contains less than 1% of the posts. We further show that our techniques are robust: they work even when we restrict our attention to the users who like both hoax and non-hoax posts. These results suggest that mapping the diffusion pattern of information can be a useful component of automatic hoax detection systems.

THE SPREAD OF FAKE NEWS BY SOCIAL BOTS

The massive spread of fake news has been identified as a major global risk and has been alleged to influence elections and threaten democracies. Communication, cognitive, social, and computer scientists are engaged in efforts to study the complex causes for the viral diffusion of digital misinformation and to develop solutions, while search and social media platforms are beginning to deploy countermeasures. However, to date, these efforts have been mainly informed by anecdotal evidence rather than systematic data. Here we analyze 14 million messages spreading 400 thousand claims on Twitter during and following the 2016 U.S. presidential campaign and election. We find evidence that social bots play a key role in the spread of fake news. Accounts that actively spread misinformation are significantly more likely to be bots. Automated accounts are particularly active in the early spreading phases of viral claims, and tend to target influential users. Humans are vulnerable to this manipulation, retweeting bots who post false news. Successful sources of false and biased claims are heavily supported by social bots. These results suggests that curbing social bots may be an effective strategy for mitigating the spread of online misinformation.

MISLEADING ONLINE CONTENT

Misleading online content presents a pervasive challenge in today's digital landscape, with far-reaching consequences for individuals, communities, and societies at large. From fabricated news stories to manipulated images and deceptive advertisements, the spectrum of misleading content is vast and varied. This phenomenon not only erodes trust in online information sources but also undermines the integrity of digital platforms and exacerbates societal divisions. One of the most concerning aspects of misleading online content is its potential to spread rapidly and widely, often fueled by social media algorithms and echo chambers. False narratives can gain traction quickly, reaching audiences far beyond their initial dissemination point. Moreover, the anonymity and ease of creating and sharing content online contribute to the proliferation of misleading information, making it challenging for users to discern fact from fiction. Misleading online content can have serious real-world consequences, ranging from influencing public opinion and swaying elections to promoting harmful

behaviors and inciting violence. In some cases, malicious actors intentionally disseminate false information to manipulate public discourse or advance their agendas, exploiting vulnerabilities in digital platforms and exploiting societal divisions. Addressing the issue of misleading online content requires a multifaceted approach that encompasses technological solutions, regulatory measures, media literacy initiatives, and responsible platform governance. Technology companies must invest in robust content moderation systems and algorithmic transparency to identify and mitigate the spread of false information. Policymakers need to enact regulations that hold platforms accountable for the dissemination of misleading content while safeguarding freedom of expression. Additionally, education efforts aimed at improving digital literacy and critical thinking skills are crucial to empower users to navigate the online landscape more effectively. Ultimately, combating misleading online content requires collaboration and concerted action from all stakeholders, including technology companies, policymakers, educators, and users themselves. Through collective effort and responsible citizenship, we can mitigate the harmful effects of misleading content and foster a more trustworthy and informed online environment.

CHAPTER 3

METHODOLOGY

EXISTING SYSTEM

There exists a large body of research on the topic of machine learning methods for deception detection, most of it has been focusing on classifying online reviews and publicly available social media posts. Particularly since late 2016 during the American Presidential election, the question of determining 'fake news' has also been the subject of particular attention within the literature. Conroy, Rubin, and Chen outlines several approaches that seem promising towards the aim of perfectly classify the misleading articles. They note that simple content-related n-grams and shallow parts-of-speech tagging have proven insufficient for the classification task, often failing to account for important context information. Rather, these methods have been shown useful only in tandem with more complex methods of analysis. Deep Syntax analysis using Probabilistic Context Free Grammars have been shown to be particularly valuable in combination with n-gram methods. Feng, Banerjee, and Choi are able to achieve 85%-91% accuracy in deception related classification tasks using online review corpora.

PROPOSED SYSTEM

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

SYSTEM ARCHITECTURE

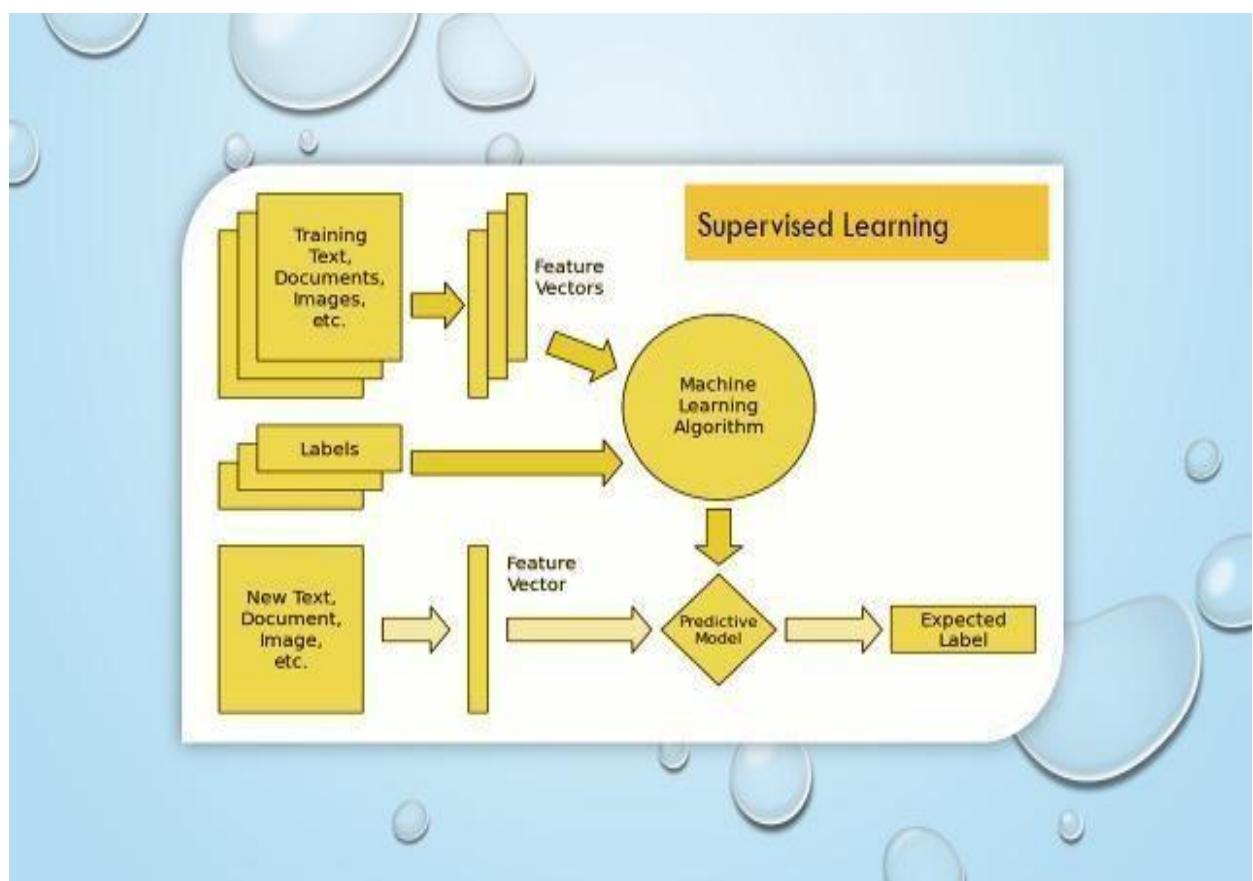


Fig:3.1 Architecture diagram

SYSTEM REQUIREMENTS

HARDWARE REQUIREMENTS:

- ✧ processor - M1 Chip
- ✧ Speed - 3.2GHz
- ✧ Hard disk - 256GB
- ✧ Monitor - Built - in Retina Display
- ✧ RAM - 8 GB

SOFTWARE REQUIREMENTS:

- ✧ Operating System - Macos ventura
- ✧ Coding language - PYTHON

SOFTWARE ENVIRONMENT

PYTHON

Python is a high-level, interpreted, interactive and object-oriented scripting language. Python is designed to be highly readable. It uses English keywords frequently where as other languages use punctuation, and it has fewer syntactical constructions than other languages.

- ✧ **Python is Interpreted** – Python is processed at runtime by the interpreter. You do not need to compile your program before executing it. This is similar to PERL and PHP.
- ✧ **Python is Interactive** – You can actually sit at a Python prompt and interact with the interpreter directly to write your programs.
- ✧ **Python is Object-Oriented** – Python supports Object-Oriented style or technique of programming that encapsulates code within objects.

- ❖ **Python is a Beginner's Language** – Python is a great language for the beginner-level programmers and supports the development of a wide range of applications from simple text processing to WWW browsers to games.

History of Python

Python was developed by Guido van Rossum in the late eighties and early nineties at the National Research Institute for Mathematics and Computer Science in the Netherlands.

Python is derived from many other languages, including ABC, **Modula-3**, **C**, **C++**, **Algol-68**, **SmallTalk**, and Unix shell and other scripting languages.

Python is copyrighted. Like Perl, Python source code is now available under the **GNU General Public License (GPL)**.

Python is now maintained by a core development team at the institute, although Guido van Rossum still holds a vital role in directing its progress.

PYTHON FEATURES

Python's features include

- **Easy-to-learn** – Python has few keywords, simple structure, and a clearly defined syntax. This allows the student to pick up the language quickly.
- **Easy-to-read** – Python code is more clearly defined and visible to the eyes.
- **Easy-to-maintain** – Python's source code is fairly easy-to-maintain.
- **A broad standard library** – Python's bulk of the library is very portable and cross-platform compatible on UNIX, Windows, and Macintosh.
- **Interactive Mode** – Python has support for an interactive mode which allows interactive testing and debugging of snippets of code.
- **Portable** – Python can run on a wide variety of hardware platforms and has the same interface on all platforms.

- **Extendable** – You can add low-level modules to the Python interpreter. These modules enable programmers to add to or customize their tools to be more efficient.
- **Databases** – Python provides interfaces to all major commercial databases.
- **GUI Programming** – Python supports GUI applications that can be created and ported to many system calls, libraries and windows systems, such as Windows MFC, Macintosh, and the X Window system of Unix.
- **Scalable** – Python provides a better structure and support for large programs than shell scripting.

Apart from the above-mentioned features, Python has a big list of good features, few are listed below .

- It supports functional and structured programming methods as well as OOP.
- It can be used as a scripting language or can be compiled to byte-code for building large applications.
- It provides very high-level dynamic data types and supports dynamic type checking.
- It supports automatic garbage collection.
- It can be easily integrated with C, C++, COM, ActiveX, CORBA, and Java.

Python is available on a wide variety of platforms including Linux and Mac OS X. Let's understand how to set up our Python environment.

```
>>>print"Hello, Python!"
```

Getting Python

The most up-to-date and current source code, binaries, documentation, news, etc., is available on the official website of Python <https://www.python.org>.

Windows Installation .

Here are the steps to install Python on Windows machine.

1. Open a Web browser and go to <https://www.python.org/downloads/>.
2. Follow the link for the Windows installer python-XYZ.msi where XYZ is the version you need to install.
3. To use this installer python-XYZ.msi, the Windows system must support Microsoft Installer 2.0. Save the installer file to your local machine and then run it to find out if your machine supports MSI.
4. Run the downloaded file. This brings up the Python install wizard, which is really easy to use. Just accept the default settings, wait until the install is finished, and you are done.

The Python language has many similarities to Perl, C, and Java. However, there are some definite differences between the languages.

INTERACTIVE MODE PROGRAMMING

Invoking the interpreter without passing a script file as a parameter brings up the following prompt.

```
$ python  
Python2.4.3(#1,Nov112010,13:34:43)
```

>>>

```
GCC 4.1.220080704(RedHat4.1.2-48)] on linux2
```

```
Type "help", "copyright", "credits" or "license" for more information.
```

Type the following text at the Python prompt and press the Enter

If you are running new version of Python, then you would need to use print statement with parenthesis as in print ("Hello, Python!");. However in Python version 2.4.3, this produces the following result .

```
Hello, Python!
```

SCRIPT MODE PROGRAMMING

Invoking the interpreter with a script parameter begins execution of the script and continues until the script is finished. When the script is finished, the interpreter is no longer active.

Let us write a simple Python program in a script. Python files have extension .py. Type the following source code in a test.py file

```
print"Hello, Python!"
```

We assume that you have Python interpreter set in **PATH** variable. Now, try to run this program as follows

```
$ python test.py
```

This produces the following result .

```
Hello, Python!
```

FLASK FRAMEWORK

Flask is a web application framework written in Python. Armin Ronacher, who leads an international group of Python enthusiasts named Pocco, develops it. Flask is based on Werkzeug WSGI toolkit and Jinja2 template engine. Both are Pocco projects.

Http protocol is the foundation of data communication in world wide web. Different methods of data retrieval from specified URL are defined in this protocol.

The following table summarizes different http method.

Sr.No	Methods & Description
1.	GET Sends data in unencrypted form to the server. Most common method.
2.	HEAD Same as GET, but without response body
3.	POST Used to send HTML form data to server. Data received by POST method is not cached by server.
4.	PUT Replaces all current representations of the target resource with the uploaded content.
5.	DELETE Removes all current representations of the target resource given by a URL .

By default, the Flask route responds to the **GET** requests. However, this preference can be altered by providing methods argument to **route()** decorator.

In order to demonstrate the use of **POST** method in **URL** routing, first let us create an **HTML** form and use the **POST** method to send form data to a **URL**.

Save the following script as login.html

```
<html>

<body>

<form action="http://localhost:5000/login" method="post"
">

<p>Enter Name:</p>

<p><input type="text" name="nm" /></p>

<p><input type="submit" value="submit" /></p>

</form>

</html>
```

Now enter the following script in Python shell.

```
from flask import Flask, redirect,url_for, request

app=Flask(__name__)
```

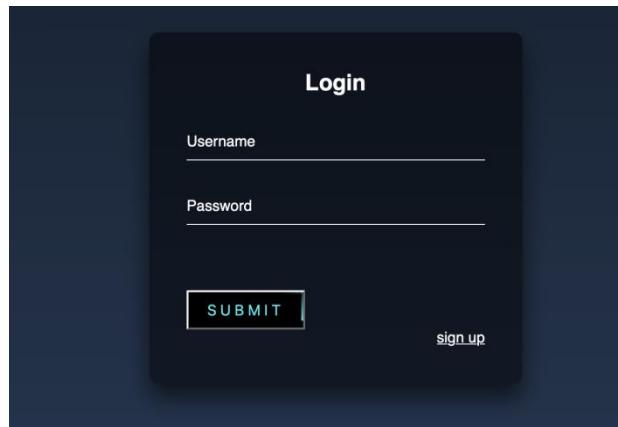


Fig:3.2 Login page

Form data is POSTed to the URL in action clause of form tag.

http://localhost/login is mapped to the **login() function**. Since the server has received data by **POST** method, value of „nm“ parameter obtained from the form data is obtained by .

```
user = request.form['nm']
```

It is passed to „/success“ URL as variable part. The browser displays a **welcome** message in the window.

MODULES

- A.**Data Use --p
- B.**Preprocessing
- C.**Feature Extraction
- D.**Training the Classifier

CHAPTER 4

MODULES DESCRIPTION

A.Data Use

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

B.Preprocessing

The data set used is split into a training set and a testing set containing in Dataset I 13256 training data and 814 testing data and in Dataset II- 1882 training data and 471 testing data respectively. Cleaning the data is always the first step. In this, those words are removed from the dataset. That helps in mining the useful information. Whenever we collect data online, it sometimes contains the undesirable characters like stop words, digits etc. which creates hindrance while spam detection. It helps in removing the texts which are language independent entities and integrate the logic which can improve the accuracy of the identification task.

C.Feature Extraction

Feature extraction s the process of selecting a subset of relevant features for use in model construction. Feature extraction methods helps in to create an accurate

predictive model. They help in selecting features that will give better accuracy. When the input data to an algorithm is too large to be handled and it is supposed to be redundant then the input data will be transformed into a reduced illustration set of features also named feature vectors. Altering the input data to perform the desired task using this reduced representation instead of the full-size input. Feature extraction is performed on raw data prior to applying any machine learning algorithm, on the transformed data in feature space.

D.Training the Classifier

As In this project I am using Scikit-Learn Machine learning library for implementing the architecture. Scikit Learn is an open source python Machine Learning library which comes bundled in 3rd distribution anaconda. This just needs importing the packages and you can compile the command as soon as you write it. If the command doesn't run, we can get the error at the same time. I am using 4 different algorithms and I have trained these 4 models i.e. Naïve Bayes, Support Vector Machine, K Nearest Neighbors and Logistic Regression which are very popular methods for document classification problem. Once the classifiers are trained, we can check the performance of the models on test-set. We can extract the word count vector for each mail in test-set and predict its class with the trained models.

Algorithms

Naive Bayes

1. One of supervised learning algorithm based on probabilistic classification technique.
2. It is a powerful and fast algorithm for predictive modelling.
3. In this project, I have used the Multinomial Naive Bayes Classifier.

Support Vector Machine- SVM

1.SVM's are a set of supervised learning methods used for classification, and regression.

2.Effective in high dimensional spaces.

3.Uses a subset of training points in the support vector, so it is also memory efficient.

Logistic Regression

1. Linear model for classification rather than regression.
2. The expected values of the response variable are modeled based on combination of values taken by the predictors

CHAPTER 5

RESULTS AND DISCUSSION

- Algorithm's accuracy depends on the type and size of your dataset. More the data, more chances of getting correct accuracy.
- Machine learning depends on the variations and relations
- Understanding what is predictable is as important as trying to predict it.
- While making algorithm choice, speed should be a consideration factor.

REQUIREMENT ANALYSIS

Requirement analysis, also called requirement engineering, is the process of determining user expectations for a new modified product. It encompasses the tasks that determine the need for analysing, documenting, validating and managing software or system requirements. The requirements should be documentable, actionable, measurable, testable and traceable related to identified business needs or opportunities and define to a level of detail, sufficient for system design.

FUNCTIONAL REQUIREMENTS

It is a technical specification requirement for the software products. It is the first step in the requirement analysis process which lists the requirements of particular software systems including functional, performance and security requirements. The function of the system depends mainly on the quality hardware used to run the software with given functionality.

Usability

It specifies how easy the system must be used. It is easy to ask queries in any format which is short or long, porter stemming algorithm stimulates the desired response for user. .

Robustness

It refers to a program that performs well not only under ordinary conditions but also under unusual conditions. It is the ability of the user to cope with errors for irrelevant queries during execution.

Security

The state of providing protected access to resource is security. The system provides good security and unauthorized users cannot access the system there by providing high security.

Reliability

It is the probability of how often the software fails. The measurement is often expressed in MTBF (Mean Time Between Failures). The requirement is needed in order to ensure that the processes work correctly and completely without being aborted. It can handle any load and survive and even capable of working around any failure.

Compatibility

It is supported by version above all web browsers. Using any web servers like localhost makes the system real-time experience.

Flexibility

The flexibility of the project is provided in such a way that it has the ability to run on different environments being executed by different users.

Safety

Safety is a measure taken to prevent trouble. Every query is processed in a secured manner without letting others to know one's personal information.

NON- FUNCTIONAL REQUIREMENTS

Portability

It is the usability of the same software in different environments. The project can be run in any operating system.

Performance

These requirements determine the resources required, time interval, throughput and everything that deals with the performance of the system.

Accuracy

The result of the requesting query is very accurate and high speed of retrieving information. The degree of security provided by the system is high and effective.

Maintainability

Project is simple as further updates can be easily done without affecting its stability. Maintainability basically defines that how easy it is to maintain the system. It means that how easy it is to maintain the system, analyse, change and test the application. Maintainability of this project is simple as further updates can be easily done without affecting its stability.

SYSTEM DESIGN AND TESTING PLAN

INPUT DESIGN

The input design is the link between the information system and the user. It comprises the developing specification and procedures for data preparation and those steps are necessary to put transaction data in to a usable form for processing can be achieved by inspecting the computer to read data from a written or printed document or it can occur by having people keying the data directly into the system. The design of input focuses on controlling the amount of input required, controlling the errors, avoiding delay, avoiding extra steps and keeping the process simple. The input is designed in such a way so that it provides security and ease of use with retaining the privacy. Input Design considered the following things:

[1.1]What data should be given as input?

[1.2]How the data should be arranged or coded?

[1.3]The dialog to guide the operating personnel in providing input.

[1.4]Methods for preparing input validations and steps to follow when error occur.

OUTPUT DESIGN

A quality output is one, which meets the requirements of the end user and presents the information clearly. In any system results of processing are communicated to the users and to other system through outputs. In output design it is determined how the information is to be displaced for immediate need and also the hard copy output. It is the most important and direct source information to the user. Efficient and intelligent output design improves the system's relationship to help user decision-making.

The output form of an information system should accomplish one or more of the following objectives.

- Convey information about past activities, current status or projections of the Future.
- Signal important events, opportunities, problems, or warnings.
- Trigger an action.
- Confirm an action.

SOCIAL FEASIBILITY

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

DATA FLOW DIAGRAM

- The DFD is also called as bubble chart. It is a simple graphical formalism that can be used to represent a system in terms of input data to the system, various processing carried out on this data, and the output data is generated by this system.
- The data flow diagram (DFD) is one of the most important modeling tools. It is used to model the system components. These components are the system process, the data used by the process, an external entity that interacts with the system and the information flows in the system.
- DFD shows how the information moves through the system and how it is modified by a series of transformations. It is a graphical technique that depicts information flow and the transformations that are applied as data moves from input to output.
- DFD is also known as bubble chart. A DFD may be used to represent a system at any level of abstraction. DFD may be partitioned into levels that represent increasing information flow and functional detail.
- It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration.

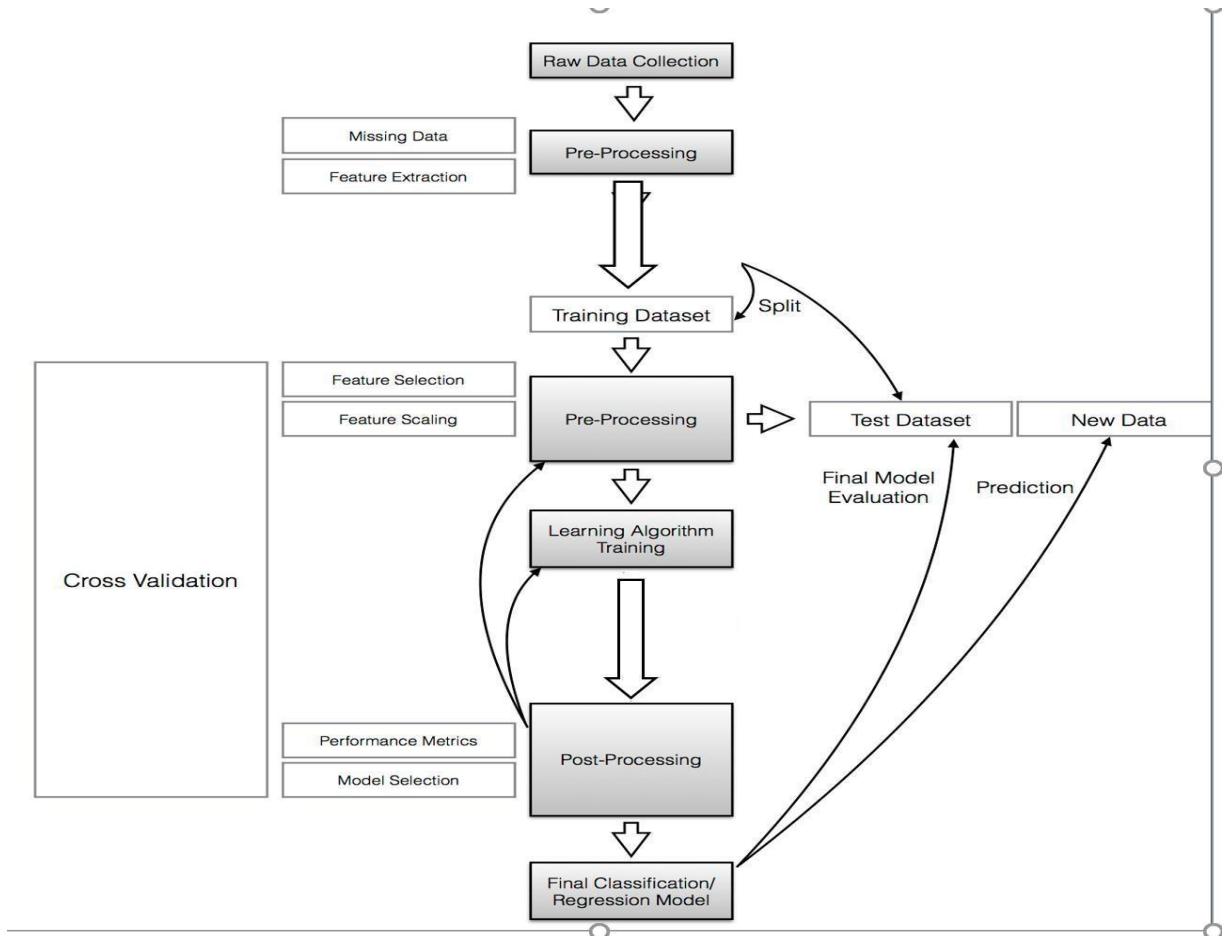


Fig:4.1 Data Flow Diagram

TEST PROCEDURE

SYSTEM TESTING

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub assemblies, assemblies and/or a finished product. It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

UNIT TESTING

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

INTEGRATION TESTING

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

FUNCTIONAL TESTING

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

Valid Input: identified classes of valid input must be accepted.

Invalid Input: identified classes of invalid input must be rejected.

Function: identified functions must be exercised.

Output: identified classes of application outputs must be exercised.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

WHITE BOX TESTING

White Box Testing is a testing in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is used to test areas that cannot be reached from a black box level. Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box .you cannot “see” into it. The test provides inputs and responds to outputs without considering how the software works .

ACCEPTANCE TESTING

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

CHAPTER 6

CONCLUSION AND FUTURE WORK

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

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential. This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

APPENDIX

A) SOURCE CODE

```
from PyQt5 import Qtcore, QtGui, QtWidgets  
  
from Admin import Ui_Admin  
  
import pandas as pd  
  
class Ui_Dialog(object):  
  
    def admin(self, event):  
  
        try:  
  
            self.admin = QtWidgets.QDialog()  
  
            self.ui = Ui_Admin(self.admin)  
  
            self.ui .setupUi(self.admin)  
  
            self.admin.show()  
  
        except Exception as e:  
  
            print(e.args[0])  
  
            tb = sys.exc_info()[2]  
  
            print(tb.tb_lineno)  
  
            event.accept()  
  
    def setupUi(self, Dialog):  
  
        Dialog.setObjectName("Dialog")  
  
        Dialog.resize(702, 435)  
  
        Dialog.setStyleSheet("background-color: rgb(0, 85, 127);")  
  
        self.label = QtWidgets.QLabel(Dialog)  
  
        self.label.setGeometry(Qtcore.QRect(60, 60, 601, 41))  
  
        self.label.setStyleSheet("color:rgb(255, 255, 255);\n" "font: 75 18pt \"Tahoma\";  
;"")  
  
        self.label.setObjectName("label")  
  
        self.label_2 = QtWidgets.QLabel(Dialog)
```

```

    self.label_2setGeometry(Qtcore.QRect(200, 150, 261, 181))

    self.label.setStyleSheet("image:url(..../N-Grams/images/admin.png);")

    self.label_2setText("")

    self.label_2setObjectName("label_2")

    self.label_2.mousePressEvent = self.admin

    self.retranslateUi(Dialog)

    QtCore.QMetaObject.connectSlotsByName(Dialog)

def retranslateUi(self, Dialog):

    _translate = QtCore.QCoreApplication.translate

    Dialog.setWindowTitle(_translate("Dialog", "Online Fake News Using NGram Analysis"))

    if __name__ == "__main__":
        import sys

        app = QtWidgets.QApplication(sys.argv)

        Dialog = QtWidgets.QDialog()

        ui = Ui_Dialog()

        ui.setupUi(Dialog)

        Dialog.show()

        sys.exit(app.exec_())

```

B)SCREENSHOTS

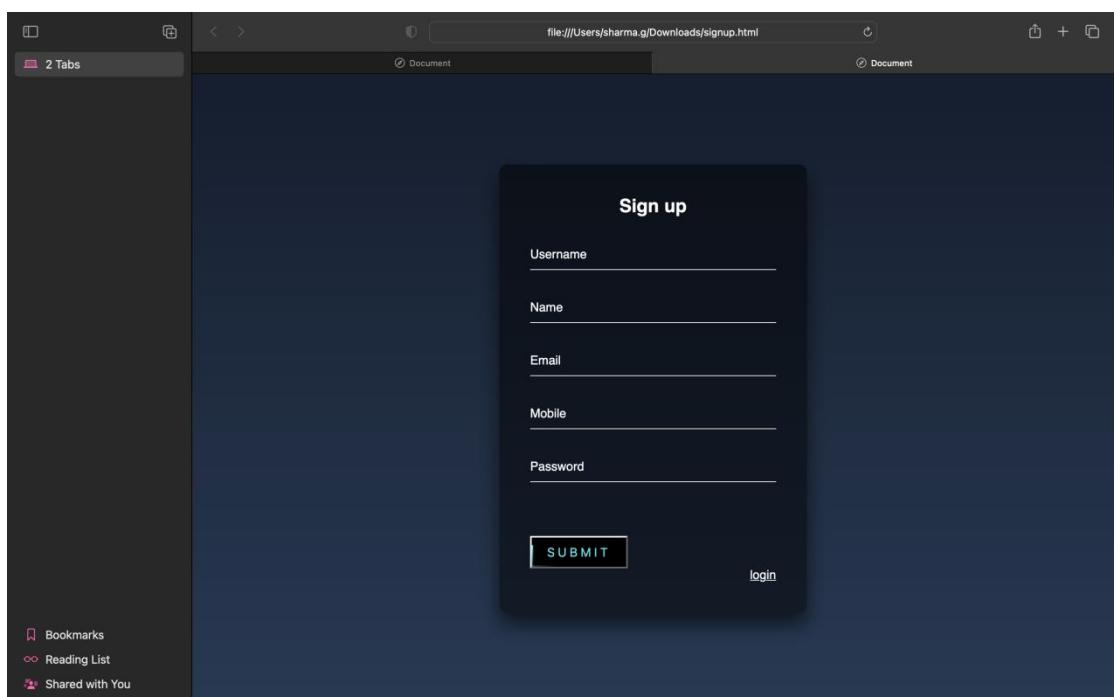


Fig:5.1 Sign up

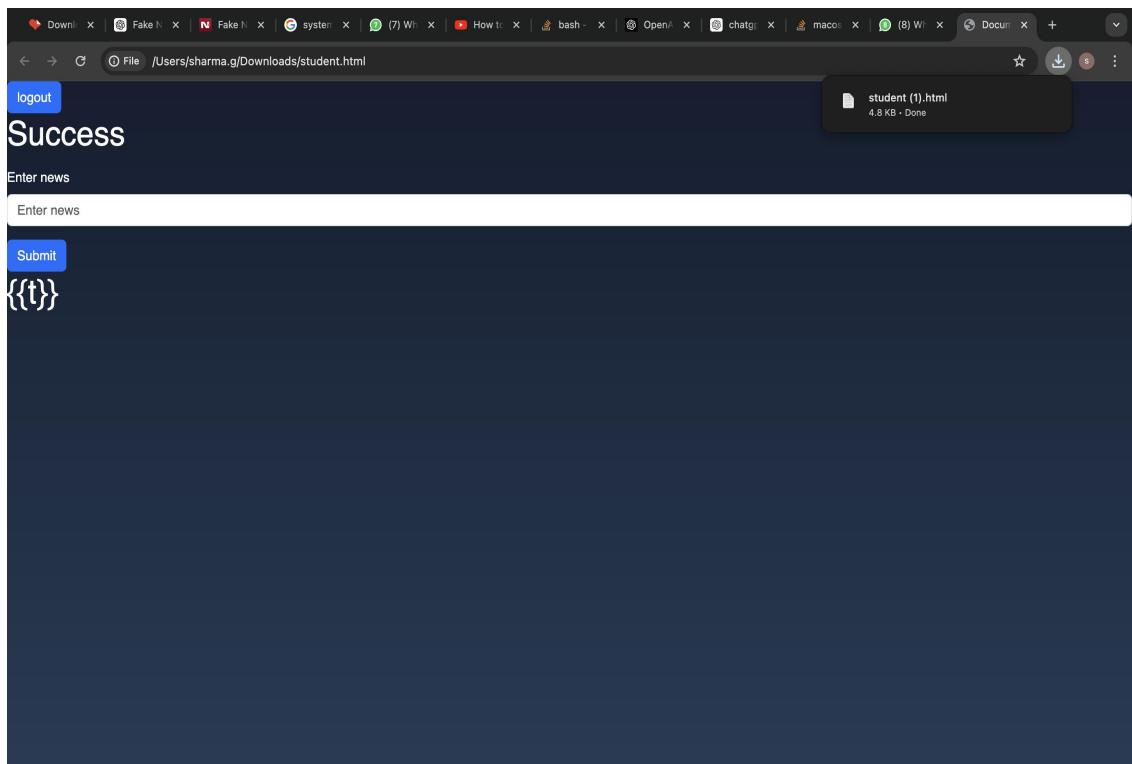


Fig:5.2 Admin Page

The screenshot shows a Jupyter Notebook interface with several cells of Python code and a PDF viewer displaying news articles.

```

In [41]: result = bnb.predict(cv.transform(["According to the Census Bureau, '5 million more people voted than expected in the 2020 election'"]))
if result==0:
    print('FAKE')
else:
    print('REAL')

FAKE

In [42]: #For testing
result = bnb.predict(cv.transform(["Comparing the price of oil and gas in June 2008 to March 2022 is a good way to understand inflation."]))
if result==0:
    print('FAKE')
else:
    print('REAL')

FAKE

```

#Conclusion and Limitations: We used 3 popular Naive Bayes algorithms of scikit-learn library - Gaussian, Bernoulli and Multinomial. The accuracy, precision, recall and F1 scores for all 3 algorithms are as follow:

Gaussian Naive Bayes:

- Accuracy (0.8416)
- Precision (0.796524626199279),
- Recall (0.796523711961647) and
- F1 (0.853711521216997)

Bernoulli Naive Bayes:

- Accuracy (0.9486666666666667),
- Precision (0.9486666666666667),
- Recall (0.9486666666666667) and
- F1 (0.9486666666666667)

Multinomial Naive Bayes:

- Accuracy (0.9426666666666667),
- Precision (0.9426666666666667),
- Recall (0.9426666666666667) and
- F1 (0.9426666666666667)

Bernoulli Naive Bayes is proven to be the most effective algorithm for detecting Fake News. Although Multinomial Naive Bayes also performed very close to Bernoulli. As we can see in the testing, our model could successfully detect the fake news and the real news. Due to some limitation in our computing capacity, we could not use all 44,898 rows and used 30,000 rows instead. This model can be tested on larger datasets.

Fig:5.3 Accuracy Level of Algorithms with dataset

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
	Font	Font	Font	Font	Font	Font	Font	Font	Font	Font	Font	Font	Font	Font
1.	title													
2.	Donald Trump Sends Out Embarrassing New Year's Eve Message; This is Disturbing		Donald Tr. News	December 31, 2017										
3.	Drunk Bragging Trump Staffer Started Russian Collusion Investigation		House Int'l News	December 31, 2017										
4.	Sheriff David Clarke Becomes An Internet Joke For Threatening To Poke People At 'In The Eye'		On Friday, News	December 30, 2017										
5.	Trump Is So Obsessed He Even Has Obamas' Name Coded Into His Website (IMAGES)		On Christ News	December 29, 2017										
6.	Pope Francis Just Called Out Donald Trump During His Christmas Speech		Pope Fran News	December 25, 2017										
7.	Racist Alabama Cops Brutalize Black Boy While He Is In Handcuffs (GRAPHIC IMAGES)		The numb News	December 25, 2017										
8.	FBI Director James Comey Just Told Congress He Wasn't Taping Conversations		Donald Tr. News	December 23, 2017										
9.	Trump Said Some INSANELY Racist Stuff Inside The Oval Office, And Witnesses Heard It Up		In the Week News	December 23, 2017										
10.	Former CIA Director Shams Trump Over UN Bullying, Openly Suggests He's Acting Like A Dictator (TWEET)		Many ppos News	December 22, 2017										
11.	WATCH: Brianne Ryan Just Told Us He Doesn't Care About Struggling Families Living In Blue States		Just when News	December 21, 2017										
12.	Papa John's Founder Retires, Figures Out Racism Is Bad For Business		A centrep. News	December 21, 2017										
13.	WATCH: Paul Ryan Just Told Us He Doesn't Care About Struggling Families Living In Blue States		Republica News	December 21, 2017										
14.	Bad News For Trump: Mitch McConnell Says No To Repealing Obamacare In 2018		Republica News	December 21, 2017										
15.	WATCH: Lindsey Graham Trashes Media For Portraying Trump As An 'Idiot' & Forgets His Own Words		The media News	December 20, 2017										
16.	Heiress To Disney Empire Knows GOP Scammed Us As 'Shredded Them For Tax Bill		Abigail Dis News	December 20, 2017										
17.	Tone Deaf Trump: Congratulates Rep. Scalise On Losing Weight After You Almost Died		Donald Tr. News	December 20, 2017										
18.	The White House Is Finally Starting To Call Donald Trump By His Real Name		A new and News	December 17, 2017										
19.	The Mueller Spokesman Just Faded Up Donald Trump's Christmas		Trump up News	December 17, 2017										
20.	SNL Hilariously Mocks Accused Child Molester Roy Moore For Losing AL Senate Race (VIDEO)		Right now News	December 17, 2017										
21.	Republican Senator Gets Dragged For Going After Robert Mueller		Senate Ma News	December 16, 2017										
22.	In A Heartless Retake To Victims, Trump Invites NRA To Xmas Party On Sandy Hook Anniversary		It almost s News	December 16, 2017										
23.	KY GOP State Rep. Commits Suicide Over Allegations He Molested A Teen Girl (DETAILS)		In this AM News	December 13, 2017										
24.	Megan McCain Tweets The Most AMAZING Response To Doug Jones' "Win In Deep Red Alabama		As a Demc. News	December 12, 2017										
25.	CNN CALLS IT: A Democrat Will Represent Alabama In The Senate For The First Time In 25 Years		Alabama ir News	December 12, 2017										
26.	White House It Wasn't A Sexist For Trump To Slut-Shame Sen. Kirsten Gillibrand (VIDEO)		A backfas! News	December 12, 2017										
27.	Despicable Trump Suggests Female Senator Would Be 'Do Anything' With Him For Campaign Money (TWEET)		Donald Tr. News	December 12, 2017										
28.	Accused Child Molester Roy Moore Loses Alabama Senate Race To Doug Jones Over Reagan (VIDEO)		Ronald Re News	December 11, 2017										
29.	WATCH: Donald Trump Calls For A 'Big Cleaning' At The FBI And To Arrest Everyone Investigating Trump		Justice Re News	December 10, 2017										
30.	Liberal Group Tells Trump To Roy Moore Rally In The Best Possible Way (VIDEO)		Donald Tr. News	December 9, 2017										
31.	Don Jr. tries To Mock Al Franken's Resignation, Backfires Immediately		Whom Sen News	December 7, 2017										
32.	BREAKING: Coi Finally Gets His Due, Walter Scott's Killer Sentenced To Prison (DETAILS)		In America News	December 7, 2017										
33.	WATCH: Is This Proof Trump Is Unfit for Service?		New cust. News	December 7, 2017										

Fig:5.4 Detecting Fake News

	A	B	C	D	E
1	title				
2	As U.S. budget fight looms, Republicans flip their fiscal script	WASHINGTON	politics	Nei	December 31, 2017
3	U.S. military to accept transgender recruits on Monday: Pentagon	WASHINGTON	politics	Nei	December 29, 2017
4	Senior U.S. Republican senator: 'Let Mr. Mueller do his job'	WASHINGTON	politics	Nei	December 31, 2017
5	FBI Russia probe helped by Australian diplomat tip-off: NYT	WASHINGTON	politics	Nei	December 30, 2017
6	Trump wants Postal Service to charge 'much more' for Amazon shipments	SEATTLE/W	politics	Nei	December 29, 2017
7	White House, Congress prepare for talks on spending, immigration	WEST PALM	politics	Nei	December 29, 2017
8	Trump says Russia probe will be fair, but timeline unclear: NYT	WEST PALM	politics	Nei	December 29, 2017
9	Factbox: Trump on Twitter (Dec 29) - Approval rating, Amazon	The followi	politics	Nei	December 29, 2017
10	Trump on Twitter (Dec 28) - Global Warming	The followi	politics	Nei	December 29, 2017
11	Alabama official to certify Senator-elect Jones today despite challenge: CNN	WASHINGTON	politics	Nei	December 28, 2017
12	Jones certified U.S. Senate winner despite Moore challenge	(Reuters) -	politics	Nei	December 28, 2017
13	New York governor questions the constitutionality of federal tax overhaul	NEW YORK	politics	Nei	December 28, 2017
14	Factbox: Trump on Twitter (Dec 28) - Vanity Fair, Hillary Clinton	The followi	politics	Nei	December 28, 2017
15	Trump on Twitter (Dec 27) - Trump, Iraq, Syria	The followi	politics	Nei	December 28, 2017
16	Man says he delivered manure to Mnuchin to protest new U.S. tax law	(In Dec. 25	politics	Nei	December 25, 2017
17	Virginia officials postpone lottery drawing to decide tied statehouse election	(Reuters) -	politics	Nei	December 27, 2017
18	U.S. lawmakers question businessman at 2016 Trump Tower meeting: sources	WASHINGTON	politics	Nei	December 27, 2017
19	Trump on Twitter (Dec 26) - Hillary Clinton, Tax Cut Bill	The followi	politics	Nei	December 26, 2017
20	U.S. appeals court rejects challenge to Trump voter fraud panel	(Reuters) -	politics	Nei	December 26, 2017
21	Treasury Secretary Mnuchin was sent gift-wrapped box of horse manure: reports	(Reuters) -	politics	Nei	December 24, 2017
22	Federal judge partially lifts Trump's latest refugee restrictions	WASHINGTON	politics	Nei	December 24, 2017

Fig:5.5 Detecting True News

REFERENCES

- [1] M. D. N. Darji, S. M. Parikh, and H. R. Patel, “Sentiment analysis of unstructured data using spark for predicting stock market price movement,” in *Inventive Computation and Information Technologies*. Singapore: Springer, 2022, pp. 521–530.
- [2] “A review on fake news detection 3T’s: typology, time of detection, taxonomies”.link: [https://link.springer.com/article/10.1007/s10207-022-00625-3.\(2023\)](https://link.springer.com/article/10.1007/s10207-022-00625-3.(2023))
- [3] “Fake news detection based on news content and social contexts: a transformer-based approach”, link: [https://link.springer.com/article/10.1007/s41060-021-00302-z.\(2023\)](https://link.springer.com/article/10.1007/s41060-021-00302-z.(2023))
- [4] P. H. A. Faustini and T. F. Covões, “Fake news detection in multiple platforms and languages,” *Expert Syst. Appl.*, vol. 158, Nov. 2022, Art. no. 11350
- [5] A. Abbasi, A. R. Javed, C. Chakraborty, J. Nebhen, W. Zehra, and Z. Jalil, “ElStream: An ensemble learning approach for concept drift detection in dynamic social big data stream learning,” *IEEE Access*, vol. 9, pp. 66408–66419, 2021.
- [6] A. Vereshchaka, S. Cosimini, and W. Dong, “Analyzing and distinguishing fake and real news to mitigate the problem of disinformation,” *Comput. Math. Org. Theory*, vol. 26, no. 3, pp. 350–364, Sep. 2020.

- [7] M. Z. Asghar, A. Habib, A. Habib, A. Khan, R. Ali, and A. Khattak, “Exploring deep neural networks for rumor detection,” *J. Ambient Intell. Hum. Comput.*, vol. 12, no. 4, pp. 4315–4333, Apr. 2021
- [8] M. H. Goldani, S. Momtazi, and R. Safabakhsh, “Detecting fake news with capsule neural networks,” 2021, arXiv:2002.01030.
- [9] M. Lai, A. T. Cignarella, D. I. Hernández Farías, C. Bosco, V. Patti, and P. Rosso, “Multilingual stance detection in social media political debates,” *Comput. Speech Lang.*, vol. 63, Sep. 2020, Art. no. 101075.
- [10] S. S. Jadhav and S. D. Thepade, “Fake news identification and classification using DSSM and improved recurrent neural network classifier,” *Appl. Artif. Intell.*, vol. 33, no. 12, pp. 1058–1068, Oct. 2019