FAKE NEWS DETECTION SYSTEM

Muhammad Ammaar Akhtar

Supervised By

Dr. Muhammad Noman Malik

Submitted for the partial fulfillment of BS Computer Science degree to the Faculty of Engineering & CS

DEPARTMENT OF COMPUTER SCIENCE

NATIONAL UNIVERSITY OF MODERN LANGUAGES

ISLAMABAD

May, 2021

ABSTRACT

Fake News Detection System is a web-based application that detects fake news over the internet. These days social media sites are extensively used as a source of news because of convenience, low cost, ease of access nature. In an attempt to stop the spreading of false information, some fact-checking websites are there to detect fake news. These websites play a significant role in filtering out the fake news, but they require the expert's analysis on the news and some use the public flagging which is a time-consuming process. Because of the diversity and volume of social media sites, it is not possible to manually label each and every news as real or fake. The proposed Fake News Detection System is capable of classifying unreliable news into real and fake news using machine learning and natural language processing techniques.

Fake News Detection System acquires the title, author name, and content of the news from the user then pre-processes those attribute's content by applying text processing techniques to clean the content of the news. A trained model used to extract features from the cleaned data and by comparing the extracted features with the model to make a decision on the news content, the model is trained using a logistic regression machine learning algorithm.

For Fake News Detection System testing phase starts with unit testing to test different modules separately, after successfully applying unit testing, the testing phase moved forward to the integration testing applied after integrating different modules, then the system testing is done by using white box testing.

As for the limitations of the Fake News Detection System, this is not able to extract news from the images, and videos and this system is not fully automated user has to input the news manually. In addition, the trained model is trained to detect news from US news agencies. A possible improvement is that make the system capable to extract news content from the images and the videos. As for future work, build a browser plugin, which is capable to get necessary data from the website page, by the enhancement the system will become a fully automated Fake News Detection System.

CERTIFICATE

Dated: ________________

Final Approval

It is certified that project report titled **'Fake News Detection System'** submitted by **Muhammad Ammaar Akhtar** for the partial fulfillment of the requirement of **"Bachelor's Degree in Computer Science"** is approved.

COMMITTEE

Dr. Basit Shahzad

Dean Engineering & CS Signature: ____________________

Dr. Sajjad Haider

HoD Computer Science: Signature: ____________________

Ms. Mehvish Sabih

Head Project Committee Signature: ____________________

Dr. Muhammad Noman Malik

Supervisor: Signature: ____________________

DECLARATION

I hereby declare that out dissertation is entirely our work and genuine / original. I understand that in case of discovery of any PLAGIARISM at any stage, I will be assigned an F (FAIL) grade and it may result in withdrawal of my Bachelor’s degree

Name	**Signature**
Muhammad Ammaar Akhtar	______________

PLAIGRISM CERTIFICATE

This is to certify that the project entitled **“Fake News Detection System”**, which is bring submitted here with for the award of the **“Degree of Bachelor’s in Computer Science”.** This is the result of the original work by **Muhammad Ammaar Akhtar** under my supervision and guidance. The work embodied in this project has not been done earlier for the basis of award of any degree or compatible certificate or similar title of this for any other diploma/examining body or university to the best of my knowledge and belief.

Turnitin Originality Report

Processed on 30-May-2021 06:42PM (UTC+0500)

ID: 1597083898

Word Count: 8921

Similarity Index:

9%

Similarity by Source

Internet Sources:	5%
Publications:	4%
Student Papers:	4%

Date: 30-May-2021

Dr. Muhammad Noman Malik (Supervisor)

TURNITIN ORIGNALITY REPORT

Fake News Detection System [BSCS] by **Muhammad Ammaar Akhtar**

From **Dr. Muhammad Noman Malik**

Processed on 30-May-2021 06:42PM (UTC+0500)

ID: 1597083898

Word Count: 8921

Similarity Index:

9%

Similarity by Source

Internet Sources:

5%

Publications:

4%

Student Papers:

4%

SOURCES:

1. 2% match (Internet Source)

www.researchgate.net

2. 1% match (Student Paper)

Submitted to Higher Education Commission Pakistan

3. 1% match (Publication)

Bashar Al Asaad, Madalina Erascu. "A Tool for Fake News Detection" , 2018 20th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC), 2018

4. 1% match (Publication)

Kai Shu, Deepak Mahudeswaran, Huan Liu. "FakeNewsTracker: a tool for fake news collection, detection, and visualization”, Computational and Mathematical Organization Theory, 2018

5. <1% match (Internet Source)

dennisholeman.com

6. <1% match (Internet Source)

eprints.utm.my

7. <1% match (Student Paper)

Submitted to University of Westminster

8. <1% match (Student Paper)

Submitted to Amity University

9. <1% match (Publication)

"Testing Software and Systems”, Springer Science and Business Media LLC, 2013

10. <1% match (Student Paper)

Submitted to Binus University International

11. <1% match (Student Paper)

Submitted to Regis University

12. <1% match (Internet Source)

repository.unika.ac.id

13. <1% match (Internet Source)

isomase.org

14. <1% match (Internet Source)

www.mdpi.com

15. <1% match (Internet Source)

www.molecules-project.eu In

16. <1% match (Student Paper)

Submitted to University of Mary HardinBaylor

17. <1% match (Student Paper)

Submitted to London School of Commerce and IT

18. <1% match (Student Paper)

Submitted to West University Of Timisoara

19. <1% match (Internet Source)

muroran-it.repo.nii.ac.jp

20. <1% match (Internet Source)

scholarbank.nus.edu.sg

21. <1% match (Student Paper)

Submitted to University of Greenwich

22. <1% match (Internet Source)

cwww.intechopen.com

23. <1% match (Internet Source)

odr.chalmers.se

24. <1% match (Student Paper)

Submitted to University of Hertfordshire

25. <1% match (Internet Source)

pure.southwales.ac.uk

26. <1% match (Internet Source)

utpedia.utp.edu.my

27. <1% match (Publication)

Elad Segev. "Mass media channels and the world of Google News", Elsevier BV, 2010

28. <1% match (Publication)

Sawinder Kaur, Parteek Kumar, Ponnurangam Kumaraguru. "Automating fake news detection system using multi-level voting model", Soft Computing, 2019

29. <1% match (Internet Source)

par.nsf.gov

30. <1% match (Publication)

Yoothana Suansook, Tanongsak Taweesri. "Modeling Language for Systems Engineering in Defense Industry", Proceedings, 2019

31. <1% match (Publication)

Somya Ranjan Sahoo, B.B. Gupta. "Multiple features based approach for automatic fake news detection on social networks using deep learning", Applied Soft Computing, 2020

ACKNOWLEDGMENT

I am very grateful to Almighty Allah for the good health and wellbeing that were necessary to complete this project.

I am sincerely thankful to Dr. Muhammad Noman Malik, who is my project supervisor for this project. I have received continuous guidance and support from him to keep my progress in an organized flow, especially in the challenging times of lockdown due to COVID-19.

In the end, I would like to say thanks to my family for providing support and keeping me motivated, especially my father Engr. Muhammad Saleem Akhtar.

TABLE OF CONTENTS

Chapter **Page**

Chapter 1

Chapter 2

Chapter 3

Chapter 4

Chapter 5

Chapter 6

LIST OF FIGURES

Figure **Caption** **Page No.**

LIST OF TABLES

Table **Caption** **Page**

Chapter 1

INTRODUCTION

Fake news are mostly fabricated to attract public attention, propagate political views, destroy reputations, or to blame religion and the culture of the people. Fake news has been there since "The New York Sun claimed there was a civilization on the moon back in 1835" a story known as the "Great moon hoax" that makes "The New York Sun" the most profitable newspaper. Fake news detection of online news gets more attention right after the US presidential election in 2016. According to a post-election report [1], there is more than 41.8% data traffic of fake news over the social media networks during the US presidential election 2016. In an attempt to stop the spreading of false information, some fact-checking websites are there to detect fake news. These websites play a significant role in filtering out the fake news, but they require the expert's analysis on the news and some use the public flagging which is a time-consuming process. Because of the diversity and volume of social media sites, it is not possible to manually label each and every news as real or fake. The motivation behind this project is to gain trust back on online news which is dropped to 34% according to Edward [2]. Also, make it easy to get easy access to the fake news detection system.

Fake News Detection system (FNDS) is a web-based application that detects fake news over the internet. This system relay on a machine learning trained model to detect fake and real news, not on the expert's time-consuming analysis as for recent trends, peoples tend to use technology to overcome the problems. Comparing FNDS with the existing systems, this system does not depend on any person's opinions or expertise which makes this system much more reliable than the traditional methods of detecting fake news.

1.1. Project Domain

Natural Language Processing (NLP), data analysis, and Machine Learning (ML) are the project domains of FNDS. Micro web framework used to acquire data from the GUI (Graphical User Interface) form to manipulatable data and make possible to perform feature extraction on the acquired data using Natural Language Processing techniques. Machine Learning techniques are used to analyze the extracted features from the data and develop a decision-making model on the given datasets. Fake News Detection System is developed using the above mention techniques, web framework is used to extract data from the GUI, then preprocessing is performed on the acquired data to get cleaned and raw data, for this purpose, Natural Language Processing techniques are used. To formulate a predictive model, Machine Learning algorithms are used.

1.2. Problem Identification

As for now, there are several fact-checking websites that are developed to detect fake news over the internet. Presently, most of the developed systems are dependent on the expert's analysis or those systems detect fake news by using public polling systems. Concerned researchers working on how to get better results from present technology and meanwhile developing alternative techniques to tackle fake news detection problems. There is not any operating commercial system which is developed using artificial intelligence methodologies. This project is a contribution in this puzzle-solving where a system is developed by using existing techniques of artificial intelligence to counter fake news.

1.2.1. Proposed Solution

Fake News Detection System is a web-based application that acquires the headline of the news, author of the news, and the content of the news from the user. On the acquired data, the system performs different NLP preprocessing techniques, after preprocessing features are extracted from the data, and then by using predictive model system predict if the entered news is real or the fake one, and displays the result to the user.

1.2.1.1. Objectives

- To design user friendly system to classify unreliable news
- To develop a system that is less time consuming and accurate results producing system
- To reduce the spreading of fake news over the internet
- To gain the trustworthiness over online news.

1.2.1.2. Scope of the project

This Fake News Detection System is not limited to a specific environment and a specific group of peoples. Anyone with the access to internet and web page supportive device can use this system and facilitate themselves. For now, due to the unavailability of the datasets, it is not possible to detect fake news other than the US news forums. And to keep the project within the manageable scope with respect to time and resources FNDS is developed only for textural news classification.

1.3. Effectiveness / Usefulness of the System

How do humans obtain a perspective about anything? Mostly it starts by gathering information about a certain subject by reading relative literature and in modern times usually it is from online platforms, due to the presence of variety of digital sources and enhanced information. This information precedes an individual to have a perspective by observation, if the information has a negative impression about something, that individual would share the same perspective. To overcome this issue, this study presents this simple and easy-to-utilize system for the end-user so they can easily classify between authentic and fake information before getting an erroneous impression or perspective about anything.

1.4. Resource Requirement

To develop a solution as proposed in Section 1.2.1.1, the required hardware, software, and data resources are discussed in the following Sections.

1.4.1. Hardware

The hardware used for this application development is a workstation (E5-1620). The workstation on which this application is developed have at a 3.70 GHz processor speed with 4 cores and 8 logical processors with a of 32 GB RAM. The workstation has 256 GB of SSD and 2TB of disk space.

1.4.2. Software

PyCharm is a user-friendly Integrated Development Environment (IDE) specifically for python development, PyCharm provide its own virtual environment for many libraries. Jupyter notebook is used for the machine learning pipeline coding.

1.4.3. Data Resources

For this system we adopted Kaggle dataset of fake news [13], consists of train dataset containing 20800 records and test dataset containing 5200 records.

1.5. Report Organization

This chapter has briefly described the project domain and the problem that will be solved by the system and also how to overcome this problem, then the effectiveness of the proposed system, and also discussed the required resources to develop the proposed system. Chapter 2 will discuss the related work of researchers and discuss the existing systems and their methodologies as well as their limitations. Chapter 3 will discuss the system specifications, briefly describe the modules of the proposed system, then the functional and non-functional requirements of the proposed system. Chapter 4 will discuss the different methods and techniques used to design the system and perform the analysis on the system and how system design and analysis perform on the FNDS. Chapter 5 will discuss the different testing techniques that are necessary to perform on any system and how the testing of the FNDS is performed, Chapter 6 will conclude the system documentation and also discuss the limitation of the FNDS and possible future work that can be performed to make this system more reliable and user friendly.

Chapter 2

BACKGROUND AND EXISTING SYSTEM

Online news are controlling lives of people from going to coffee and to arrange an important event, people rely heavily on the news. Currently, due to the number of unreliable resources, it is very difficult to avail valid news. Fake News Detection System is a tool that assists the users with its predictive capabilities to classify deceptive news over authentic ones.

This chapter, will discuss the related literature, evaluate currently used systems, identify problems in the present systems/applications, and define the scope of the proposed system.

2.1. Related literature Review

In a study by B. Al Asaad and M. Erascu [3], a number of algorithms are presented that extract the news data and verify the fake news by using machine learning and parsing techniques. The presented algorithm first extracts the textual data like news title, the content of the news article, the publication date of the news, and the author of the news from the news site using the news URL. Then by using another algorithm tool verify the URL if the URL is trustworthy or not if not, the tool marked the news as fake and if the URL is a trustworthy tool, then classify the title of the news either clickbait or not, and the content of the news either fake or real by using machine learning techniques. In the end, the tool uses the extracted publication date to find out the if the incident happened on the same date by extracting the list of the news title happened on the same date and match the extracted title with the list of the news titles and perform the cosine similarity approach to find the title which is more similar to the extracted title.

In a study by Bourgonje [4], a fake news detection system is proposed where the detection of the fake news is based on either the title of the news is clickbait or not and the title is related to the news content or not. The system finds the relationship of the title of the article and the content of the news. To develop that system the fake news challenge [8] dataset is used. Fake news challenge dataset has the news title and the content of the news and labeled as related or unrelated. The proposed algorithm finds the relationship of the title of the article and the content of the news by using CoreNLP Lemmatizer [10]. The n-gram techniques used to calculate the similarity score of the title and the content of the news by finding and matching content of the news title and the article of the news. Term frequency (TF) and the Term Frequency Inverse document frequency (TF-IDF) [11] are used to calculate the matching score. Different from this, the proposed system does not check if the title of an article is related to its content.

In a study by Rubin [5], present a tool for fake news detection. The authors of the paper convinced to distinguish between fake and real news are possible by using satirical news as a guide. The authors present a tool that analyzes the textural characteristics of satire news using Support Vector Machine models and 10-fold cross validation to train and evaluate the machine learning model. That model supposed to detect deceptive news fast and efficiently. The main difference to the proposed approach is that this focus on detecting satire articles. The reviewed literature summarized in Table 2.1.

Table 2. 1:Summary of Reviewed Literature

Year	Author	Contribution	Techniques	Limitations
2018	B. Al Asaad and M. Erascu	Present an algorithm that uses machine learning to detect fake news	Parsing and machine learning techniques to classify the articles and the title	Only textural news can be classified
2017	P. Bourgonje, J. M. Schneider	Present an algorithm that verifies the similarity between a headline and content	Similarity score is calculated based on the n-gram technique, the frequency of appearance and the inverse document frequency of a term	If the title of an article is not related to its content system tag it fake
2016	V.Rubin, N. Conroy	Present an automated tool which can indicate deceptive information fast and efficiently by analyzing the satire news text characteristics	Support Vector Machines models and 10-fold cross validation to train and evaluate a machine learning model	They focus on detecting satire articles

2.2. Related Systems/Applications

Our.news is a web-based application that provides the news trustworthiness through the crowdsourcing. Users reads the news and rate that news based on different terminologies like is the news politically biased or not, how much the user trust the author and the publisher of that particular news, user also rates the accuracy of the news, and user choose the kind of that news for better understanding see Figure AP- 1. Additionally, to weight user ratings bias-detection algorithms are used. Users also sees the fact-checking information about the news.

PolitiFact is a web-based application that checks the "newsworthy and significant" of the news and rates them accordingly. PolitiFact use Truth-O-Meter have six different level of the truthfulness as False, Mostly False, Half True, Mostly True, and True. PolitiFact also have a Flip-O-Meter, it has three different level of rating as No Flip, Half Flip, and Full Flop. Filp-O-Meter counter the stand of the politician on its statement how well a politician stands on its position.

Trive is a chrome browser extension that uses crowdsourcing using blockchain indexing to provides fact-checking. It uses an incentive structure of Nash equilibrium game theory for fact-checking. There are several roles and actors to perform their roles that get in the process. The process starts from the consumer, consumer use the Trive browser plugin to submit the news stories those are doubtful then the Curators (to bids their fees for a story to take under consideration), then the Research (picks up the claims one at a time and by using their expertise and tools they come up with the findings), then the Verifier (reviews picked up claims and verify the researcher clam), the Witness (reviews two claims and select any one of them).

To differentiate between developed systems summary of the existing systems is mentioned in table 2.2.

Table 2. 2:Summary of Existing System

Year	System	Contribution	Tool/Technologies	Limitations	Applications
2016	Our.news	Our.news is a working website that provides fact-checking of	Crowdsourcing is used to rate the news article. Web-based application,	Manual system to classify news not an	News classification

		the news through the crowdsourcing.	and also, mobile application available to access	ideal option in present time	
2007	PolitiFact	PolitiFact is a web-based application that checks the "newsworthy and significant" of the news and rates them accordingly	Six level of Truth-O-Meter is used to categorize the news	People manually categorize the news	Fact-checking, news classification
2018	Trive	Trive is a chrome browser extension that uses crowdsourcing using blockchain indexing to provides fact-checking.	Human Swarmed crowd wisdom, produce results on the Block-Chain	Manual classification of news	Truthfulness of news

2.3. Identified Problem from Existing Work

As discussed above about existing work, which covers their working, application, and the limitations, all existing systems are totally dependent on the humans to classify the news, which is a time consuming and difficult process to produce accurate results. To overcome these problems, a system is proposed which is developed by using a machine learning predictive model. It can classify the news rapidly without any biasness and it is designed to be more accurate than humans.

2.4. Selected Boundary for Proposed Solution

The proposed system gets the name of the author of news article, the title of the news, and the content of the news from the user. On the acquired data the system performs different NLP preprocessing techniques, after preprocessing features are extracted from the data, and then by using predictive model, the system predicts whether the given news is real or the fake, and displays the result to the user.

Now to discuss the limitations of the proposed system, currently the proposed system can only classify textual news, which is to keep the project scope manageable by the author.

Chapter 3

SYSTEM REQUIREMENT AND SPECIFICATIONS

This chapter covers the system specifications of FNDS and its different modules, functional requirements, and non-functional requirements of the system.

The first section contains the discussion on the system specification, the second section contains the discussion on different modules of the system, the third section contains the discussion on functional requirements of the system, fourth section, contains the discussion on the non-functional requirements of the system.

3.1. System Specification

System Specification consist of the documentation that elaborate the requirements of the software application that need to take under consideration while developing the software application. The documentation contains all the functional and non-functional requirements of the software application. Functional requirements are the explanations of the functions and the functionality that function performed, as a single unit or as a set of functions and their behavior on single or multiple inputs and the outcomes of those functions. When it comes to Non-Functional Requirements (NFRs) are the Operational capabilities and constraints of the system that positively enhance the functionality of the system depends on non-functional requirement of the system. The non-functional requirements contain the high performance of tools, the accuracy of the structure, usability, speed, and security of the developed system. Detailed discussion of the functional and non-functional requirements of the FNDS are discussed in this chapter.

3.2. System Modules

3.2.1. Graphical User Interface (GUI)

The Graphical User Interface (GUI) is the only way for the user to interact with the system to perform the user related functions and tasks.

3.2.2. Dataset

Dataset is a collection of related information, mostly datasets are text-based data, and stored in the '.csv' files. The dataset is in the tabular form where every column represents a particular variable or the feature and each row contain the record corresponding to that column category.

3.2.3. Dataset Cleaning

Data cleaning is the process to remove null records as mostly datasets are recorded by the automated systems, there are many chances to get inaccurate and unnecessary and stored in the dataset file, to identify and remove the unwanted data is a necessary process that have to take under consideration.

3.2.4. Feature Extraction

Feature extraction is the process to identify the required information and to get that information in the required form are mainly vectorized data is used to train a machine learning classifier. The feature extraction process is performed after the data cleaning to vectorize the feature and make them suitable to be processed by the machine learning classifier.

3.2.5. Modeling

Several machine learning algorithms are available to train a predictive model, to achieve maximum accuracy for FNDS Logistic Regression techniques is used.

3.2.6. User Data Acquiring

To facilitate the user data, this system acquires data from the user through the HTML form.

3.2.7. User Data Cleaning

To extract features from the user data it is important to first clean the data from stopwords as shown in Figure AP- 2 and punctuations.

3.2.8. User data feature extraction

After the cleaning of the user data, it is necessary to extract the features from the user data by vectorizing and make it suitable to be processed by the machine learning classifier.

3.2.9. Inherit Model

To make decisions over the user provided data it is necessary to inherit the trained predictive model pipeline in the system.

3.2.10. Testing

After encapsuling the system, a system testing is performed as mention in section 5.2.5 to verify the working of the system.

3.3. Functional Requirements

Functional Requirements are the basic and essential requirement for any application which is under development functional requirements are basically the function of the system or application in which parameters operate normally and works properly. In this regard, all functional requirements are discussed in detail.

3.3.1. Data Acquisition

The first stage of the system from the user perspective is the data acquisition. Data will be acquired from the user in the textural form with the help of HTML form, user has to enter the author of the news, the title of the news, and the content of the news article.

3.3.2. Data Processing

Acquired data is full of punctuations, null values, and stopwords. To extract necessary features, it is important to first clean the extracted data and to accomplish this the system use different natural language tool libraries.

3.3.3. Feature Extraction

For feature extraction the system first, vectorize the cleaned data that acquired after performing data processing on the data acquired from the HTML form fields, and then data is weighted according to the number of times a specific vector appears in a document over the total number of vectors in the document.

3.3.4. Predictive Model

Predictive model produces the results in accordance to the predictions that represent the probability of the outcome based on estimated significance from a set of input variables. A predictive model is trained on the given features and their significance, for this purpose Kaggle fake news datasets [13] are used to train the model. Data processing techniques punctuation removal, removing stopwords, filtering using WordNetLemmatizer and count vectorizer is used to vectorize the data and Term Frequency – Inverse Document Frequency is performed for feature extraction on the given dataset and then using machine learning algorithms a predictive model is developed.

3.3.5. Predicting results

To predict results, extracted features are passed through a trained predicting model, that model compare the extracted features of the user entered news with the features on the which the predictive model is trained then by analyzing patterns of the given information to find the likeliness of the toward which category (fake/real) and then return the outcome of the prediction.

3.4. Non-Functional Requirements

Operational capabilities and constraints of the system that positively enhance the functionality of the system depends on non-functional requirement of the system. The non-functional requirements contain the high performance of tools, the accuracy of the structure, usability, speed, the consistency of its GUI, and its effects on software. Nonfunctional requirements are recognized as quality attributes. All the non-functional requirements which are playing their role are:

3.4.1. Reliability

The reliability of a system depends on frequency of the failure and the chances of the recovery. FNDS is a web application possible failure will be website hosting server and if it fails could be recovered easily in short time.

3.4.2. Performance

When many tasks are done by the device accurately and properly it is called the performance of that device. The performance is speedy because nothing is processed at the backend it is very easy to use and easy to handle.

3.4.3. Speed

System response time is directly dependent on the end user internet speed.

3.4.4. Availability

This system is a web-based application it has no restriction of time. It can be accessed at any time and at any device with the availability of internet and capable browse.

3.4.5. Security

No special security measure is taken to secure or to recover this application as this is mostly used to get information from the user and predict that information and there is no way to manipulate the predictive model.

Chapter 4

SYSTEM MODELING AND DESIGN

This chapter will cover the System Design and Analysis of Fake News Detection System, system design briefly describes as the blueprint for constructing the software. The system design provides a high level of abstraction, a level that can present the objective and more detailed data, functionalities, and behavior of the system requirements. Moreover, further refinement leads to a much lower level of abstraction, that can still trace back to the requirements, but in more detailed connections. It is very important to create a design using recognizable architectural styles and patterns that help to understand the design correctly.

By keeping these points in consideration this chapter is organized as the first use-case for identifying what should be outside the system and what are the responsibilities of the system to perform, then activity diagram to graphically represent the use-case and the flow of interaction, then data flow diagram to represent the flow of the data of a process or of the system, then sequence diagram represents how events cause transitions from one object to the other, then design class diagram shows the interaction of different classes or instances, and finally architectural diagrams.

4.1. System Design and Analysis.

System Design and Analysis refers to the methods of examining the situation of the software with the intention of improving it through better procedures and methods. System Design and Analysis relate to improving performance, shaping organizations, and achieving objectives for achieve best outcomes from the system. The emphasis is on systems in action, their contribution and the relationships between the subsystems of the main system to meets the goal.

The system analysis of any system is a detailed study understand the different operations and functions performed by a system to achieve the desired results and the relationships between the functions and the operations of the systems to support the smooth functioning of the system. A key question is, what must be performed to solve the problem? Once the system analysis is completed the analyst has a firm understanding of what have to be done. After understanding the system thoroughly now have to determine how the problem could be solved. In system design, analyst move from the logical to the physical aspects of the system’s life cycle.

The system design of any system describes the developing phase as well as the whole system’s working and workflow the system. This phase of the system is the most creative, challenging, and time-consuming process of system development.

This chapter consist the implementation of the different software engineering models, used to perform a better understanding of the system working. The model discussed in this chapter is Unified Modeling Language (UML), UML is a general-purpose modeling language that intended to provide a visual implementation of the flow of the system and related process in the field of software engineering. Scenario-based modeling to examine the features of the system instead of predicting the outcome of the system. In scenario-based modeling, system use the input or scenarios to identify the possible outcomes of the designed system.

4.2. Use Case Diagrams

The use case is the methodology used to organize and identify functional requirements of the software project, use case also helps to explain the system behavior and its processes, use case summarizes the core functionalities and represents the user interaction with the system, which may help while developing the system.

Use case diagram consist of an actor and the project environment consists of different processes. Processes connected with the actor are the processes or functionalities of the project that need actor assistant to perform, processes that does not connected to the actors but connected to the processes that are connected to the actors are the supportive processes, those processes support the actor's connected processes to complete their tasks efficiently and accurately. For the Fake News Detection System use case is designed which is shown in Figure 4.1. Fake News Detection System use case represents user as the actor and user need to input title, author, and article of the news and if any field remains incomplete "Incomplete data" process interferes and if all fields are filled by the user, data forwarded to "Data cleaning" process to clean raw data collected from the user. Then forwarded to the predictive model to predict if the entered news is fake or real.

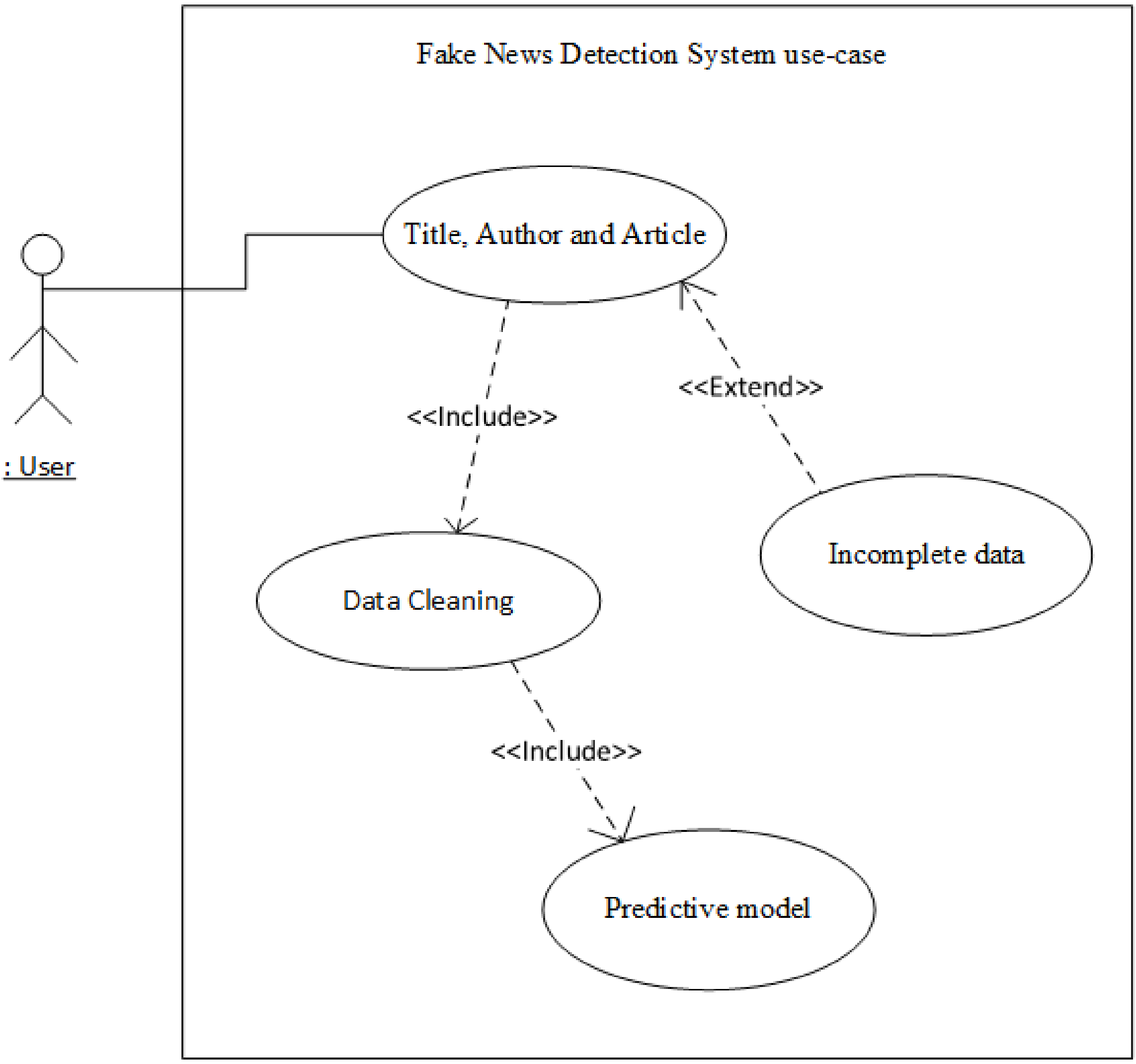


Figure 4. 1: News prediction use case diagram

4.3. Full Dress Use Case

To better understand the use case scenario, full dress use case or detailed use case are designed in detailed where use case answers the following questions: who is the primary actor(s) and the secondary actor(s)? what are the goals of actor? what pre-conditions should exist before the story begins? what tasks or operations are performed by the actor? what exceptions could be considered? what variations in the actor’s interaction are possible? what information in the output does the actor gets from the system?

As illustrated in Figure 4.1, use case diagram of Fake News Detection System shows the user interaction with the system and to understand the whole scenario a full dressed or detailed use case is provided in Table 4.1.

Table 4. 1: Full Dress Use Case of News prediction

Use Case Selection	Comment
Use Case name	Predict news either real or fake
Primary Actor	User
Level	User Goal
Stakeholders and interests	User who wants to classify a news
Pre-conditions	User must have the internet and access to the system
Success Guarantee	User have to fill all the fields to classify the news
Main Success Scenario	User have the access of the system and have enter the news's details and system predict the news
Extensions	User doesn't have the access of the system; user may not fill all fields
Special Requirements	Systems overall performance depends on the user internet connection

4.4. Activity diagram

The activity diagram is much like the flowchart diagram, a graphical form for representing the control flow sequences and logic. The activity diagram is a high-level to better understanding the functionalities of the system, that can also be used for modeling the workflow, the business requirements, and later be used for investigating the business requirements. In simple words, an activity diagram is a flowchart to graphically represent the flow of processes from one activity to another activity within a specific system.

An activity diagram consists rounded corner rectangles to represent the activity or the function of the system, arrows are used to represent the flow of data between those activities or the functions of the system, the diamond shape is used to represent the decision block.

To better understand the Fake News Detection System an activity diagram is designed for this system shown in Figure 4.2. The activity starts by entering the title of the news, author, and article in their desire fields in not entered correct user get prompt from the system, then the entered data forwarded for cleaning and for applying preprocessing algorithms to preprocess the collected data, then the preprocessed data pass through the predictive model to make a prediction on the data, and then the result of the prediction is displayed to the user.

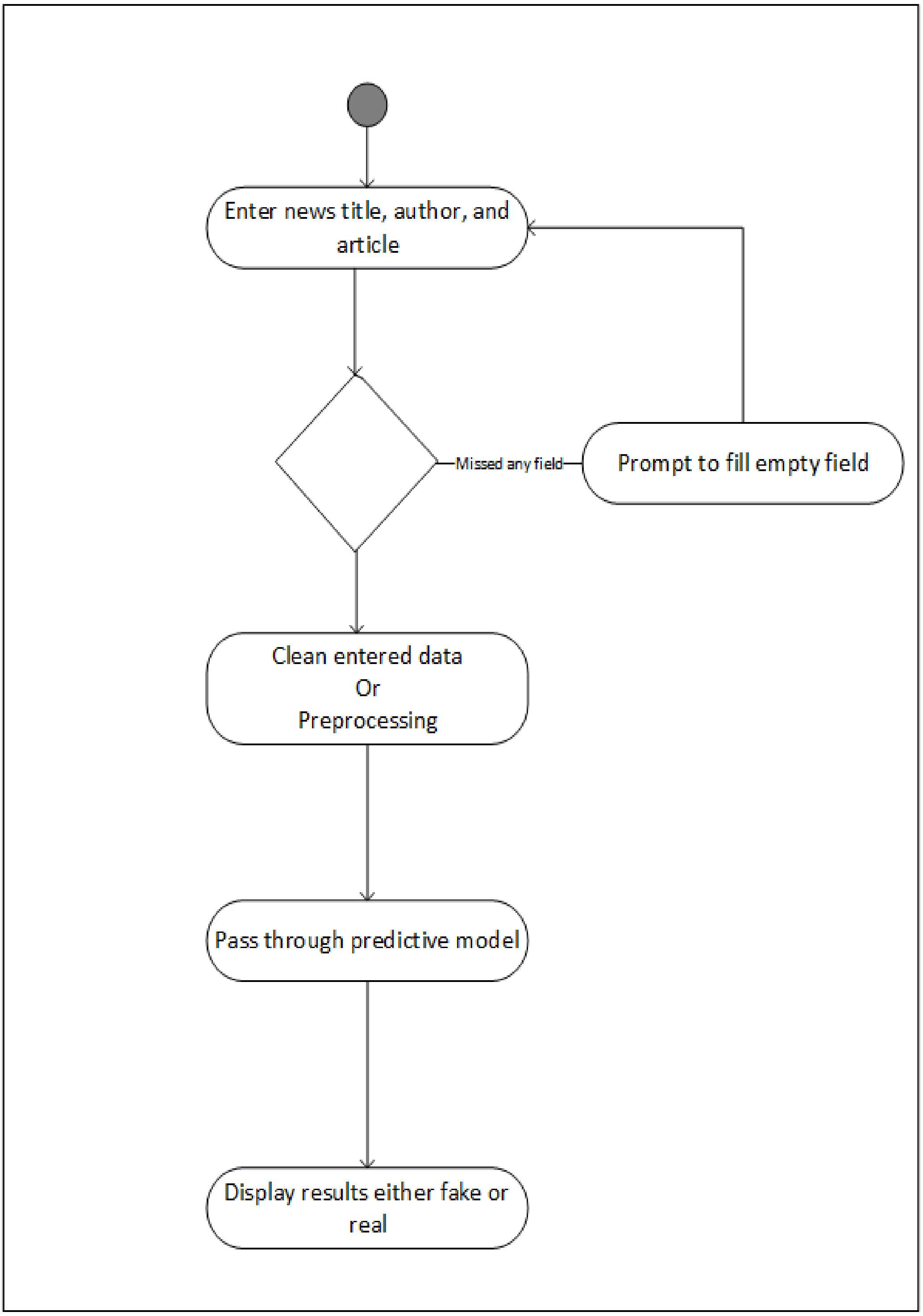

Enter news title, author, and article
Missed any field
Prompt to fill empty field
Clean entered data Or Preprocessing
Pass through predictive model
Display results either fake or real

Figure 4. 2: Activity Diagram

4.5. Data Flow Diagram

A data flow diagram is a way to graphically represent the data flow through the processes of the system, it provides information about the inputs and the outputs of the process or the function of the system. Data flow modeling is the most important modeling activity in structured analysis. DFD (Data flow Diagram) is the overview of the data from input to process and then to the output, data objects represented as the labeled arrows lines, and the transformations are represented by the circles, also known as bubbles. DFD consists of different levels as the level gets higher more detailed DFD is formed. For level 0 whole system must be shown as a single bubble, gradually level by level more details must be added. In this section of the chapter, DFDs for the FNDS are designed from level 0 as shown in Figure 4.3 to level 1 as shown in Figure 4.4.

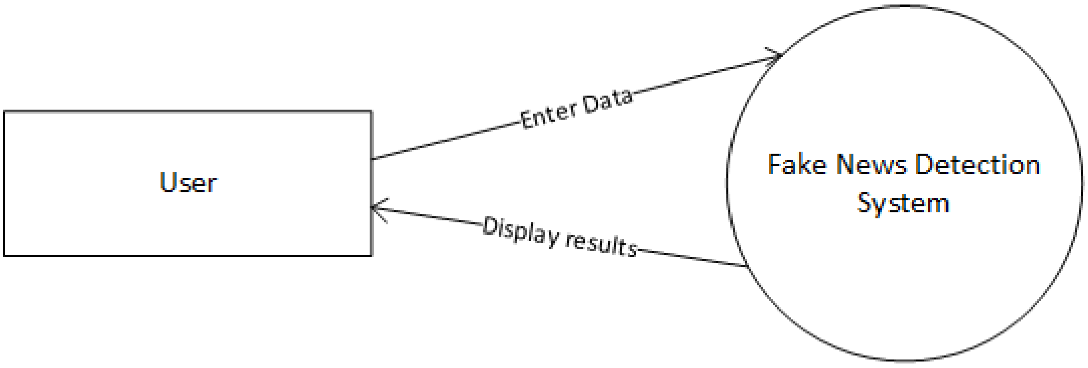


Figure 4. 3: Data flow Diagram Level 0

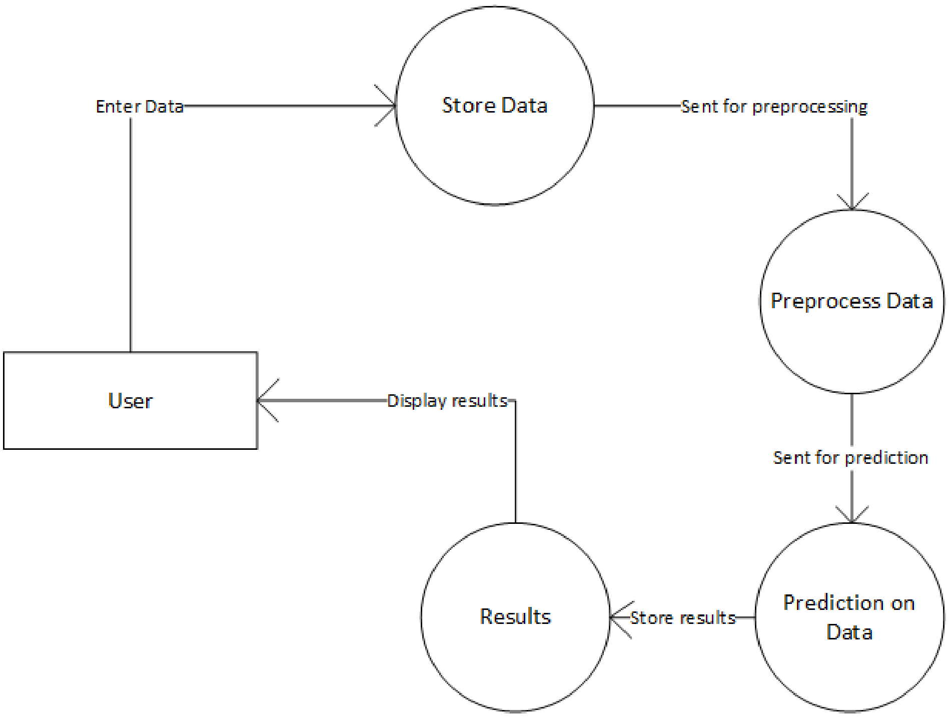


Figure 4. 4: Data flow Diagram Level 1

4.6. System Sequence Diagram

The system sequence diagram represents the behavior of the system with respect to the actor’s action. It illustrates the transitions of the event from one object to another. Events are identified by examining a use case, the designer design a sequence diagram with respect to the use case. A representation of how events occurred during the normal process from one object to the other. It represents key classes of the system and the events that occurred from class to class. For FNDS a system sequence diagram is designed as a user access the web interface enters the news details then forwarded to data cleaning then for prediction at the end displays the results on the web interface, as shown in Figure 4.5.

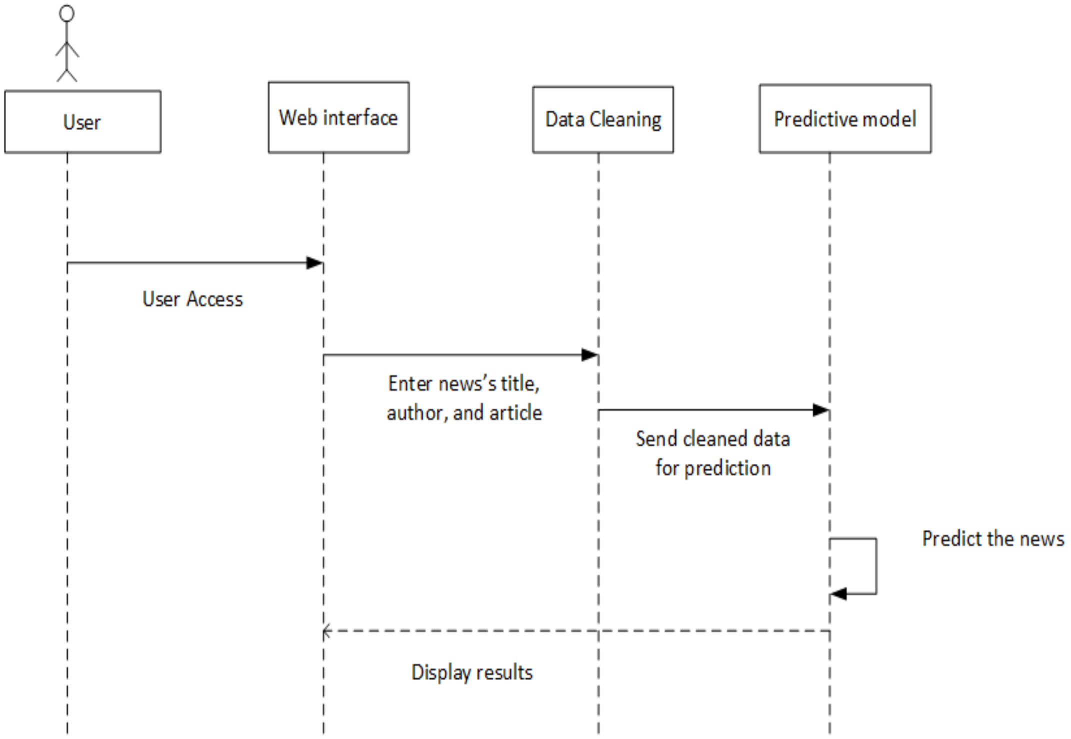


Figure 4. 5: System Sequence Diagram

4.7. Sequence Diagram

A sequence diagram is the illustration of the input and output events of the system. Typically sequence diagrams are related to the use case of the system visualized in the logical view. In simple words, sequence diagrams are the diagrammatic view of the full-dress use cases. For FNDS sequence diagram is shown in Figure 4.6.

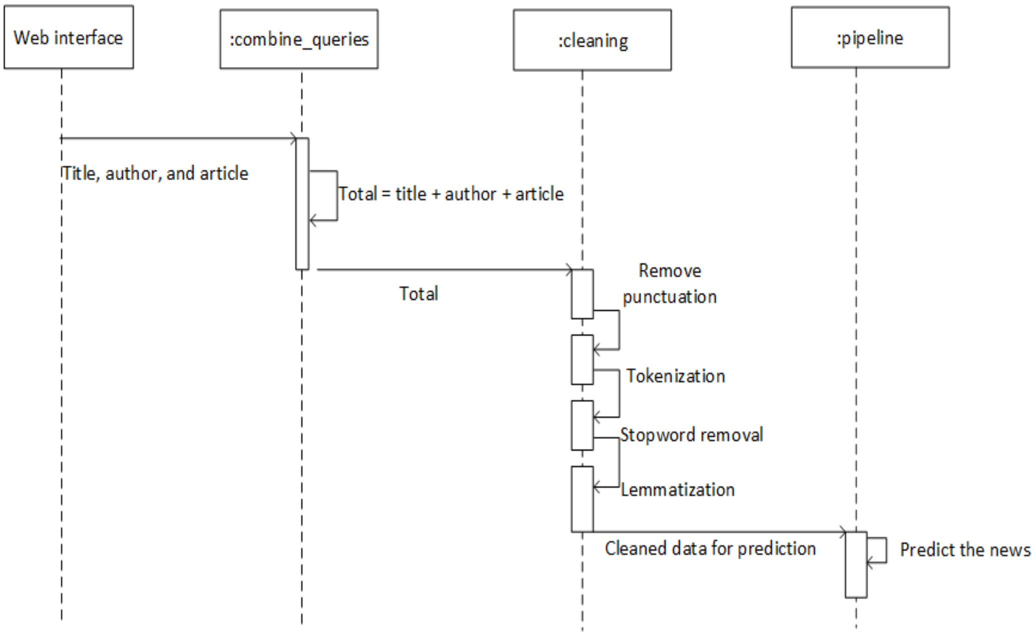


Figure 4. 6: Sequence Diagram

4.8. Architectural Diagrams

Architectural diagrams are the blueprints for any system that has to be developed. Architectural diagrams are designed to provide the semantics and the syntax for describing the software architecture. It is a very important diagram because it provides an overview of the physical deployment of the software and its lifecycle during the development phase. Building an architecture of any software system is a complex and challenging job. By developing single diagram to represent the whole architecture is very confusing and complicating to overcome this problem several related systems are developed and discussed in following sections.

4.8.1. System Context Diagram

A System Context diagram may be a best place to begin for visualizing and documenting a software system, it helps to visualize the bigger picture. A simple diagram showing the system as a box, surrounded by its users and the other components of the systems that it interacts with. As for FNDS system context diagram is shown in the Figure 4.7.

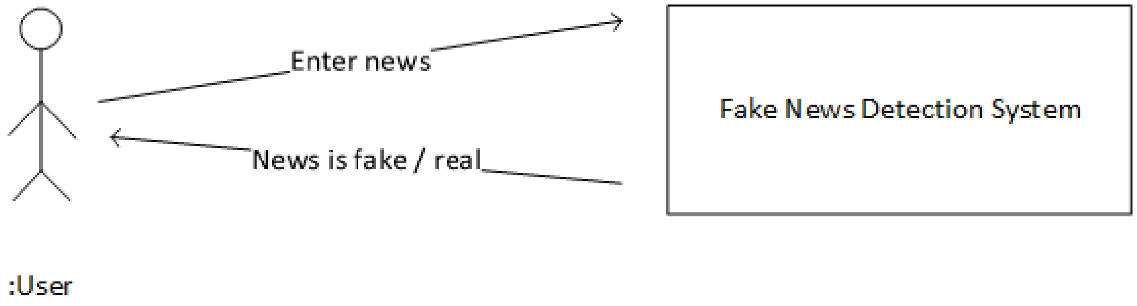


Figure 4. 7: Context Diagram

4.8.2. Container Diagram

The Container diagram shows the high-level image of the software architecture and represents how the responsibilities are distributed across it. It also shows how the containers communicate with one another. It is a simple, high level diagram that is useful for software development and its testing. As for FNDS Figure 4.8 below shows the Container diagram.

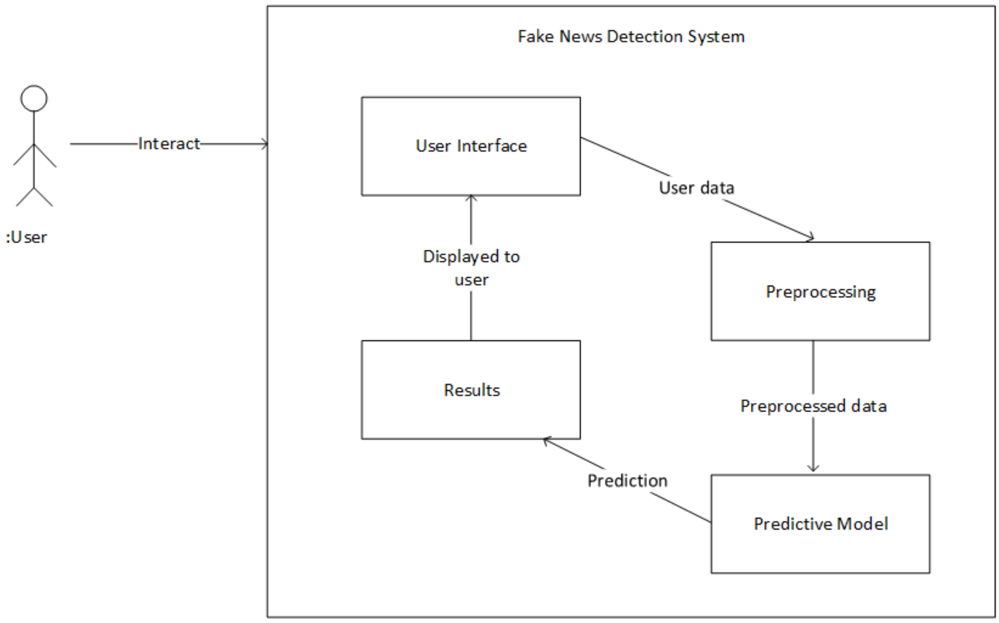


Figure 4. 8: Container Diagram

4.8.3. Component Diagram

The Component diagram shows how a container is created from a number of "components", it defines the aim of those components as, what are their responsibilities, and their implementation details.

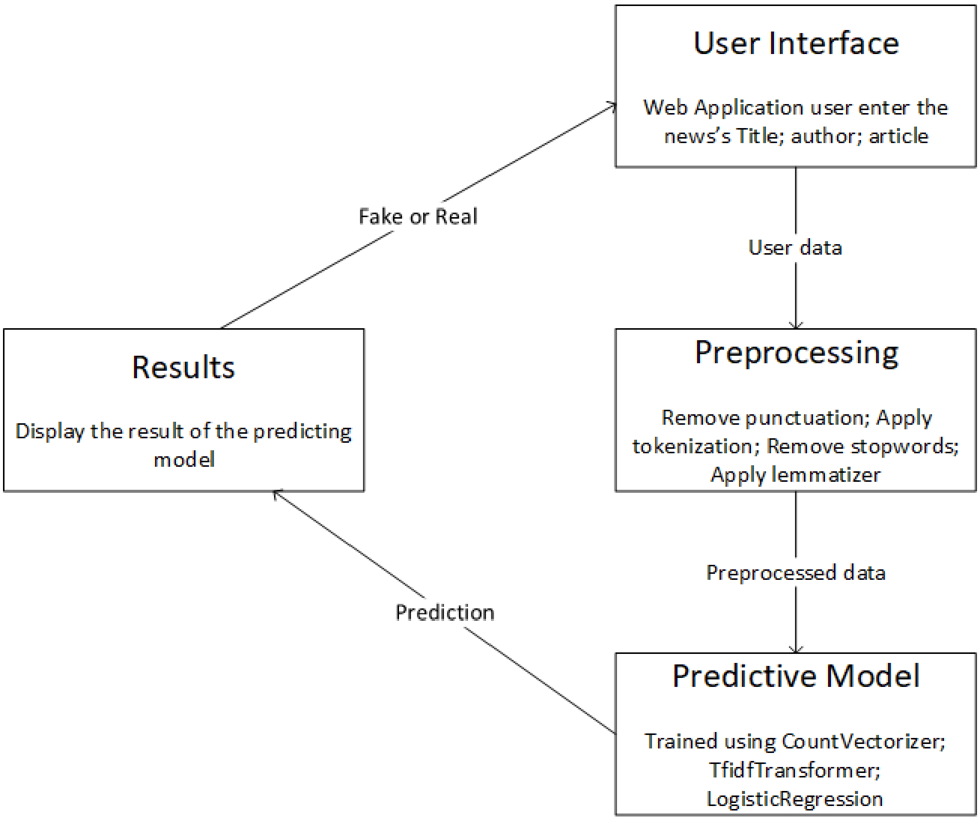

User Interface
Web Application user enter the news's Title; author; article
User data
Fake or Real
Results
Display the result of the predicting model
Preprocessing
Remove punctuation; Apply tokenization; Remove stopwords; Apply lemmatizer
Preprocessed data
Prediction
Predictive Model
Trained using CountVectorizer; TfidfTransformer; LogisticRegression

Figure 4. 9: Component Diagram

Chapter 5

SYSTEM TESTING AND VALIDATION

To evaluate the system intense testing and evaluation techniques are considered. Testing a system is not a onetime activity, testing is performed from the beginning at the component level and continue outward to the system level. This chapter will cover the testing of the FNDS and discuss the different testing techniques that are applied to test the whole system at different stages of development and deployment.

5.1. System testing

System testing is one element of a broader topic that's often stated as verification and validation (V&V) [12]. Testing functions with respect to their requirements and the consistency of the required results. The main aim to test an application is to make sure the system works flawlessly and working in the error free environment. After successfully test the different components separately it is mandatory to verify and validate the whole system by performing the system testing.

The functional behavior of the system is evaluated by application testing. Usually, system testing is carried out to measure the outcomes of the system, effectiveness of the implemented functions, behavior to environment, accuracy in the results, and overall performance of the system. In FNDS application testing, it is found that the FNDS takes the title, author, and article of the news through a web interface from the user, then clean and preprocess the data, then pass through the predictive model to make a prediction on the data and displays the results to the user on the web interface.

5.2. Testing Techniques

To evaluate the system different testing techniques are used. Generally testing techniques can be categorized into two different categories static and dynamic testing [7]. Static testing tends more towards manual testing which is a slow process. It is done in the early stages of the system life cycle either by the analyst, developer, and testing team. On the other hand, dynamic testing tends more towards automatic testing, automatic testing technique needs the script on the testing tool.

Testing involved in every stage of the system life cycle, and at each stage of the system development testing is different in nature. Testing starts from the unit testing after that integration testing, then the system testing, and at the end acceptance testing. For FNDS testing phase starts with unit testing to test different modules separately, after successfully applying unit testing, the testing phase moved forward to the integration testing applied after integrating different modules, then the system testing is done by using white box testing.

5.2.1. GUI Testing

To make sure GUI (Graphical User Interface) is working as per required GUI testing is very important to perform as user only way to interact with the system, GUI testing is also performed to validate it is working according to those requirements of the system.

Usually, GUI of the system is the only component of the system which have the direct interaction with the user as in FNDS it is the only component interacting with the end-user. So, easy navigation and predictive response must be needed for a better user experience. It is very important to test GUI if it is working respectively to the system requirements.

In the FNDS case, GUI consists of three text fields and a submit button. The testing of this GUI is very simple, either field getting data from the user and button successfully submitting the entered data to the system. For FNDS GUI see Appendix-I Figure AP-01

5.2.2. Unit Testing

Unit is the smallest component of the big picture of the system, in unit testing mostly a small collection of code lines is tested. It supports integration and system-level testing. The unit testing mostly referred to type of testing called white box testing. Developers perform unit testing.

Training a predictive model is a very precise and time-consuming process and it is very crucial to keep the code free from logical error, so it is very important to keep up testing with every small portion of the code.

For FNDS, unit testing is carried out by testing every small module and verifying its outcomes with the predefined outcomes. For reference in FNDS model training, there are datasets to train the predictive model, to remove null values from the datasets, function called fillna () is used, to verify the output of this function another function, called isnull () is used.

5.2.3. Component Testing

To separately test the functionalities and the features of the system, component testing is applied after the unit testing. Before performing integration testing it is always recommended to perform component testing first. Component testing mostly referred to the black box testing. Testers perform component testing.

Component testing validates the test requirements or test cases. While building multiple small units of a bigger picture, it is important to verify the functionalities of those components.

For FNDS component testing is performed on multiple components of the system, for reference data cleaning component is a very important component of the machine learning based projects as the system manipulating and managing data and the performance of the system depends on the data. To verify if the system performing data cleaning as it is designed or is it lacking somewhere, component testing is performed.

5.2.4. Integration Testing

Integration testing is performed when two or more tested components of the system combined to perform tasks.

As the single component cannot ensure that the bigger picture will work as it is required to be. To ensure combined components working according to the design and specifications, integration testing is performed on combined components.

For FNDS, there are several components that must integrate with each other to perform a specific task of the system. Components of FNDS that are integrated together are predictive model with the main code of the system and the interaction of the web interface with the main program of the system. Integration testing is performed to ensure the working of those modules with each other.

5.2.5. System Testing

System testing is the last phase of the testing in the development life cycle of any system, system testing makes sure that each and every function and features of the whole system are functioning as they are designed to and according to the specified requirements of the system.

On successfully passing all the unit testing, component testing, and integration testing, the time comes to testify the overall system according to the specified requirements. To make sure all the features and the functionality of the FNDS working to the set standards it is very important and essential phase of testing.

There are three main components of the FNDS; GUI, program to preprocess the user data, and the predictive model, the data travel path between them is GUI get data from the user, sends to the preprocessing program, after preprocessing data passed through the predictive model and displayed the results to the user on the GUI so their successful communication is very important to accomplish the goals of the whole system.

5.3. Test Cases

A test case is the documentation of the test performed on the particular function or feature of the system. A test case is a specific input scenario based on the requirement of the function or component of the system that will be tested. The tester has to perform procedural steps to get the required results from the system. After performing the test, tester then verifies the results with the expected results of the system as defined in the requirement. This section of the chapter consists of several test cases performed on the FNDS, to verify the different functionalities of the developed system in reference to the requirements of the system.

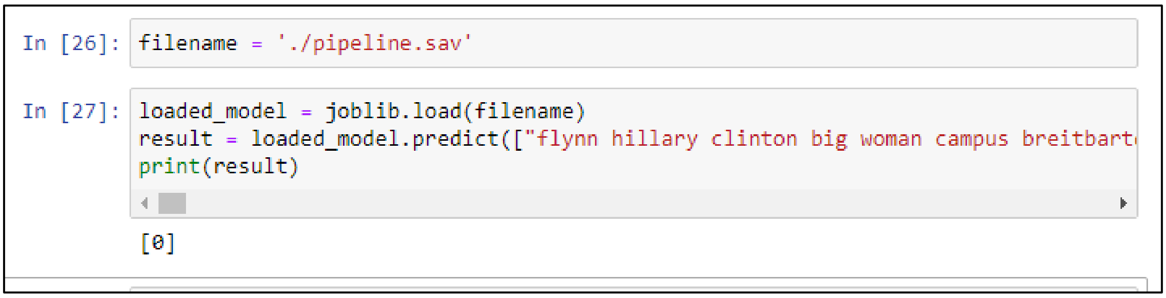
5.3.1. Test Case1: Prediction of the news

As described in Section 3.3.4, predictive model is the main component of the FNDS that classifies the news, so to verify it is working this test is performed on the predictive model. Table 5.1 shows the documented test case of this test.

Table 5. 1: Test Case 1

GENERAL INFORMATIN			
Test Stage:	☐ Unit ☑ Functionality ☐ Integration ☐ System		
Test performed on:	01/23/2021	System Date:	Not applicable
Tester:	Muhammad Ammaar Akhtar	Test Case number:	TC-01

Test case Description:	This test case is for predictive model		
Results:	☑ Pass ☐Fail	Incident Number:	
INTRODUCTION			
Requirement to be tested:	Prediction of the news as described in section 3.3.4 of this report		
Roles and Responsibilities:	Muhammad Ammaar Akhtar developer and tester of FNDS		
Set Up Procedures:	To perform this test tester, require news in its preprocessed form		
Stop Procedures:	To terminate this test a function is there to display the result		
ENVIRONMENTAL NEEDS			
Hardware:	Computer, or laptop		
Software:	Browser, Jupiter notebook, python installed on system		
Procedural Requirements:	Tester have to load preprocessed news in loading function and run the cell		
TEST			
Test Items and Features:	This test case identifies the predictions of the predictive model		
Input Specifications:	Output of the feature extraction as described in section 3.3.3 is the input for this test case		
Procedural Steps:	Code file must be running on Jupiter notebook, function is available in the file, function loaded the predictive model, tester inputs the preprocessed news to the function, tester runs the cell, function pass the input through the predictive model then displays the result of the model to tester		
Expected Results of Case:	Either 1 or 0		
ACTUAL RESULTS			
Output Specifications:	If the test produce 1 in this case news is fake and if output is 0 the news is real for better understanding see Figure 5.1 and 5.2, Figure 5.1 shows the result of real news and Figure 5.2 shows the result of fake news		

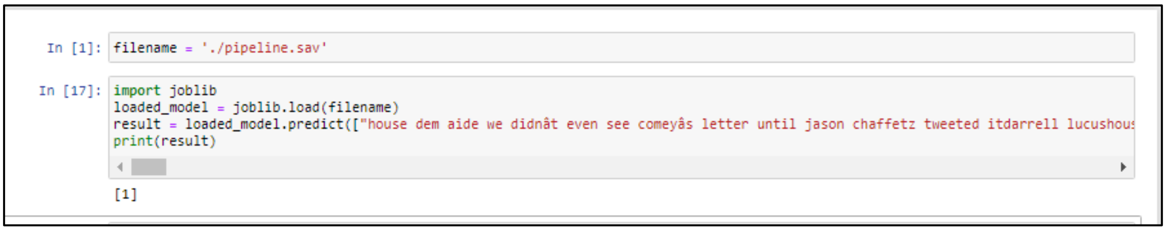


```
In [26]: filename = './pipeline.sav'

In [27]: loaded_model = joblib.load(filename)
         result = loaded_model.predict(["flynn hillary clinton big woman campus breitbart
         print(result)

         [0]
```

Figure 5. 1: Prediction result of real news



```
In [1]: filename = './pipeline.sav'

In [17]: import joblib
         loaded_model = joblib.load(filename)
         result = loaded_model.predict(["house dem aide we didnât even see comeyâs letter until jason chaffetz tweeted itdarrell lucushous
         print(result)

         [1]
```

Figure 5. 2: Prediction result of fake news

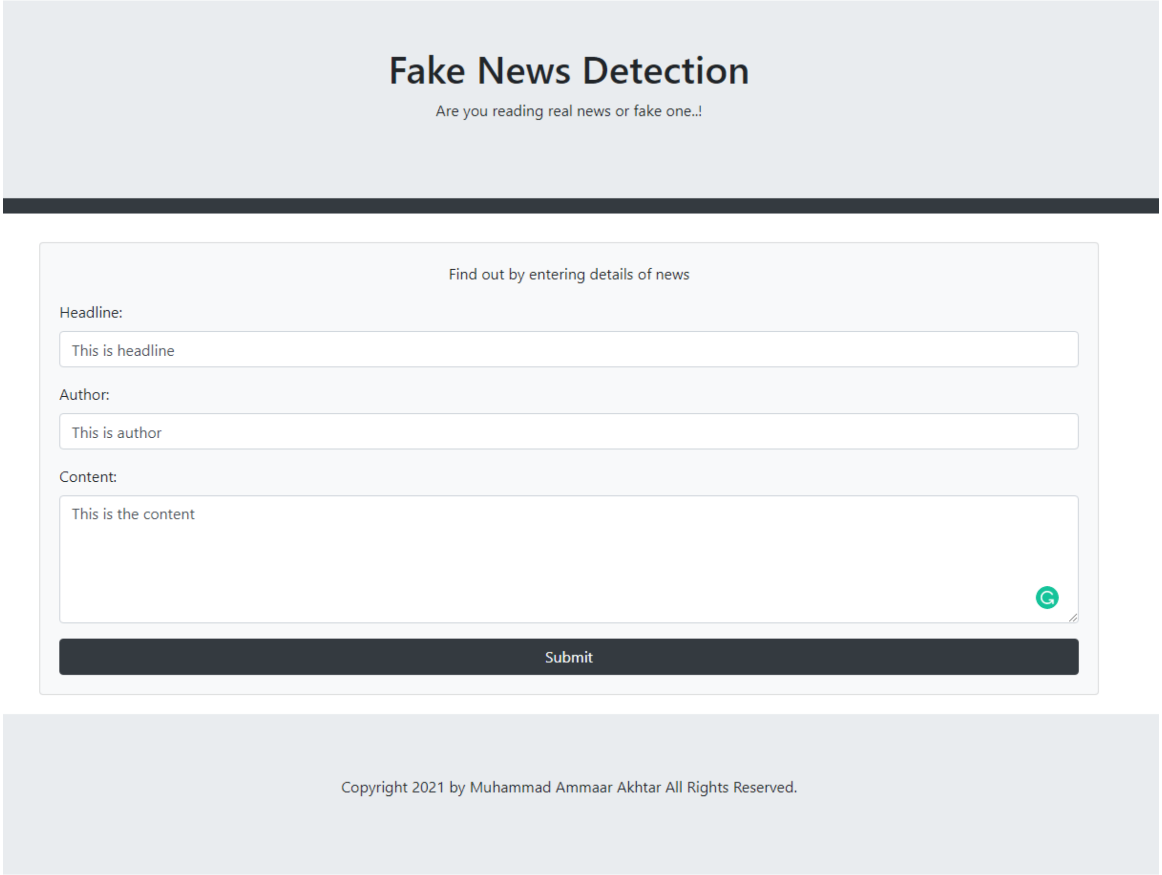
5.3.2. Test Case2: Integration of GUI and the program code

This test is performed after the integration of the GUI with the code to get data from HTML form into the program, to perform further functionalities on the acquired data. The documentation of this test case is shown below in Table 5.2.

Table 5. 2: Test Case 2

GENERAL INFORMATIN			
Test Stage:	☑ Unit ☐ Functionality ☑ Integration		
Test performed on:	03/06/2021	System Date:	Not applicable
Tester:	Muhammad Ammaar Akhtar	Test Case number:	TC-02
Test case Description:	This test case is testing the integration of the GUI with the program code		
Result:	☑ Pass ☐Fail	Incident Number:	
INTRODUCTION			
Requirement to be tested:	GUI and user data acquired to access in code, for reference see section 3.2.1 and 3.2.6		
Roles and Responsibilities:	Muhammad Ammaar Akhtar developer and tester of FNDS		

Set Up Procedures:	To perform this test tester, require to enter news detail in GUI and access in program code
Stop Procedures:	To terminate this test, print the accessed data from the GUI
ENVIRONMENTAL NEEDS	
Hardware:	Computer, or laptop
Software:	Browser, GUI, python installed on system and PyCharm
Procedural Requirements:	Tester have to write the news title, author and article of the news in their respective field
TEST	
Test Items and Features:	This test case identifies that data obtained from the GUI and accessed in the program code
Input Specifications:	Title of the news, author of the news, and the article of the news.
Procedural Steps:	Tester enter the title, author and article of the news in the required field on the GUI of FNDS and submit to the program code for further processing, for confirmation on the program file print the received data as output
Expected Results of Case:	Successfully print the received data on the console
ACTUAL RESULTS	
Output Specifications:	For successfully pass this test program have to display the received data on the console to verify if the data received from the GUI for reference see Figure 5.3 and 5.4



Fake News Detection

Are you reading real news or fake one..!

Find out by entering details of news

Headline:

This is headline

Author:

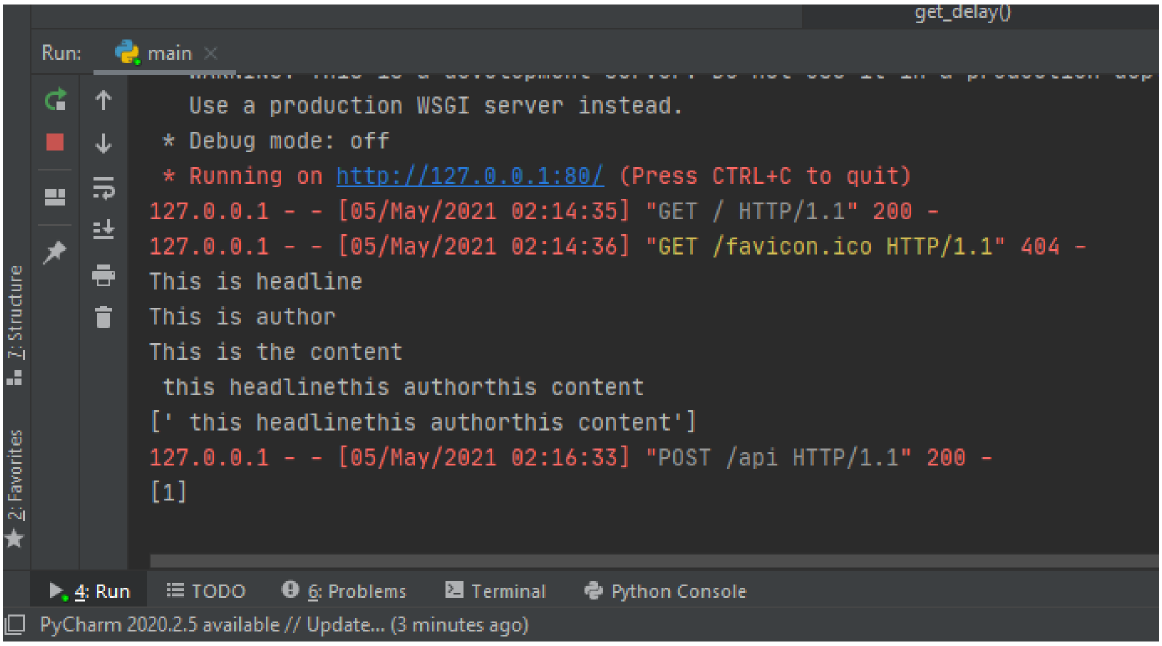
This is author

Content:

This is the content

Copyright 2021 by Muhammad Ammaar Akhtar All Rights Reserved.

Figure 5. 3: Text entered in the fields



```
   WARNING: This is a development server. Do not use it in a production dep
   Use a production WSGI server instead.
 * Debug mode: off
 * Running on http://127.0.0.1:80/ (Press CTRL+C to quit)
127.0.0.1 - - [05/May/2021 02:14:35] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [05/May/2021 02:14:36] "GET /favicon.ico HTTP/1.1" 404 -
This is headline
This is author
This is the content
 this headlinethis authorthis content
[' this headlinethis authorthis content']
127.0.0.1 - - [05/May/2021 02:16:33] "POST /api HTTP/1.1" 200 -
[1]
```

Figure 5. 4: Received data displayed on console

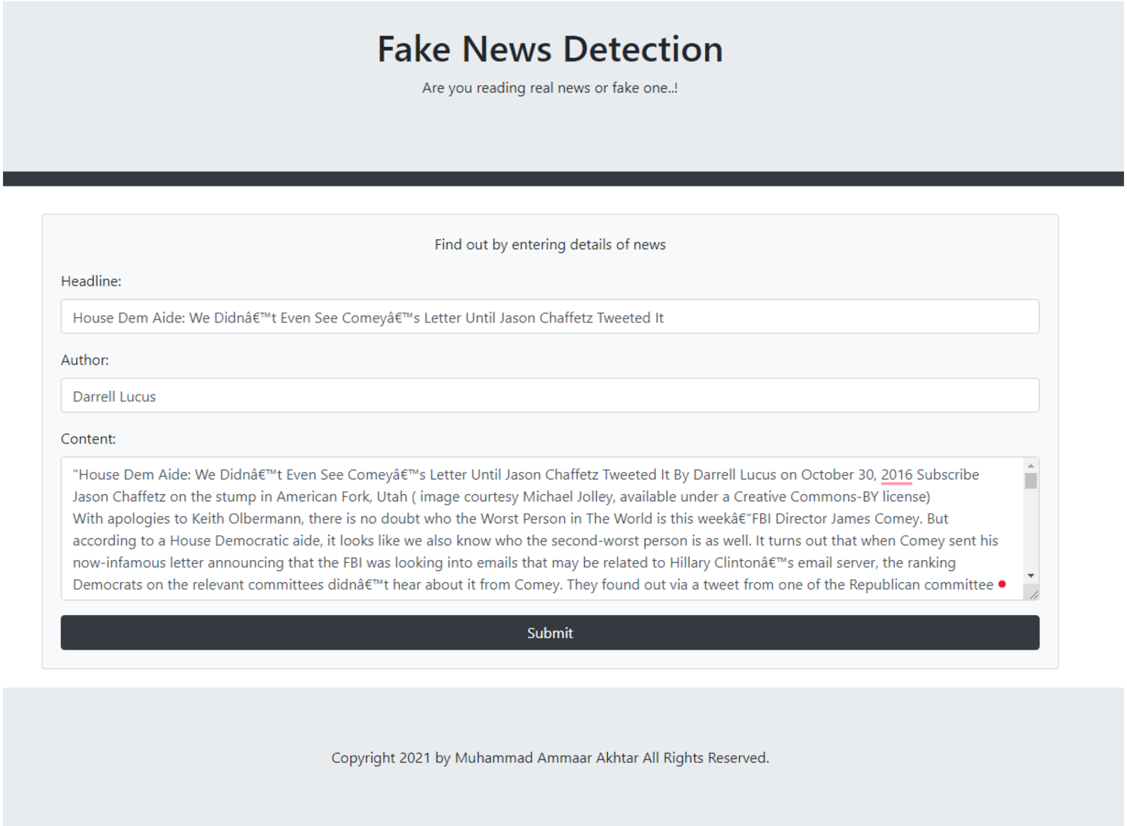
5.3.3. Test Case3: Predicting news from GUI to verify the system

After integrating all the components and interface of the FNDS system test is performed, as all the individual components and the features of the FNDS are working flawlessly and as per their requirements. The documented test case of this test is shown below in Table 5.3.

Table 5. 3: Test Case 3

GENERAL INFORMATIN			
Test Stage:	☑ System ☐ Interface ☐ Performance		
Test performed on:	04/24/2021	System Date:	Not applicable
Tester:	Muhammad Ammaar Akhtar	Test Case number:	TC-03
Test case Description:	This test case is Predicting news from GUI to verify the system		
Result:	☑ Pass ☐Fail	Incident Number:	
INTRODUCTION			
Requirement to be tested:	Predicting results as mentioned in the section 3.2.5		
Roles and Responsibilities:	Muhammad Ammaar Akhtar developer and tester of FNDS		
Set Up Procedures:	To perform this test, tester require to enter news detail in GUI and submit		
Stop Procedures:	To terminate this test, system display the result		
ENVIRONMENTAL NEEDS			
Hardware:	Computer, or laptop		
Software:	Browser		
Procedural Requirements:	Tester have to write the news title, author and article of the news in their respective field and submit for prediction		
TEST			
Test Items and Features:	This test case identifies that system successfully predicting the news		

Input Specifications:	Title of the news, author of the news, and the article of the news.
Procedural Steps:	Tester enter the title, author and article of the news in the required field on the GUI of FNDS and submit to the system for prediction
Expected Results of Case:	Successfully print the prediction results
ACTUAL RESULTS	
Output Specifications:	On successfully pass this test system displays the result of the prediction for reference see figure 5.5 and 5.6



Fake News Detection

Are you reading real news or fake one..!

Find out by entering details of news

Headline:

House Dem Aide: We Didnâ€™t Even See Comeyâ€™s Letter Until Jason Chaffetz Tweeted It

Author:

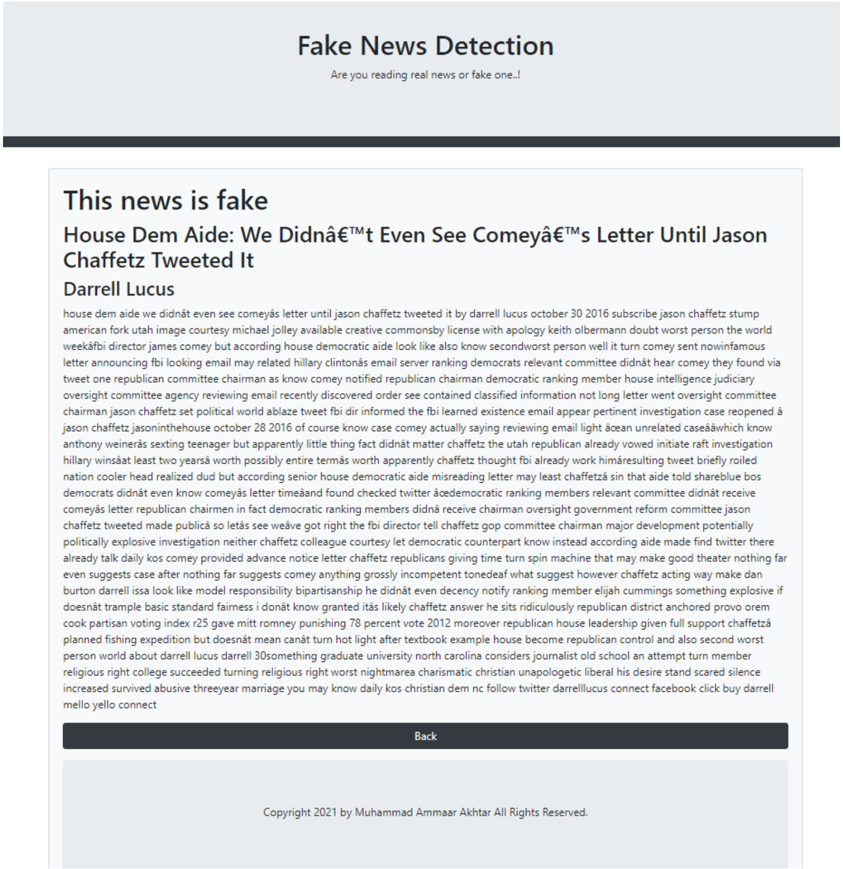
Darrell Lucus

Content:

"House Dem Aide: We Didnâ€™t Even See Comeyâ€™s Letter Until Jason Chaffetz Tweeted It By Darrell Lucus on October 30, 2016 Subscribe Jason Chaffetz on the stump in American Fork, Utah (image courtesy Michael Jolley, available under a Creative Commons-BY license)
With apologies to Keith Olbermann, there is no doubt who the Worst Person in The World is this weekâ€“FBI Director James Comey. But according to a House Democratic aide, it looks like we also know who the second-worst person is as well. It turns out that when Comey sent his now-infamous letter announcing that the FBI was looking into emails that may be related to Hillary Clintonâ€™s email server, the ranking Democrats on the relevant committees didnâ€™t hear about it from Comey. They found out via a tweet from one of the Republican committee

Copyright 2021 by Muhammad Ammaar Akhtar All Rights Reserved.

Figure 5. 5: News entered in the fields



Fake News Detection

Are you reading real news or fake one..!

This news is fake

House Dem Aide: We Didnâ€™t Even See Comeyâ€™s Letter Until Jason Chaffetz Tweeted It

Darrell Lucus

house dem aide we didnât even see comeyâs letter until jason chaffetz tweeted it by darrell lucus october 30 2016 subscribe jason chaffetz stump american fork utah image courtesy michael jolley available creative commonsby license with apology keith olbermann doubt worst person the world weekâfbi director james comey but according house democratic aide look like also know secondworst person well it turn comey sent nowinfamous letter announcing fbi looking email may related hillary clintonâs email server ranking democrats relevant committee didnât hear comey they found via tweet one republican committee chairman as know comey notified republican chairman democratic ranking member house intelligence judiciary oversight committee agency reviewing email recently discovered order see contained classified information not long letter went oversight committee chairman jason chaffetz set political world ablaze tweet fbi dir informed the fbi learned existence email appear pertinent investigation case reopened â jason chaffetz jasoninthehouse october 28 2016 of course know case comey actually saying reviewing email light âœan unrelated caseââwhich know anthony weinerâs sexting teenager but apparently little thing fact didnât matter chaffetz the utah republican already vowed initiate raft investigation hillary winsâat least two yearsâ worth possibly entire termâs worth apparently chaffetz thought fbi already work himâresulting tweet briefly roiled nation cooler head realized dud but according senior house democratic aide misreading letter may least chaffetzâ sin that aide told shareblue bos democrats didnât even know comeyâs letter timeâand found checked twitter âœdemocratic ranking members relevant committee didnât receive comeyâs letter republican chairmen in fact democratic ranking members didnâ receive chairman oversight government reform committee jason chaffetz tweeted made publicâ so letâs see weâve got right the fbi director tell chaffetz gop committee chairman major development potentially politically explosive investigation neither chaffetz colleague courtesy let democratic counterpart know instead according aide made find twitter there already talk daily kos comey provided advance notice letter chaffetz republicans giving time turn spin machine that may make good theater nothing far even suggests case after nothing far suggests comey anything grossly incompetent tonedeaf what suggest however chaffetz acting way make dan burton darrell issa look like model responsibility bipartisanship he didnât even decency notify ranking member elijah cummings something explosive if doesnât trample basic standard fairness i donât know granted itâs likely chaffetz answer he sits ridiculously republican district anchored provo orem cook partisan voting index r25 gave mitt romney punishing 78 percent vote 2012 moreover republican house leadership given full support chaffetzâ planned fishing expedition but doesnât mean canât turn hot light after textbook example house become republican control and also second worst person world about darrell lucus darrell 30something graduate university north carolina considers journalist old school an attempt turn member religious right college succeeded turning religious right worst nightmarea charismatic christian unapologetic liberal his desire stand scared silence increased survived abusive threeyear marriage you may know daily kos christian dem nc follow twitter darrelllucus connect facebook click buy darrell mello yello connect

Copyright 2021 by Muhammad Ammaar Akhtar All Rights Reserved.

Figure 5. 6: Result displayed on submission

5.4. Non-functional requirements

Operational capabilities and constraints of the system that positively enhance the functionality of the system depends on non-functional requirement of the system. The non-functional requirements contain the high performance of tools, the accuracy of the structure, usability, speed, the consistency of its GUI, and its effects on software. Non-functional requirements are recognized as quality attributes. All the non-functional requirements which are playing their role are as:

5.4.1. Reliability

The reliability of FNDS is depends on website hosting server as FNDS is a web application possible failure will be website hosting server and if it fails could be recovered easily in less time.

5.4.2. Performance

When many tasks are done by the device accurately and properly it is called the performance of that device. The performance is speedy because nothing is processed at the backend it is very easy to use and easy to handle.

5.4.3. Speed

System response time is directly dependent on the user’s internet speed.

5.4.4. Availability

System is a web-based application it has no restriction of time. The system can be accessed at any time and at any device with the availability of internet and capable browse.

5.4.5. Security

No special security measure is taken to secure or to recover this application as this is mostly used to get information from the user and predict that information and there is no way to manipulate the predictive model.

Table 5. 4: Non-functional requirements

Property	**Measure**
Reliability	FNDS is web application possible failure will be website hosting server and if it fails could be recovered easily in less time.
Performance	The performance is very good because nothing is processed at the backend it is very easy to use and easy to handle.
Speed	System response time is directly dependent on the user’s internet speed.
Availability	System is a web-based application it has no restriction of time.
Security	No special security measure is taken to secure or to recover this application.

Chapter 6

CONCLUSION

This chapter contains an overview of the developed system, briefly discuss the key features, and goals achieved by the developed system in the first section of this chapter, and also discuss the current limitations of the system in second section, and possible improvements and enhancements in the future in the third section of this chapter.

6.1. Conclusion

Fake News Detection system (FNDS) is a web-based application that detects fake news over the internet. The system relay on a machine learning trained model to classify fake and real news. Comparing FNDS with the existing systems, FNDS doesn't depend on public or expert opinions to classify the news, which makes FNDS much more reliable than the traditional methods of detecting fake news. Main goal to develop this system is to facilitate the public to detect either the information they are getting is real or fake, and secure them from false information. There are different ways to classify the news, FNDS approach the problem with the consideration to keep system development more easy and simple. For user convenience, the Graphical User Interface (GUI) is very simple and easy to use, user has to enter the news title or headline, author of the news, and the content or article of the news in their respective fields and press the submit button. The result of the news classification displayed to the user, there is no limitation of the time or the maximum number of attempts by the user.

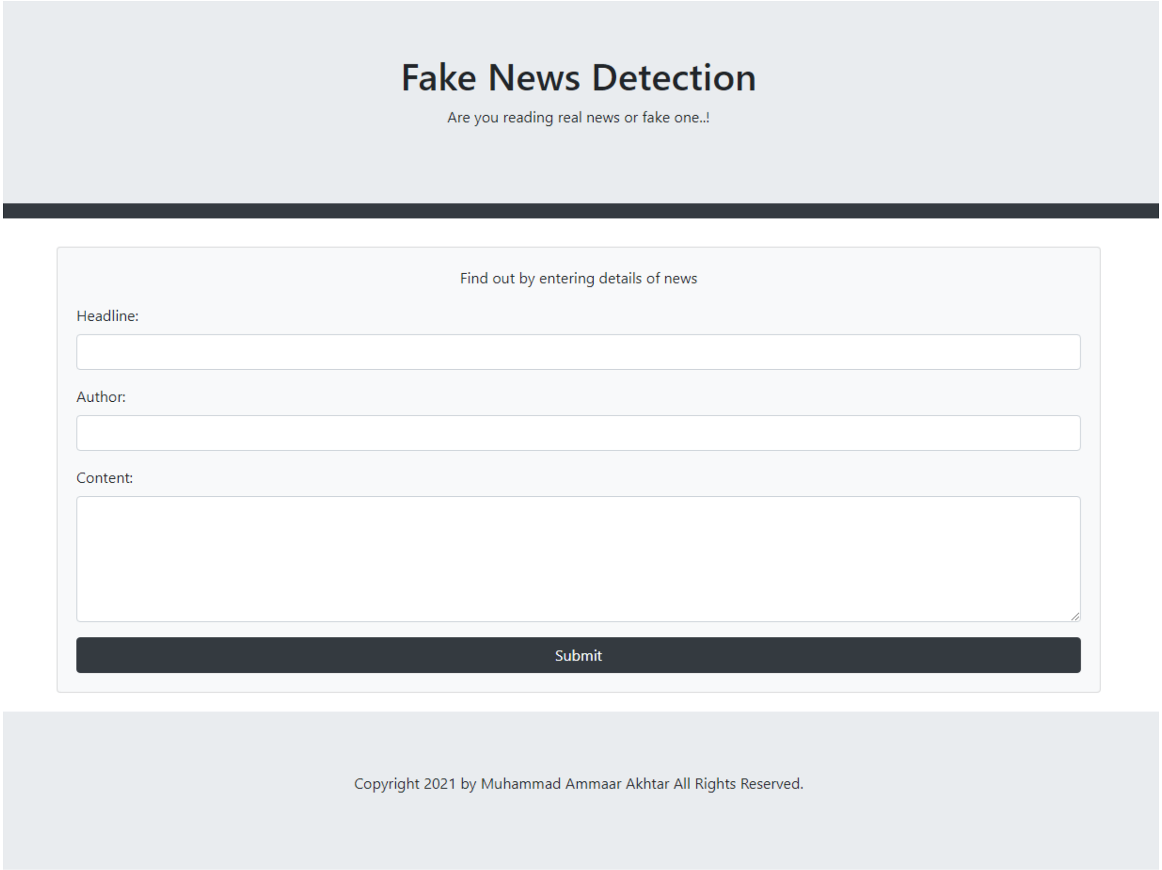
6.2. Limitations

This is a web-based system that can be accessed anywhere from any device which supports web pages. There are some limitations of this system, FNDS classify the news only if the news is in textural form. This system is not designed to work with the image and video-based news, where the graphics are included. Considering the limitation that it only classifies the textural news; it does not mean that there is no possibility to classify the graphical news in the future. There is always a possibility to add some modules to this system that can classify the graphical news.

6.3. Future Work

The other limitation is that users have to enter the news details to the respective fields on GUI, in the future it is also possible to design some browser plug-in to classify the news from the news publishing websites. By using a browser plug-in, the system will become more user-friendly and easier to use.

APPENDIX – I



Fake News Detection

Are you reading real news or fake one..!

Find out by entering details of news

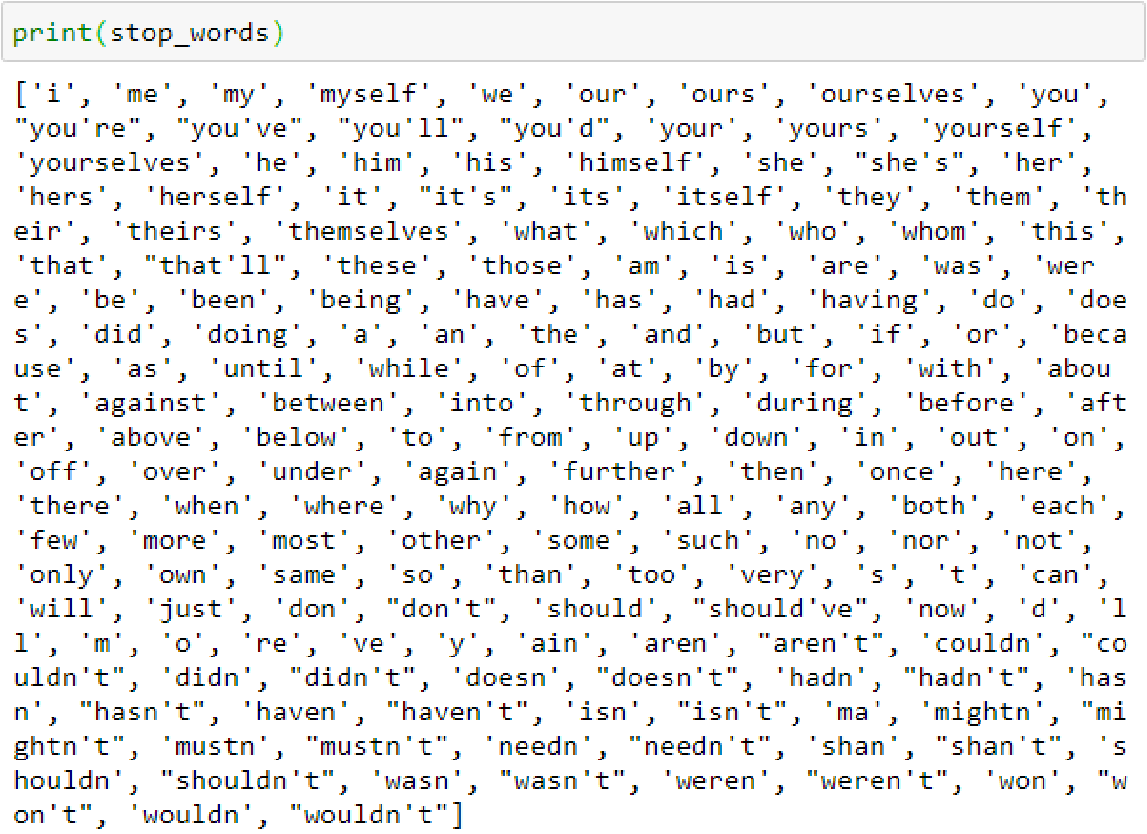
Headline:

Author:

Content:

Copyright 2021 by Muhammad Ammaar Akhtar All Rights Reserved.

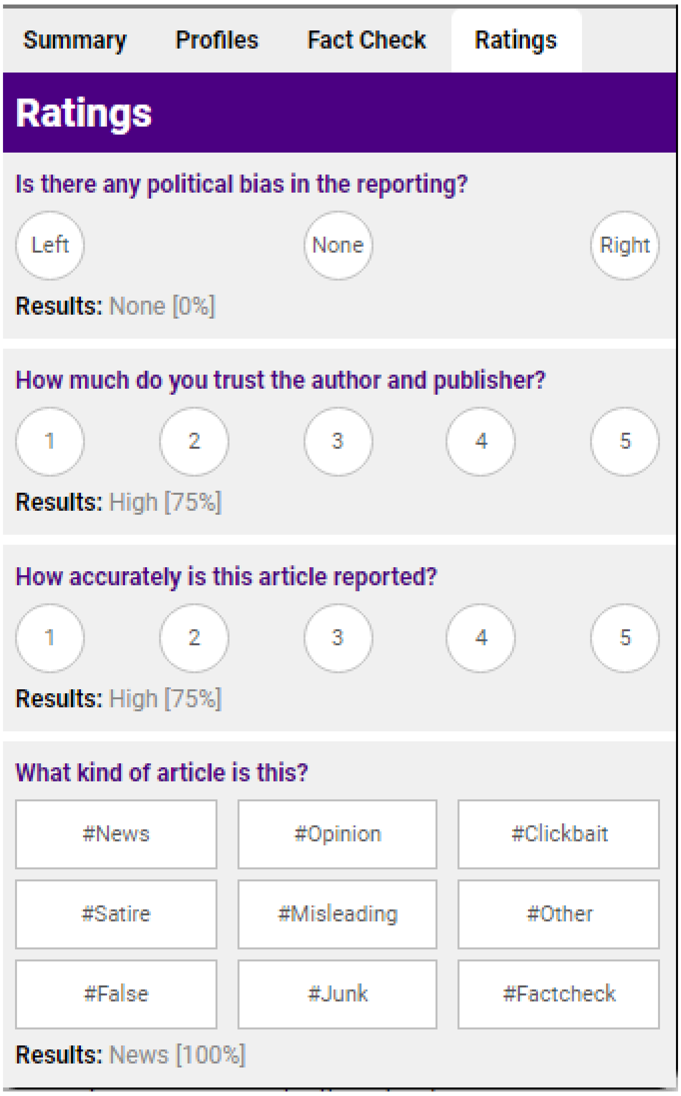
Figure AP- 2: GUI of FNDS



```
print(stop_words)
```

```
['i', 'me', 'my', 'myself', 'we', 'our', 'ours', 'ourselves', 'you',
"you're", "you've", "you'll", "you'd", 'your', 'yours', 'yourself',
'yourselves', 'he', 'him', 'his', 'himself', 'she', "she's", 'her',
'hers', 'herself', 'it', "it's", 'its', 'itself', 'they', 'them', 'th
eir', 'theirs', 'themselves', 'what', 'which', 'who', 'whom', 'this',
'that', "that'll", 'these', 'those', 'am', 'is', 'are', 'was', 'wer
e', 'be', 'been', 'being', 'have', 'has', 'had', 'having', 'do', 'doe
s', 'did', 'doing', 'a', 'an', 'the', 'and', 'but', 'if', 'or', 'beca
use', 'as', 'until', 'while', 'of', 'at', 'by', 'for', 'with', 'abou
t', 'against', 'between', 'into', 'through', 'during', 'before', 'aft
er', 'above', 'below', 'to', 'from', 'up', 'down', 'in', 'out', 'on',
'off', 'over', 'under', 'again', 'further', 'then', 'once', 'here',
'there', 'when', 'where', 'why', 'how', 'all', 'any', 'both', 'each',
'few', 'more', 'most', 'other', 'some', 'such', 'no', 'nor', 'not',
'only', 'own', 'same', 'so', 'than', 'too', 'very', 's', 't', 'can',
'will', 'just', 'don', "don't", 'should', "should've", 'now', 'd', 'l
l', 'm', 'o', 're', 've', 'y', 'ain', 'aren', "aren't", 'couldn', "co
uldn't", 'didn', "didn't", 'doesn', "doesn't", 'hadn', "hadn't", 'has
n', "hasn't", 'haven', "haven't", 'isn', "isn't", 'ma', 'mightn', "mi
ghtn't", 'mustn', "mustn't", 'needn', "needn't", 'shan', "shan't", 's
houldn', "shouldn't", 'wasn', "wasn't", 'weren', "weren't", 'won', "w
on't", 'wouldn', "wouldn't"]
```

Figure AP- 3: Stopwords



Ratings

Is there any political bias in the reporting?

Results: None [0%]

How much do you trust the author and publisher?

Results: High [75%]

How accurately is this article reported?

Results: High [75%]

What kind of article is this?

Results: News [100%]

Figure AP- 4: Our.news

References

[1] Greenspon, E. (2017). The Shattered Mirror: News, Democracy and Trust in the Digital Age. DOI: 978192700986. Ottawa, ON, CA. Public Policy Forum.

[2] B. Al Asaad and M. Erascu, "A Tool for Fake News Detection," 2018 20th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC), 2018, pp. 379-386, doi: 10.1109/SYNASC.2018.00064.

[3] K. Shu, A. S. (2017.). Fake news detection on social media: A data mining perspective. SIGKDD Explor. Newsl.,.

[4] P. Bourgonje, J. M. (2017). From clickbait to fake news detection: An approach based on detecting the stance of headlines to articles. Proceedings of Natural Language Processing meets Journalism.

[5] V. Rubin, N. C. (2016). Fake news or truth? using satirical cues to detect potentially misleading news. Proceedings of the Second Workshop on Computational Approaches to Deception Detection.

[6] David Mondrus, M. M. (n.d.). Trive™ Whitepaper.

[7] Sawant, Abhijit & Bari, Pranit & Chawan, Pramila. (2012). Software Testing Techniques and Strategies. International Journal of Engineering Research and Applications(IJERA). 2. 980-986.

[8] Fake news dataset available at http://www.fakenewschallenge.org

[9] Fake News Challenge Stage 1 (FNC-I) algorithms available at https://www.kaggle.com/abhinavkrjha/fake-news-challenge

[10] CoreNLP Lemmatizer at (https://stanfordnlp.github.io/CoreNLP/simple.html)

[11] Term Frequency Inverse document frequency (TF-IDF) at (nlp.stanford.edu/IR-book/html/htmledition/inverse-document-frequency-1.html)

[12] Software Engineering A Practitioner’s Approach by Roger S. Pressman

[13] Kaggle fake news data set at https://www.kaggle.com/c/fake-news/data