A Project Report

on

FAKE NEWS DETECTION SYSTEM USING MACHINE LEARNING

Submitted in partial fulfilment of the requirements for the award of the degree

Of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

BY

MOHD MISBAH UDDIN **160317733036**

Under The Guidance Of

MR. YASAR AHMED

Assistant Professor

Department Of CSE, DCET

Department of Computer Science Engineering

Deccan College of Engineering and Technology

(Affiliated to Osmania University)

Hyderabad

2021

CERTIFICATE

This is to certify that the project report on “FAKE NEWS DETECTION SYSTEM USING MACHINE LEARNING” being submitted by

MOHD MISBAH UDDIN 160317733036

in partial fulfilment for the award of the Degree of Bachelor of Engineering in Computer Science by the Osmania University is a record of Bonafide work carried out by him under my guidance and supervision. The results embodied in this project report have not been submitted to any other University or Institute for the award of any Degree or Diploma.

Internal Guide

Mr. Yasar Ahmed

Assistant Professor

Dept. of CSE, DCET.

Head of the Department

Dr. Syed Raziuddin

Dept. of CSE, DCET

External Examiner

DECLARATION

This is to certify that the work reported in the present project entitled **"FAKE NEWS DETECTION SYSTEM USING MACHINE LEARNING"** is an authentic record of my genuine work done under the guidance of my guide **Mr. Yasar Ahmed** in the Department of Computer Science & Engineering, Deccan College of Engineering and Technology, Osmania University. The reports are based on the project work done entirely by me and not copied from any other source. I further declare that the work reported in this project has not been submitted and will not be submitted , either in part or in full , for the award of any other degree or diploma in this institute or any other institute or university.

MOHD MISBAH UDDIN (160317733036)

ACKNOWLEDGEMENT

I am thankful to Principal **DR. M.A. MALIK** for providing excellent infrastructure and a nice atmosphere for completing this project successfully and on time. I am thankful to Head of the department **DR. SYED RAZIUDDIN** for providing the necessary facilities during the execution of our project work. I would like to express my sincere gratitude to my project guide **Mr. Yasar Ahmed** for his valuable suggestions and interest throughout the course of this project. This project would not have been a success without his guidance. so, I would extend my deep sense of gratitude towards him for the effort he took in guiding and advising me in all the stages of completion of our project work. I convey my heartfelt thanks to my parents, friends, technical and non-technical staff of the college for their constant support in the successful completion of this project.

ABSTRACT

Fake News is a growing problem in the modern world, it aims at swaying the opinion of the vast majority of people who use social media on a day-to-day basis. This project aims to solve the problem of fake news on internet.Social media and the internet for news consumption is a double- edged sword. On the one hand, it is low cost, easy to access, and its rapid dissemination of information leads people to seek out and consume news from social media.On the other hand, it enables the wide spread of “fake news”, i.e., low quality news with intentionally false information.Fake news is where individuals or organizations intentionally publish hoaxes, propaganda and other misinformation and present it as factual.This project aims to solve the problem of fake news on internet. The project is a web based application which determines whether a news Article is fake or credible, using different machine learning models, which are trained on a large dataset.

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVATIONS

SDLC	Software Development life Cycle
ER	Entity Relationship
DFD	Data Flow Diagram
KNN	K Nearest Neighbour
CNN	Convolutional neural network
SVM	Support vector machine
TF-IDF	Term frequency–inverse document frequency
NLP	Natural language processing
DNN	Deep Neural Network
DDS	Design Document Specification

TABLE OF CONTENTS

Chapter 1

INTRODUCTION

1.1 INTRODUCTION

Fake News is a growing problem in the modern world, it aims at swaying the opinion of the vast majority of people who use social media on a day-to-day basis. This project aims to solve the problem of fake news on internet. Social media and the internet for news consumption is a double- edged sword. On the one hand, it is low cost, easy to access, and its rapid dissemination of information leads people to seek out and consume news from social media. On the other hand, it enables the wide spread of “fake news”, i.e., low quality news with intentionally false information. Fake news is where individuals or organizations intentionally publish hoaxes, propaganda and other misinformation and present it as factual. This can include blog and social media posts and fake online media releases. It also does not include articles that are written from the perspective of a particular opinion or editorial standpoint, provided the information included is factually correct. Fake news and lack of trust in the media are growing problems with huge ramifications in our society. 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 project , it is seeked to produce a model that can accurately predict the likelihood that a given article is fake news. However, in order to solve this problem, it is necessary to have an understanding on what Fake News is. Later, it is needed to look into how the techniques in the fields of machine learning, natural language processing help us to detect fake news. The project is a web-based application which determines whether a news Article is fake or credible, using different machine learning models, which are trained on a large dataset. Web application takes a URL (or a bunch of text) as an input from the user and extracts the relevant text from the URL and then extracts feature vectors from the text using NLP. Machine Learning Models are then used on the feature vectors to classify news source as fake or credible Macmillan English Dictionary defines fake news as "a story that is presented as being a genuine item of news but is in fact not true and is intended to deceive people". But the definition isn't as clear cut in the 21st century. The water becomes murky as we start to question if rumors are fake news or if parodies or political humor is fake news. Can an exaggeration of a simple news article be considered fake as they may portray thesubject in a different, if not in a negative view? While some of these questions have been covered extensively by Ammara . [1] and Aswini *h* [2], it is not on the agenda of this paper.

One can ask the question, why even attempt to combat fake news? The average internet user is not always knowledgeable about the news that they view. The chances of them spreading the said news increases exponentially when the news has an emotional attachment to themselves. This is most seen in the topics related to politics or religion. If left ignored, fake news can have disastrous consequences.

It is of paramount importance that we confront and try to eliminate the problem of fake news. There have been many methods that are proposed ranging from NLP analysis to clustering. [3] uses a system where trusted news is used to create clusters. Whereas, Dinesh Kumar et [4]. takes a unique approach to classifying the news. It proposes an architecture where the text is extracted and cleaned from an image of news. This text is then fed to Google and the scrapped results are compared to determine if the news is true or fake. While these two papers look at the news, [ ]analyses the website itself. It checks for three levels which include URL, Blacklist and Image screening. Classifying the website itself as fake can nip the bud of the fake news originating from the website. While this classifies the platform, [6] proposes a system where the users in twitter can be given a credibility rating after analysis of the users' tweets. While this paper does not directly identify the fake news, it can mark the users who have the highest potential to create and spread fake news. Currently, there exist some systems which one can use to help in the identification of fake news. A plugin known as BS Detector searches through a catalog of web pages that has been flagged as unreliable or fake in their database. Politifact is another US-based website that gives credibility to the claims by US politicians.While the above systems take various methods and perspectives, this paper is confined to dealing with fake news in both Machine Learning and NLP. We have acquired various datasets from different sources whose descriptions will be in present under the section [III]. We have chosen to use Multinomial Naive Bayes, Passive-aggressive classifier and deep neural network on the datasets. In-depth information on the methodology has been provided in the section [IV] and models generated have been analyzed and documented in the section [V]. The following section provides insight into some of the literature papers that also used machine learning and other similar methods to identify fake new

1.2 Problem Statement

The extensive spread of fake news has the potential for extremely negative impacts on individuals and society. Therefore, fake news detection on social media and the internet has recently become an emerging research that is attracting tremendous attention. Analyzing and

detecting fake news on the internet is one of the hardest problems to be solved. The project is designed to identify fake news on the internet by taking the URL/text of the news available on the internet as input

1.3 Objective

The objective of the project is to design a web-application which can accurately classify news as real or fake. The system will take URL of news articles or relevant text of news articles as input and then process it using NLP and by using machine learning techniques, accurately classify news as real or fake. Obviously, it is difficult to achieve 100% accuracy while classifying news, however our aim is to get an accuracy which is as close as possible to the ideal value (i.e.100%) The system should also be able to process the input and classify news as quickly as possible. However, the speed of processing could sometimes depend upon the processing capability of the machine the web-application is being run on. The project should be simple and elegant with a user-friendly interface. Users should be able to access the web-application to identify fake news easily. Users can also provide their feedback to the system , which can later be used to enhance the web-application and also correct the software bugs.

1.4 Scope

The scope of our project is to detect fake news from online articles using machine learning. Our fake news detector makes use of ML techniques to detect fake news in content. Our project has major impact on social media like Facebook and Twitter because major population of world has access to these platforms. Fake news has an impact on decision making of people which could lead to serious mistakes

Chapter 2

Literature Survey

2.1 Literature Survey

In this section, we discuss some of the papers who used machine learning to identify and classify the fake news. [ ] uses a distinctive technique in detecting fake news by creating an 'Ensemble Voting classifier' . It uses many well-known machine-learning classifiers such as Naïve Bayes, K-NN, SVM and many more to verify the news. Further, cross-validation was used and the top three machine learning algorithms with the best accuracy were used in Ensemble Voting Classifier. This model proposed a recognition structure that can productively predict the output and find the important highlights of the news. This allowed for a results ranging from early to late 90s. Text-mining based methods for the detection of fake news have been evaluated by Harita Reddy et [ ]. This paper provided a hybrid approach that combines word vector representations and stylometric features using ensemble methods like bagging, boosting and voting. After the selection of important features, Random Forest, Naïve Bayes, SVM and many more algorithms were applied. This resulted in accuracies up to 95.49 %. Natural Language processing technique was exploited by Kushal Agarwalla et [ ]. to verify the news. NLTK from Python was used with various models including Logistic Regression, SVM, and Naïve Bayes with Lidstone Smoothing. Naive bayes with Lidstone smoothing performed admirably and gave a result of 83% . Perhaps, using only the vector-based methods to extract certain features and to train the classifiers is not an accurate solution as these are fixed to the particular dataset .

Mykhailo Granik et [ ]. implemented a basic approach using Naive Bayes classifier for the detection of fake news. The model was built as a software system and validated over a set of Facebook news posts. It describes the similarity between spam messages and fake news articles by concluding that identical approaches can be taken for both fake news detection and spam filtering by producing a result of 75% accuracy. Akshay Jain et [ ]. proposed a model with two variants which uses Naive Bayes classifier to

predict whether a post on Facebook will be labeled as REAL or FAKE. The first model used the title as their source for vocabulary building, using count vectorizer. And the second model

used text as their source. The results were compared based on their AUC score and the second model was found to be better with a score of 0.93 and 0.912 with and without n_grams respectively. Aswini Thota et [ ] used Deep Learning architectures to detect fake news. Tf-IDF, GloVe and Word2Vec were used along with the DNN model to precisely predict the stance between the article body and given pair of the headline.This paper was able to produce an overall accuracy of 94.31%. [ ] explored different models ranging from Logistic Regression to CNN, RNN, and GRU. This work is mainly concentrated on using pure NLP perspective to identify the presence of fake news by utilizing the linguistic features. Highest precision of 0.97 was obtained using CNN with Max Pooling and Attention. This approach might lose its viability as the fake news gets better at replicating true news. A.Lakshmanarao et [ ] . employed SVM, KNN, Decision tree, and Random forest to build a four models and compared them.It was observed that Random Forest Classification gave the

highest score of 90.7% while least was provided by Support Vector Machines at 75.5%. Shlok Gilda et [ ]. worked only using Natural Language Processing technique to identify the fake news. Probabilistic context-free grammar (PCFG) and Term frequency-inverse document frequency (TF-IDF) of bigrams were applied with various models like gradient boosting and stochastic gradient descent. Among other models, TFIDF of bi-grams with stochastic gradient descent identified fake news with higher accuracy. While the previous papers applied machine learning to the detection of fake news.

[ ] brings something new to the table in the form of human testing. This is the only paper that has tallied and compared the performance of humans against machines. This paper uses two different datasets namely FakeNewsAMT which consists of general news and Celebrity news which as the name suggests contains news about celebrities. The paper used two annotators to classify if the news were true or fake and they had an agreement rate of 70 percent. It is observed that the annotators beat the automated system when dealing with celebrity news dataset but lost by a margin of 3-4% in FakenewsAMT. Multi-feature extraction was also done and it showed that FakeNewsAMT performs best when relying on stylistic features and Celebrity on LIWC feature While the previously mentioned papers have used a plethora of machine learning algorithms, This

paper confines itself to three most important algorithms namely Naive Bayes, Passive Aggressive Classifier and Deep neural networks. The models are built in 8 different datasets whose description is present in the following section

2.2 Datasets

This section covers the description of seven different datasets that were used in this paper which were acquired from a diverse set of sources. Superset is the dataset obtained by extracting the Satement attribute from other various datasets used in this paper. All the datasets underwent similar pre-processing such as cleaning the dataset of corrupted data or dropping the missing value rows. All datasets were divided into train, dev and test sets for the models. If the number of articles in the dataset was less than 10000, the ratio of the split was

Dataset	Attributes	Size
FND-[16]	URL+Headline+Body	4K*3
[17]	Statement+Speaker+URL	10K*3
[18] Route 1	Domain+Content+Title+Author	93K*4
[18] Route 2	Domain+Content+Title	140K*3
[19]	Text+Headline+Source+Keywords	10K*4
[20] competition	Title+Author+Text	25K*3
[21]	Statement+Sub+Speaker+Job+State+Party+Context	8.4K*7
[22]	Statement	18K
Superset	Statement	233K

Table 2.2.2: Dataset Description

taken as 90:5:5. Anything greater than 10000 was split in the ratio 80:10:10. All the datasets contain an almost equal number of true and false articles without being skewed towards one side.

2.3 Analysis

After the careful construction of models on multiple datasets. The results have been documented in the following table.

Dataset	Methodology	Results		
Jru	Naïve Bayes	Stop	Test	Train
			96%	99%
	Passive aggressive	yes	99%	100%
		No	99%	100%
	DNN		99%	100%
Pontes	Naïve Bayes		96.6%	96.5%
	Passive aggressive	yes	98.5%	99%
		no	98.4%	98.9%
	DNN		97%	98%
	Naïve Bayes		96.9%	96.8%
	Passive aggressive	Yes	100%	100%
		no	98.9%	99.8%
	DNN		98%	99%
ClaimsKG	Naïve Bayes		77.2%	89.4%
	Passive aggressive	yes	73.5%	97.3%
		no	73.7%	99.8%
	DNN		77.9%	99.8%
Kaggle	Naïve Bayes		85.1%	85.6%
	Passive aggressive	Yes	83.8%	89.2%
		no	82.6%	89.8%
	DNN		87%	99%
Liar	Naïve Bayes		71.8%	77.4%
	Passive aggressive	yes	64.7%	86.6%
		no	64.3%	99.9%
	DNN		63.9%	73.6%
Newsfiles	Naïve Bayes		97.8%	99.3%
	Passive aggressive	yes	99%	99%
		no	99%	100%

Table 2.2.3: Dataset Documentation

It can be seen from the documented table, that certain datasets have performed significantly better than other datasets. It is observed that the JRU dataset with only three attributes has performed quite well with almost 100 percent accuracy. While one could chalk up the success of the dataset to its small size, the Pontes dataset, on the other hand, proves the conjecture false.

Pontes dataset was split into two routes based on author attribute, which contained 40% of missing values. Although the dataset is massive, both the routes performed equally well with the accuracies in the high 90s. On further analysis, we can conclude that Route 2 models are better when expanded and generalized. ClaimsKG models performed relatively poor compared to other datasets with accuracies in the early 70s. This performance can be ascribed to the fact that only 10000 articles were present. Perhaps more articles and revaluation of its attributes could provide better results.The Kaggle competition dataset, on the other hand, has performed relatively well with accuracies in the mid-80s. Liar dataset performed disappointingly even though it contained the highest number of attributes. Naive Bayes stood out on top, surprisingly performing better than DNN but only in this dataset. This can be attributed to robustness of Naïve Bayes algorithms. A lot more news articles from varied sources are required to buttress the models The final two datasets were approached with a purely NLP perspective. NewsFiles like JRU performed stupendously well in all metrics. Superset with its massive number of articles (233413) showed that an NLP only approach is viable with a respectable mid 80's accuracy. It can be observed from the table that DNN outperformed both naive Bayes and passive-aggressive classifiers in every dataset except one. The success of DNN is due to the fact Neural networks better represent complex, nonlinear structures which in this case is a perfect fit. Naive Bayes and Passive aggressive classifiers both produced similar if not the same results. While adding early stopping helped, there was no significant improvement to warrant its usage. Comparing our results with that of the reference papers. It is seen that some of our models have performed better than those papers that used similar algorithms.

2.4 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. Conroy, Rubin, and Chen [1] outlines several approaches that seem promising towards the aim of perfectly classify the misleading articles. However, these methods have been shown useful only in tandem with more complex methods of analysis. Deep Syntax analysis using Probabilistic Context Free Grammars (PCFG) have been shown to be particularly valuable in combination with n-gram methods. Marco L. Della offered a paper which allows us to recognize how social networks and gadget studying (ML)strategies may be used for faux news detection . Machine Learning is one of the most powerful tools that are available right now. The task of choosing the classifiers emerged from the suitable properties of algorithms. As Naïve Bayes is popular for its multi-class prediction, it was picked up for its

ease and robustness of predicting the class of the text. In fact, one of the problem with other methods is when new samples are collected, model must be retrained to predict the output for new data. This is overcome by using passive aggressive classifier which trains the model incrementally, allowing modifications of the parameters only when needed, while discarding these updates when they don’t alter the equilibrium.

Chapter 3

Proposed System

3.1 Proposed System

The only solution to the problem defined in the earlier sections is to design and implement such a Web based application which will take a news URL or news text as input and give result of its authenticity with higher accuracy. It will be difficult to achieve higher accuracy because of limited dataset.

3.2 Algorithms/Modules used

3.2.1 NLP (Natural Language Processing)

Natural Language Processing (NLP) allows machines to break down and interpret human language. It's at the core of tools we use every day – from translation software, chatbots, spam filters, and search engines, to grammar correction software, voice assistants, and social media monitoring tools. Natural Language Processing (NLP) is a field of Artificial Intelligence (AI) that makes human language intelligible to machines. NLP combines the power of linguistics and computer science to study the rules and structure of language, and create intelligent systems (run on machine learning and NLP algorithms) capable of understanding, analyzing, and extracting meaning from text and speech Natural language understanding (NLU) is used to understand the structure and meaning of human language by analyzing different aspects like syntax, semantics, pragmatics, and morphology.

3.2.2 Pickle

The pickle module implements binary protocols for serializing and de-serializing a Python object structure. Pickling" is the process whereby a Python object hierarchy is converted into a byte stream, and "unpickling" is the inverse operation, whereby a byte stream (from a binary file or bytes-like object) is converted back into an object hierarchy.

3.2.3 Flask APP

It is a Web Server Gateway Interface (WSGI) web application framework designed to create web apps. It is passed the name of the module or package of the application. Once it is created it will act as a central registry for the view functions, the URL rules, template configuration

and much more. Usually, you create a Flask instance in your main module or in the init .py file of your package like this:

from flask import Flask app=Flask(_name_)

3.2.3 Newspaper3k Library

The Newspaper3k package is a Python library used for Web Scraping articles, It is built on top of requests and for parsing lxml. This module is a modified and better version of the Newspaper module which was used for the same purpose To install this module type the below command in the terminal.

pip install newspaper3k

3.2.4 TF-IDF

Term Frequency-Inverse Document Frequency is a numerical statistic that is intended to reflect how important a word is to a document in a collection or corpus The tf–idf value increases proportionally to the number of times a word appears in the document and is offset by the number of documents in the corpus that contain the word, which helps to adjust for the fact that some words appear more frequently in general. tf–idf is one of the most popular term-weighting schemes today A survey conducted in 2015 showed that 83% of text-based recommender systems in digital libraries use tf–idf

3.2.5 Multinomial Naïve Bayes

Naive Bayes is based on Bayes' theorem, where the adjective Naïve says that features in the dataset are mutually independent, occurrence of one feature does not affect the probability of occurrence of the other feature. For small sample sizes, Naïve Bayes can outperform the most powerful alternatives. Being relatively robust, easy to implement, fast, and accurate, it is used in many different fields. Multinomial Naive Bayes relies on very simple representation of document as a 'Bag of words'. The multinomial Naive Bayes classifier is suitable for classification with discrete features (e.g., word counts for text classification). The multinomial distribution normally requires integer feature counts. However, in practice, fractional counts such as tf-idf may also work. Naive Bayes classifier is used in Text Classification, Spam filtering and Sentiment Analysis. It has a higher success rate than other algorithms. Naïve Bayes along with Collaborative filtering are used in Recommended Systems. It is also used in disease prediction based on health parameters. This algorithm has also found its application in

Face recognition. Naive Bayes is used in prediction of weather reports based on atmospheric conditions (temp, wind, clouds, humidity etc.)

3.2.6 Numpy

It is a scientific computing package generating N-dimensional array objects. As for this project, several machine learning models use Numpy as the data container

3.2.6 Scikit-Learn

A Python library built on Numpy. This project uses it mainly for data classification

3.2.7 NLTK

A Python library used for NLP (natural language processing). We will be using NLTK for feature extraction from the news article

Chapter 4

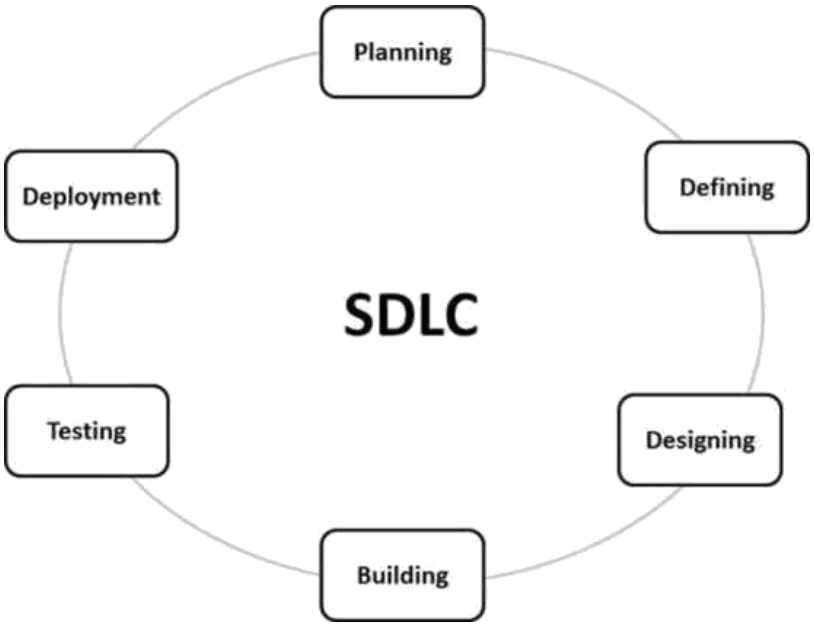
System Analysis

4.1 Introduction

System analysis is the process of gathering and interpreting facts, diagnosing problems and using the information to recommend improvements on the system. System analysis is a problem solving activity that requires intensive communication between the system users and system developers. System analysis or study is an important phase of any system development process. The system is studied to the minutest detail and analysed. The system analyst plays the role of an interrogator and dwells deep into the working of the present system. The system is viewed as a whole and the inputs to the system are identified. The outputs from the organization are traced through the various processing that the inputs phase through in the organization. A detailed study of these processes must be made by various techniques like Interviews, Questionnaires etc. The data collected by these sources must be scrutinized to arrive to a conclusion. The conclusion is an understanding of how the system functions. This system is called the existing system. Now, the existing system is subjected to close study and the problem areas are identified. The designer now functions as a problem solver and tries to sort out the difficulties that the enterprise faces. The solutions are given as a proposal. The proposal is then weighed with the existing system analytically and the best one is selected. The proposal is presented to the user for an endorsement by the user. The proposal is reviewed on user request and suitable changes are made. This loop ends as soon as the user is satisfied with the proposal.

4.2 Software Development Life Cycle

Software Development Life Cycle (SDLC) is a process used by the software industry to design, develop and test high quality software's. The SDLC aims to produce a high-quality software



Fig(4.2.1) SDLC

that meets or exceeds customer expectations, reaches completion within times and cost estimates. The following figure is a graphical representation of the various stages of a typical SDLC.

A typical Software Development Life Cycle consists of the following stages −

4.2.1 Planning and Requirement Analysis

Requirement analysis is the most important and fundamental stage in SDLC. It is performed by the senior members of the team with inputs from the customer, the sales department, market surveys and domain experts in the industry. This information is then used to plan the basic project approach and to conduct product feasibility study in the economical, operational and technical areas. Planning for the quality assurance requirements and identification of the risks associated with the project is also done in the planning stage. The outcome of the technical feasibility study is to define the various technical approaches that can be followed to implement the project successfully with minimum risks.

4.2.2 Defining Requirements

Once the requirement analysis is done the next step is to clearly define and document the product requirements and get them approved from the customer or the market analysts. This is done through an SRS (Software Requirement Specification) document which consists of all the product requirements to be designed and developed during the project life cycle.

4.2.3 Designing the Product Architecture

SRS is the reference for product architects to come out with the best architecture for the product to be developed. Based on the requirements specified in SRS, usually more than one design approach for the product architecture is proposed and documented in a DDS

- Design Document Specification. This DDS is reviewed by all the important stakeholders and based on various parameters as risk assessment, product robustness, design modularity, budget and time constraints, the best design approach is selected for the product. A design approach clearly defines all the architectural modules of the product along with its communication and data flow representation with the external and third party modules (if any). The internal design of all the modules of the proposed architecture should be clearly defined with the minutest of the details in DDS.

4.2.4 Building or Developing the Product

In this stage of SDLC the actual development starts and the product is built. The programming code is generated as per DDS during this stage. If the design is performed in a detailed and organized manner, code generation can be accomplished without much hassle. Developers must follow the coding guidelines defined by their organization and programming tools like compilers, interpreters, debuggers, etc. are used to generate the code. Different high level programming languages such as C, C++, Pascal, Java and PHP are used for coding. The programming language is chosen with respect to the type of software being developed.

4.2.5 Testing the Product

This stage is usually a subset of all the stages as in the modern SDLC models, the testing activities are mostly involved in all the stages of SDLC. However, this stage refers to the testing only stage of the product where product defects are reported, tracked, fixed and retested, until the product reaches the quality standards defined in the SRS.

4.2.6 Deployment in the Market and Maintenance

Once the product is tested and ready to be deployed it is released formally in the appropriate market. Sometimes product deployment happens in stages as per the business strategy of that organization. The product may first be released in a limited segment and tested in the real business environment (UAT- User acceptance testing). Then based on the feedback, the product may be released as it is or with suggested enhancements in the targeting market segment. After the product is released in the market, its maintenance is done for the existing customer base.

Chapter 5

System Specification

5.1 Software Requirements

The software requirements for the development for this project are: -

1.Python (Minimum version 3.7.x)

2.Anaconda Package

3.Google Chrome

4.Xampp (For executing PHP files)

5.HTML5

6.Operating systems: Windows 7 or later, macOS, and Linux

5.2 Hardware Requirements

The minimum hardware requirements for the development of this project are as follows,

1.Physical server or virtual machine.

2.CPU: 2 x 64-bit, 2.8 GHz, 8.00 GT/s CPUs or better.

3.Memory: minimum RAM size of 32 GB, or 16 GB RAM with

4.1600 MHz DDR3 installed, for a typical installation with 50 regular users

5.Storage: Recommended minimum of 100 GB, or 300 GB

6.Internet access to download the files

5.3 System Architecture

The system works on already trained Machine Learning algorithms. Multiple machine learning algorithms have been trained by providing a data set of both fake and authentic news. The summary of overall procedure is as follows :-

1.User enters URL/Text

2.URL is verified, then Newspaper3k extracts relevant text from that news URL.

3.NLP is applied on text extracted.

4.Features extracted from NLP are fed to ML Algorithms.

5.The computed result is displayed to the user

6.Users can provide feedback so that the system can be improved

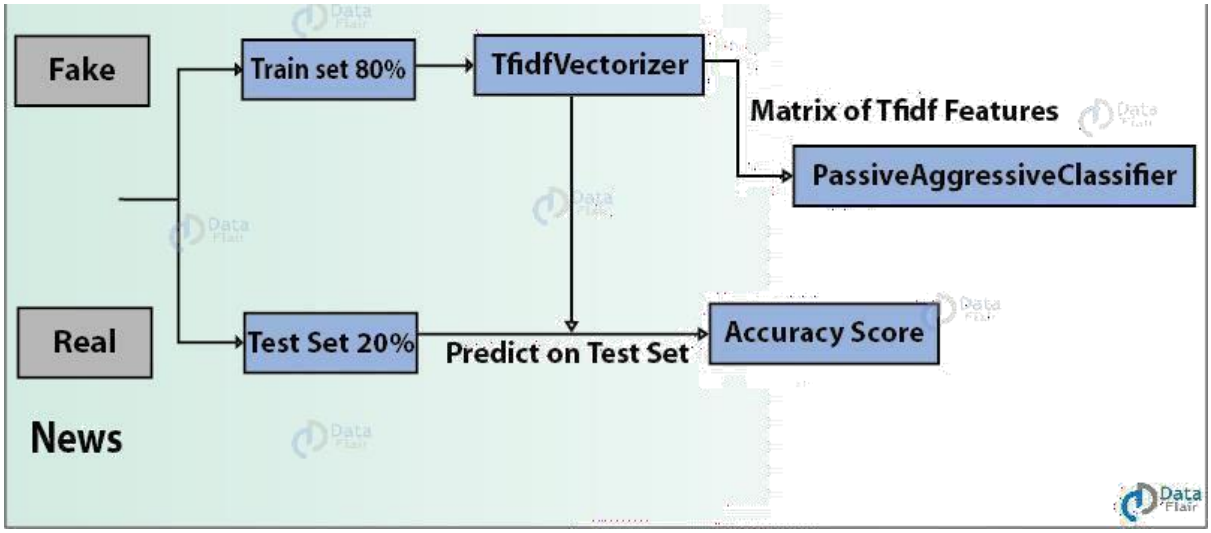


Fig (5.3.1) Fake News Detection System

Chapter 6

System Design

6.1 About UML Diagrams

UML, short for Unified Modeling Language, is a standardized modeling language consisting of an integrated set of diagrams, developed to help system and software developers for specifying, visualizing, constructing, and documenting the artifacts of software systems, as well as for business modeling and other non-software systems. The UML represents a collection of best engineering practices that have proven successful in the modeling of large and complex systems. The UML is a very important part of developing object oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects. Using the UML helps project teams communicate, explore potential designs, and validate the architectural design of the software. It is analogous to the blueprints used in other fields, and consists of different types of diagrams. In the aggregate, UML diagrams describe the boundary, structure, and the behavior of the system and the objects within it. UML is not a programming language but there are tools that can be used to generate code in various languages using UML diagrams. UML has a direct relation with object-oriented analysis and design.

6.2 Use Case Diagram

Use Case diagrams are used to analyze the system's high-level requirements .Use case diagrams give a graphic overview of the actors involved in a system, different functions needed by those actors and how these different functions interact. A use case diagram at its simplest is a representation of a user's interaction with the system that shows the relationship between the user and the different use cases in which the user is involved. A use case diagram can identify the different types of users of a system and the different use cases and will often be accompanied by other types of diagrams as well.

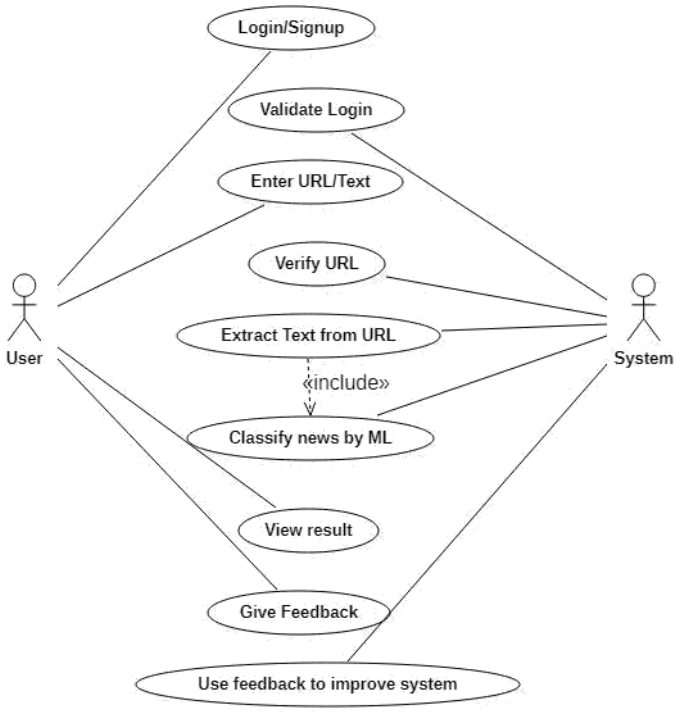


Fig (6.2.1) Use Case Diagram

6.3 Class Diagram

Class diagrams are the main building block of any object-oriented solution. It shows the classes in a system, attributes, and operations of each class and the relationship between each class.In most modeling tools, a class has three parts. Name at the top, attributes in the middle and operations or methods at the bottom. In a large system with many related classes, classes are grouped together to create class diagrams.

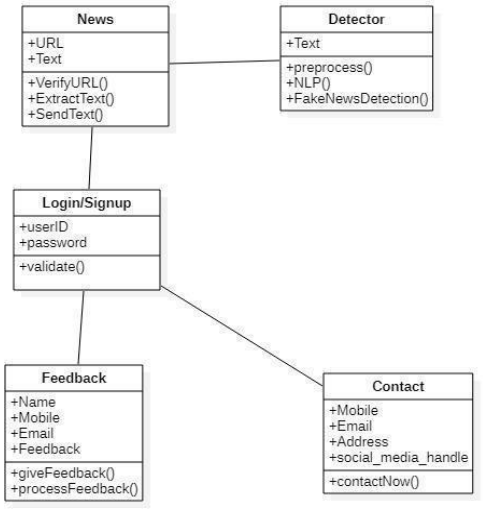


Fig (6.3.1) Class Diagram

6.4 Activity Diagram

Activity diagrams represent workflows in a graphical way. They can be used to describe the business workflow or the operational workflow of any component in a system. Sometimes activity diagrams are used as an alternative to State machine diagrams.

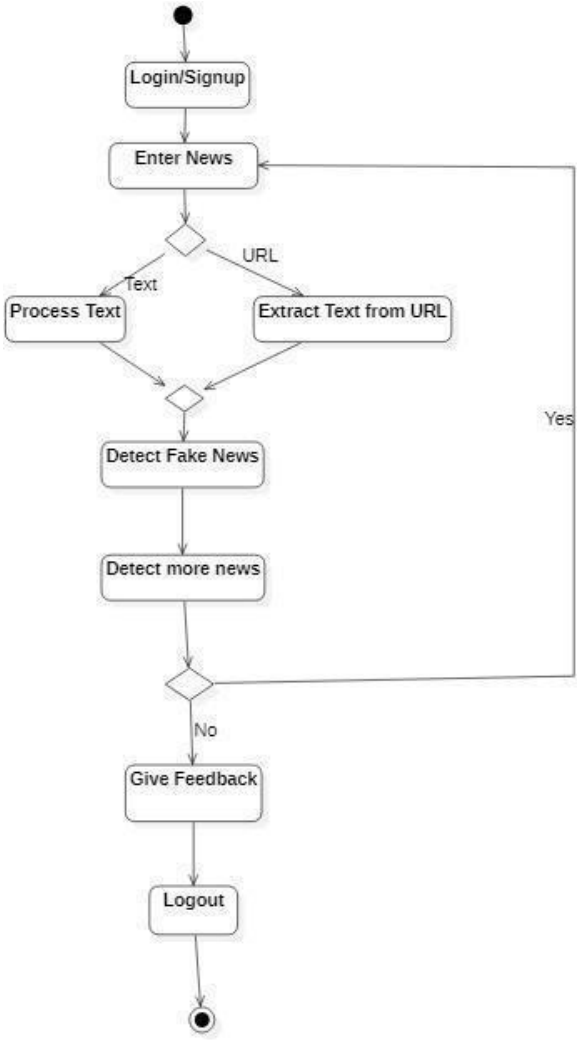


Fig (6.4.1) Activity Diagram

6.5 Sequence Diagram

Sequence diagrams are probably the most important UML diagrams among not only the computer science community but also as design-level models for business application development. Lately, they have become popular in depicting business processes, because of their visually self-explanatory nature. As the name suggests, sequence diagrams describe the sequence of messages and interactions that happen between actors and objects. Actors or objects can be active only when needed or when another object wants to communicate with them. All communication is represented in a chronological manner.As the name suggests, structural diagrams are used to depict the structure of a system. More specifically, it is used in software development to represent the architecture of the system and how the different components are interconnected (not how they behave or communicate, simply where they stand).

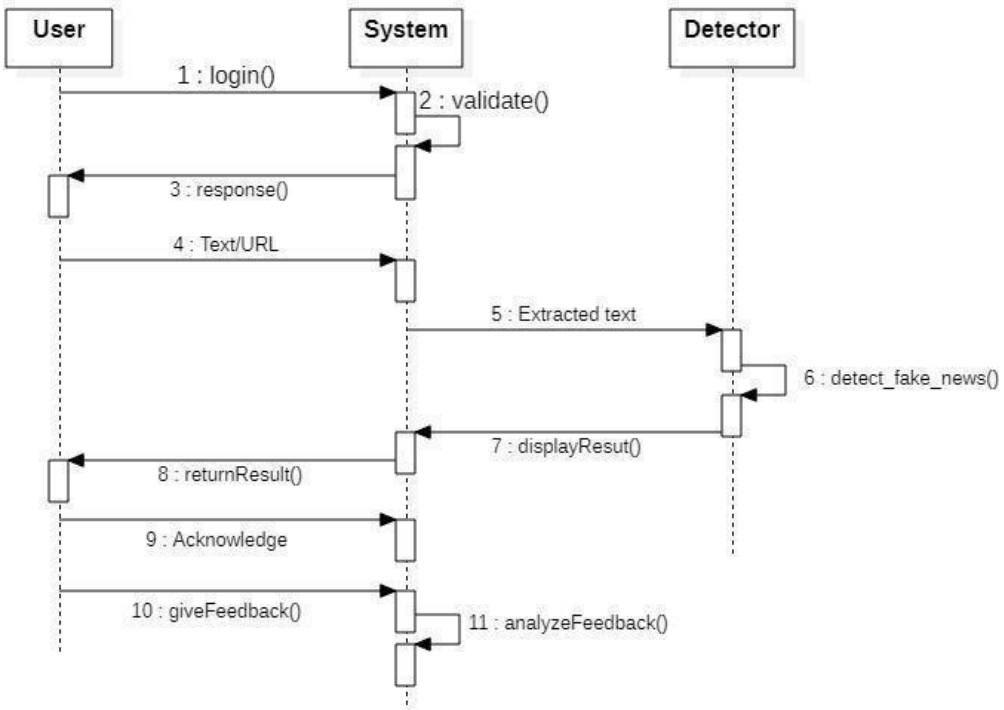


Fig (6.5.1) Sequence Diagram

6.6 ER Diagram

An ER diagram shows the relationship among entity sets. An entity set is a group of similar entities and these entities can have attributes. In other words, an entity is a table or attribute of a table in database, so by showing relationship among tables and their attributes, ER diagram shows the complete logical structure of a database.

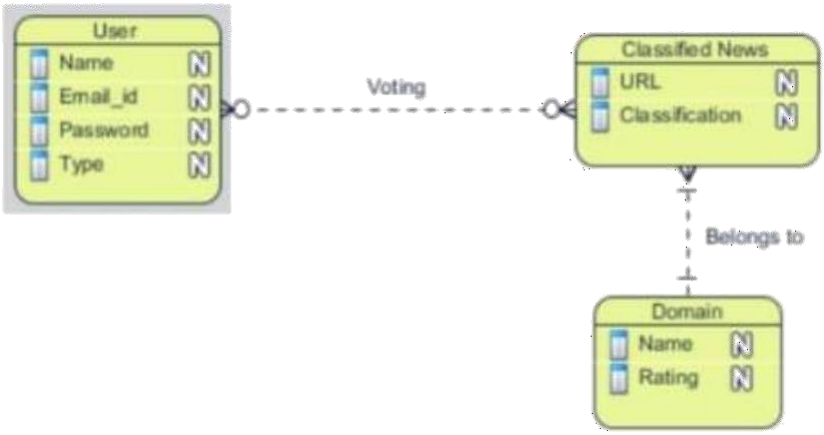


Fig (6.6.1) ER Diagram

6.7 Data-Flow Diagram

A data-flow diagram is a way of representing a flow of data through a process or a system (usually an information system). The DFD also provides information about the outputs and inputs of each entity and the process itself.

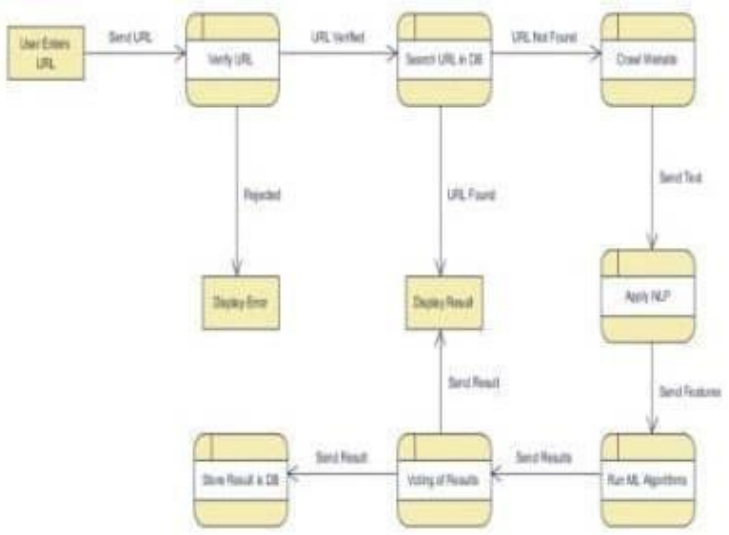


Fig (6.7.1) DFD Diagram

Chapter 7

Coding and Screenshots

7.1 Fake News Detection Module

7.1.1 Fake News Detection Using URL

fake_news_detection.py

#Importing the libraries import pandas as pd import numpy as np

from sklearn.metrics import classification_report, confusion_matrix from sklearn.pipeline import Pipeline from sklearn.model_selection import train_test_split from sklearn.naive_bayes import MultinomialNB

from sklearn.feature_extraction.text import TfidfVectorizer import pickle

#Importing the cleaned file containing the text and label news = pd.read_csv('news.csv') X = news['text']

y = news['label']

#Splitting the data into train

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2)

#Creating a pipeline that first creates bag of words(after applying stopwords) & then applies Multinomial Naive Bayes model

pipeline = Pipeline([('tfidf', TfidfVectorizer(stop_words='english')), ('nbmodel', MultinomialNB())])

#Training our data pipeline.fit(X_train, y_train)

#Predicting the label for the test data pred = pipeline.predict(X_test)

#Checking the performance of our model print(classification_report(y_test, pred)) print(confusion_matrix(y_test, pred))

#Serialising the file

with open('model.pickle', 'wb') as handle:

pickle.dump(pipeline, handle, protocol=pickle.HIGHEST_PROTOCOL)

Output :-

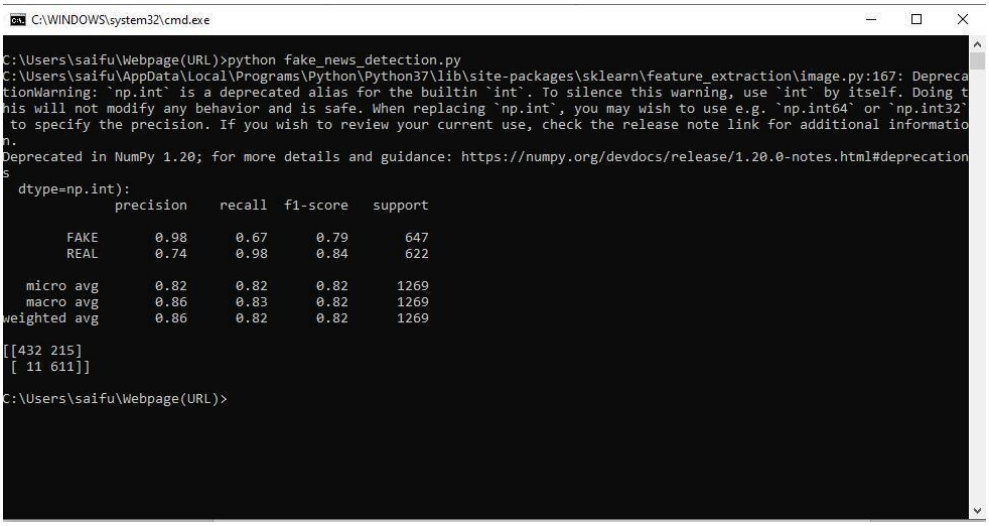


Fig (7.1.1) fake_news_detection.py

app.py

#Importing the Libraries import numpy as np from flask import Flask,

request,render_template from flask_cors import CORS import os

```
import joblib import pickle import flask import os

import newspaper

from newspaper import Article import urllib


#Loading Flask and assigning the model variable app = Flask( name )

CORS(app)

app=flask.Flask( name ,template_folder='templates')



with open('model.pickle', 'rb') as handle: model = pickle.load(handle)


@app.route('/') def main():

return render_template('main.html')

#Receiving the input url from the user and using Web Scrapping to extract the news content @app.route('/predict',methods=['GET','POST']) def predict():


url =request.get_data(as_text=True)[5:] url = urllib.parse.unquote(url)

article = Article(str(url))


article.download() article.parse() article.nlp()

news = article.summary

#Passing the news article to the model and returing whether it is Fake or Real pred = model.predict([news])

return render_template('main.html', prediction_text='The news is "{}"'.format(pred[0]))
```

if name ==" main ": port=int(os.environ.get('PORT',5000))

app.run(port=port,debug=True,use_reloader=False)

Output :-

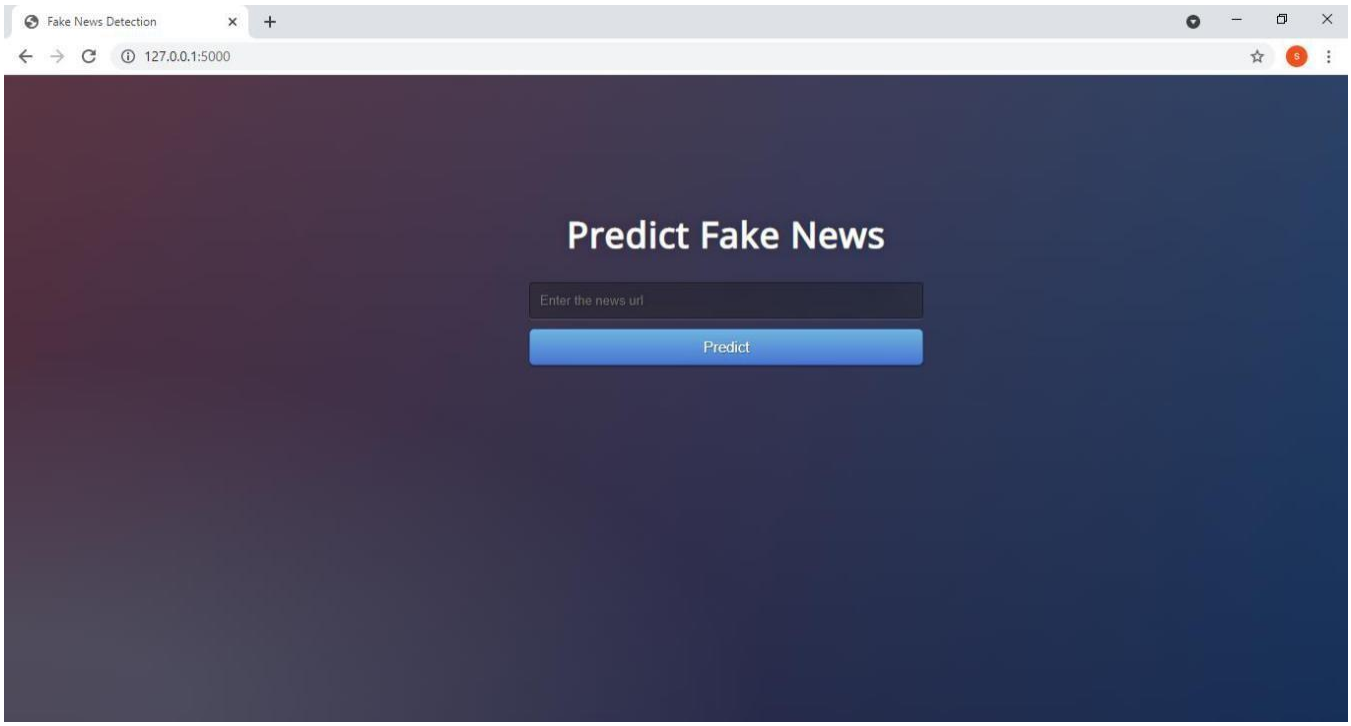


Fig (7.1.2) app.py

7.1.2 Fake News Detection Using Text

predictionModel.py

preprocessing import timeit

```
from nltk.stem import WordNetLemmatizer from nltk import pos_tag
from nltk.corpus import stopwords from nltk.corpus import wordnet import _pickle as pickle
import string import nltk


nltk.data.path.append('./nltk_data')


start = timeit.default_timer()
with open("pickle/pipeline.pkl", 'rb') as f: pipeline = pickle.load(f)
stop = timeit.default_timer()
print('=> Pickle Loaded in: ', stop - start)



class PredictionModel: output = {}

# constructor
def       init (self, text): self.output['original'] = text

def predict(self): self.preprocess()
self.pos_tag_words()

# Merge text
```

```
clean_and_pos_tagged_text = self.output['preprocessed'] + \ ' ' + self.output['pos_tagged']

self.output['prediction'] = 'FAKE' if pipeline.predict( [clean_and_pos_tagged_text])[0] == 0
else 'REAL'

return self.output

# Helper methods def preprocess(self):

# lowercase the text

text = self.output['original'].lower()

# remove the words counting just one letter text = [t for t in text.split(" ") if len(t) > 1]

# remove the words that contain numbers

text = [word for word in text if not any(c.isdigit() for c in word)]

# tokenize the text and remove puncutation

text = [word.strip(string.punctuation) for word in text]

# remove all stop words

stop = stopwords.words('english') text = [x for x in text if x not in stop]
```

```
# remove tokens that are empty text = [t for t in text if len(t) > 0]

# pos tag the text pos_tags = pos_tag(text)

# lemmatize the text
text = [WordNetLemmatizer().lemmatize(t[0], self.get_wordnet_pos(t[1]))
for t in pos_tags]

# join all
self.output['preprocessed'] = " ".join(text)

def get_wordnet_pos(self, pos_tag): if pos_tag.startswith('J'):
return wordnet.ADJ
elif pos_tag.startswith('V'): return wordnet.VERB
elif pos_tag.startswith('N'): return wordnet.NOUN
elif pos_tag.startswith('R'): return wordnet.ADV
else:
return wordnet.NOUN

def pos_tag_words(self): pos_text = nltk.pos_tag(
```

nltk.word_tokenize(self.output['preprocessed'])) self.output['pos_tagged'] = " ".join(

[pos + "-" + word for word, pos in pos_text])

Output :-

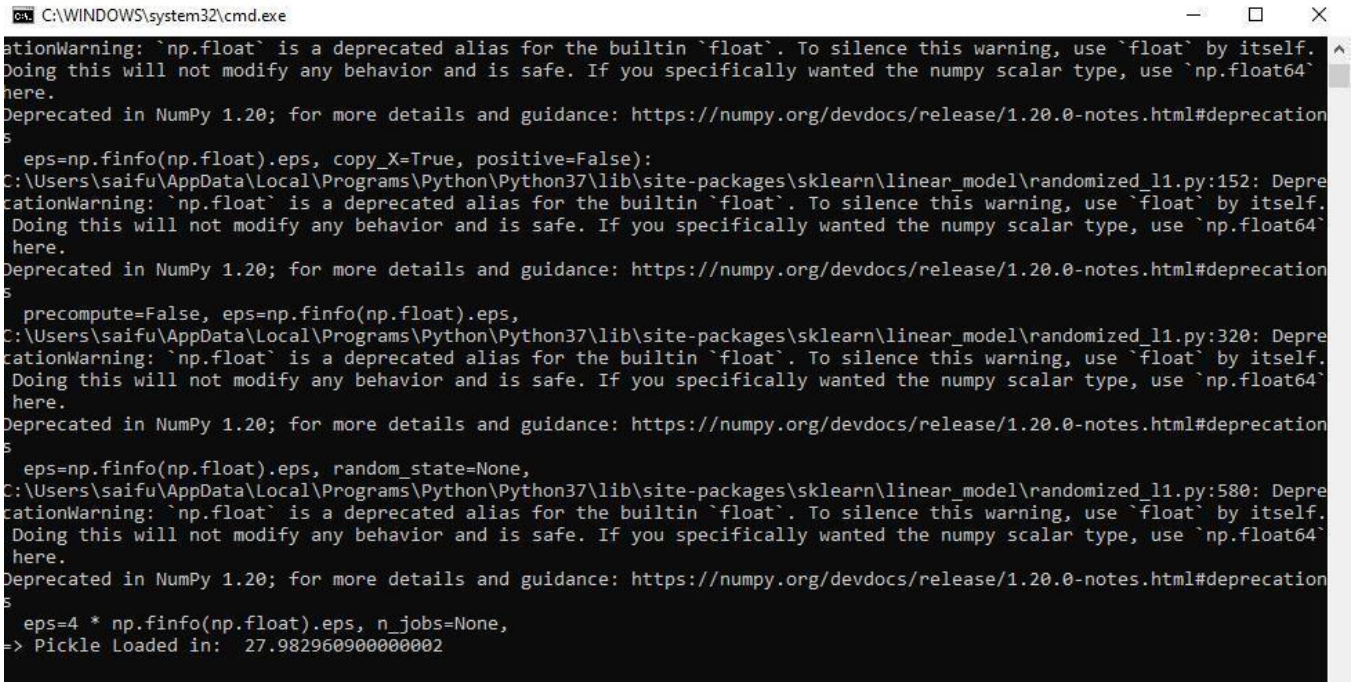


Fig (7.1.3) predictionModel.py

app.py

```
from flask import Flask, jsonify, request, render_template from predictionModel import
PredictionModel import pandas as pd

from random import randrange



app = Flask( name , static_folder="./public/static", template_folder="./public")

@app.route("/") def home():

return render_template('index.html') @app.route('/predict', methods=['POST'])

def predict():

model = PredictionModel(request.json)


return jsonify(model.predict()) @app.route('/random', methods=['GET']) def random():

data = pd.read_csv("data/fake_or_real_news_test.csv") index = randrange(0, len(data)-1, 1)

return jsonify({'title': data.loc[index].title, 'text': data.loc[index].text})



@app.route("/notebook") def notebook():

return render_template('Notebook.html') # Only for local running

if                  name   == '   main       ': app.run()
```

Output :-

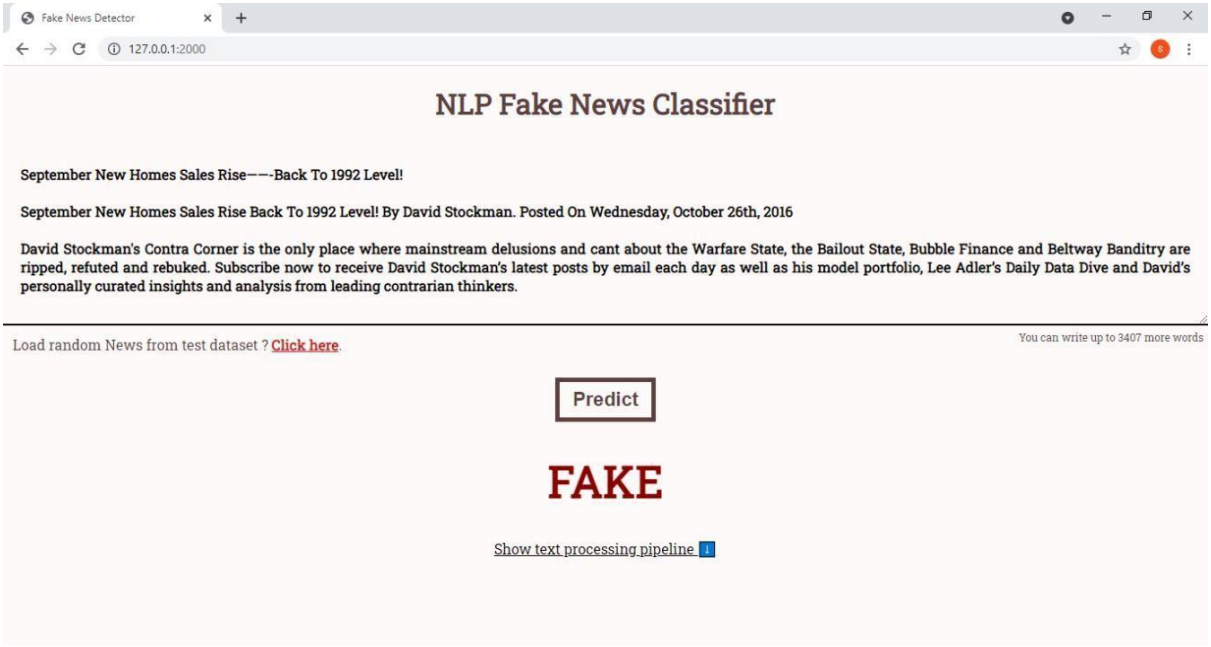


Fig (7.1.4) app.py

7.2 User-Interface Module

7.2.1 Home Page Index.html

<!doctype html>

<html lang="en">

<head>

<!-- Required meta tags -->

<meta charset="utf-8">

<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">

<link rel="icon" href="img/favicon.png" type="image/png"> <title>Pied Piper</title>

<!-- Bootstrap CSS -->

<link rel="stylesheet" href="css/bootstrap.css">

<link rel="stylesheet" href="css/themify-icons.css">

<link rel="stylesheet" href="vendors/fontawesome/css/all.min.css"> <link rel="stylesheet" href="vendors/owl-carousel/owl.carousel.min.css"> <link rel="stylesheet" href="vendors/animate-css/animate.css"> <!-- main css

-->

<link rel="stylesheet" href="css/style.css">

<link rel="stylesheet" href="css/responsive.css">

</head>

<body>

<!--================Header Menu Area =================-->

<header class="header_area">

<nav class="navbar navbar-expand-lg navbar- light">

<!-- Brand and toggle get grouped for better mobile display -->

<img src="img/logo.jpg"

alt="">

<button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">

```
</button>

<!-- Collect the nav links, forms, and other content for toggling -->

<div class="collapse navbar-collapse offset" id="navbarSupportedContent">

<ul class="nav navbar-nav menu_nav ml-auto">

<li class="nav-item"><a class="nav-link" href="index.html">Home</a></li> <li class="nav-item"><a class="nav-link"

href="feature.html">Features</a></li>

<li class="nav-item"><a class="nav-link"
href="http://localhost/fl/index.php">FNDS</a></li>


<li class="nav-item"><a class="nav-link" href="contact.html">Contact</a></li>

<li class="nav-item"><a class="nav-link" href="FAQ.html">FAQ</a></li>

<li class="nav-item"><a class="nav-link" href="contact/index.php">Feedback</a></li>

</ul>

</div>

<div class="right-button">

<ul>

<li><a class="sign_up" href="http://localhost/fl/signup.php">Sign Up</a></li> </ul>

</div>

</div>

</nav>

</div>

</header>

<!--================Header Menu Area =================-->
```

```
<!--================Home Banner Area ================-->

<section class="home_banner_area">

<div class="banner_inner d-flex align-items- center"> <div class="overlay"></div>

<div class="container">

<div class="row">

<div class="col-lg-6 offset-lg-6 col-xl-5 offset-xl-7">

<div class="banner_content">

<h3>Tired of Fake News?<br/>We can help!</h3>

<p>Pied Piper is a smart fake news detection system that can accurately predict whether a news article is real or fake.<br>Pied Piper has an accuracy percentage of upto 85%.<br>You no longer have to worry about being tricked by fake news articles or messages. </p>

<a class="banner_btn" href="http://localhost/fl/index.php">Click here to use Pied Piper<i class="ti-arrow-right"></i></a>

</div>

</div>

</div>

</div>

</div>

</section>

<!--================End Home Banner Area ================-->
```

```
<!--================Service Area ================-->

<section class="service-area area-padding">

<div class="container">

<div class="row">

<!-- Single service -->

<div class="col-md-6 col-lg-4">


<div class="single-service">

<div class="service-icon">

<i class="ti-pencil-alt"></i>

</div>

<div class="service-content">

<h5>User-friendy Interface</h5>

<p>Pied Piper uses a simple yet efficient User Interface , making it easy for users to use Pied Piper and identify fake news articles</p>

<a href="#">Read More</a>

</div>

</div>

</div>



<!-- Single service -->

<div class="col-md-6 col-lg-4">

<div class="single-service">

<div class="service-icon">
```

```
<i class="ti-image"></i>

</div>

<div class="service-content">

<h5>Accurate Prediction</h5>

<p>Pied Piper can classify Fake news with a precision of 90% and Real news with a precision of 79% </p>

<a href="#">Read More</a>

</div>

</div>

</div>

<!-- Single service -->

<div class="col-md-6 col-lg-4">

<div class="single-service">
```

features.html

```
<!doctype html>

<html lang="en">

<head>

<!-- Required meta tags -->

<meta charset="utf-8">

<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">

<link rel="icon" href="img/favicon.png" type="image/png">

<title>Features</title>

<!-- Bootstrap CSS -->
```

<link rel="stylesheet" href="css/bootstrap.css">

<link rel="stylesheet" href="css/themify-icons.css">

<link rel="stylesheet" href="vendors/fontawesome/css/all.min.css"> <link rel="stylesheet" href="vendors/owl-carousel/owl.carousel.min.css"> <link rel="stylesheet" href="vendors/animate-css/animate.css"> <!-- main css --

>

<link rel="stylesheet" href="css/style.css">

<link rel="stylesheet" href="css/responsive.css">

</head>

<body>

<!--================Header Menu Area =================-->

<header class="header_area">

<nav class="navbar navbar-expand-lg navbar-light">

<!-- Brand and toggle get grouped for better mobile display -->

<img src="img/logo.jpg" alt="">

<img src="img/logo.png" alt=""> <button class="navbar-toggler" type="button" data-toggle="collapse" data-

target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">

```
<span class="icon-bar"></span>

<span class="icon-bar"></span>

</button>

<!-- Collect the nav links, forms, and other content for toggling -->

<div class="collapse navbar-collapse offset" id="navbarSupportedContent">

<ul class="nav navbar-nav menu_nav ml-auto">

<li class="nav-item"><a class="nav-link" href="index.html">Home</a></li> <li class="nav-item"><a class="nav-link" href="feature.html">Features</a></li>

<li class="nav-item"><a class="nav-link"
href="http://localhost/fl/index.php">FNDS</a></li>

<li class="nav-item"><a class="nav-link" href="contact.html">Contact</a></li> <li class="nav-item"><a class="nav-link" href="FAQ.html">FAQ</a></li>

<li class="nav-item"><a class="nav-link" href="contact/index.php">Feedback</a></li>

</ul>

</div>

<div class="right-button">

<ul>

<li><a class="sign_up" href="http://localhost/fl/signup.php">Sign Up</a></li> </ul>

</div>

</div>
```

```
</nav>
</div>
</header>
<!--================Header Menu Area =================-->

<!--================Hero Banner Area Start =================-->
<section class="hero-banner d-flex align-items-center">
<div class="container text-center">
<h2>Features</h2>
<nav aria-label="breadcrumb" class="banner-breadcrumb">
<ol class="breadcrumb">
<li class="breadcrumb-item"><a href="#">Home</a></li>
<li class="breadcrumb-item active" aria-current="page">Features</li>
</ol>
</nav>
</div>
</section>
<!--================Hero Banner Area End =================-->

<!--================Service Area =================-->
<section class="service-area area-padding">
<div class="container">
<div class="row">
```

<!-- Single service -->

<h5>User-friendy Interface</h5>

Pied Piper uses a simple yet efficient User Interface , making it easy for users to use Pied Piper and identify fake news articles

Read More

contact.html

<!doctype html>

<html lang="en">

<head>

<!-- Required meta tags -->

<meta charset="utf-8">

<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">

<link rel="icon" href="img/favicon.png" type="image/png"> <title>Contact</title>

<!-- Bootstrap CSS -->

<link rel="stylesheet" href="css/bootstrap.css">

<link rel="stylesheet" href="css/themify-icons.css">

<link rel="stylesheet" href="vendors/fontawesome/css/all.min.css"> <link rel="stylesheet" href="vendors/owl-carousel/owl.carousel.min.css"> <link rel="stylesheet" href="vendors/animate-css/animate.css"> <!-- main css

-->

<link rel="stylesheet" href="css/style.css">

<link rel="stylesheet" href="css/responsive.css">

</head>

<body>

<!--================Header Menu Area =================-->

<header class="header_area">

<nav class="navbar navbar-expand-lg navbar- light">

<!-- Brand and toggle get grouped for better mobile display -->

<img src="img/logo.jpg" alt=""> <img src="img/logo.png"

alt="">

<button class="navbar-toggler" type="button" data-toggle="collapse" data-

target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">

```
<span class="icon-bar"></span>

<span class="icon-bar"></span>

</button>

<!-- Collect the nav links, forms, and other content for toggling -->

<div class="collapse navbar-collapse offset" id="navbarSupportedContent">

<ul class="nav navbar-nav menu_nav ml-auto">

<li class="nav-item"><a class="nav-link" href="index.html">Home</a></li> <li class="nav-item"><a class="nav-link"

href="feature.html">Features</a></li>

<li class="nav-item"><a class="nav-link"
href="http://localhost/fl/index.php">FNDS</a></li>



<li class="nav-item"><a class="nav-link" href="contact.html">Contact</a></li>

<li class="nav-item"><a class="nav-link" href="FAQ.html">FAQ</a></li>

<li class="nav-item"><a class="nav-link" href="contact/index.php">Feedback</a></li>

</ul>

</div>

<div class="right-button">

<ul>



<li><a class="sign_up" href="http://localhost/fl/signup.php">Sign Up</a></li> </ul>

</div>

</div>
```

```
</nav>

</div>

</header>

<!--================Header Menu Area =================-->




<!--================Hero Banner Area Start =================-->

<section class="hero-banner d-flex align-items-center">

<div class="container text-center">

<h2>Contact</h2>

<nav aria-label="breadcrumb" class="banner- breadcrumb"> <ol class="breadcrumb">

<li class="breadcrumb-item"><a href="#">Home</a></li>

<li class="breadcrumb-item active" aria- current="page">Contact</li> </ol>

</nav>

</div>

</section>

<!--================Hero Banner Area End =================-->




<!-- ================ contact section start ================ --

> <section class="contact-section area-padding">
```

<iframe
src="https://www.google.com/maps/embed?pb=!1m18!1m12!1m3!1d7615.03188257797 1!2d7
8.462492024616!3d17.387010352315215!2m3!1f0!2f0!3f0!3m2!1i1024!2i768!4f13.1!3m3!1m2!

1s0x3bcb9779c7128a67%3A0xfda8b2710ebbf716!2sDeccan%20College%20Of%20Enginee ri ng%20And%20Technology!5e0!3m2!1sen!2sin!4v1619902759566!5m2!1sen!2sin" width="1000" height="450" style="border:0;" allowfullscreen="" loading="lazy"></iframe>

FAQ.html

<!doctype html>

<html lang="en">

<head>

<!-- Required meta tags -->

<meta charset="utf-8">

<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">

<link rel="icon" href="img/favicon.png" type="image/png"> <title>Frequently Asked Questions</title>

<!-- Bootstrap CSS -->

<link rel="stylesheet" href="css/bootstrap.css">

<link rel="stylesheet" href="css/themify-icons.css">

<link rel="stylesheet" href="vendors/fontawesome/css/all.min.css"> <link rel="stylesheet" href="vendors/owl-carousel/owl.carousel.min.css"> <link rel="stylesheet" href="vendors/animate-css/animate.css"> <!-- main css

-->

<link rel="stylesheet" href="css/style.css">

<link rel="stylesheet" href="css/responsive.css">

</head>

<body>

<!--================Header Menu Area =================-->

<header class="header_area">

<nav class="navbar navbar-expand-lg navbar- light">

<!-- Brand and toggle get grouped for better mobile display -->

<img src="img/logo.jpg" alt=""> <img src="img/logo.png"

alt="">

<button class="navbar-toggler" type="button" data-toggle="collapse" data-

```
target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-
expanded="false" aria-label="Toggle navigation">

<span class="icon-bar"></span>

<span class="icon-bar"></span>

<span class="icon-bar"></span>

</button>

<!-- Collect the nav links, forms, and other content for toggling -->

<div class="collapse navbar-collapse offset" id="navbarSupportedContent">

<ul class="nav navbar-nav menu_nav ml-auto">

<li class="nav-item"><a class="nav-link" href="index.html">Home</a></li> <li class="nav-
item"><a class="nav-link"

href="feature.html">Features</a></li>

<li class="nav-item"><a class="nav-link"
href="http://localhost/fl/index.php">FNDS</a></li>



<li class="nav-item"><a class="nav-link" href="contact.html">Contact</a></li>

<li class="nav-item"><a class="nav-link" href="FAQ.html">FAQ</a></li>

<li class="nav-item"><a class="nav-link" href="contact/index.php">Feedback</a></li>

</ul>

</div>

<div class="right-button">

<ul>

<li><a class="sign_up" href="http://localhost/fl/signup.php">Sign Up</a></li> </ul>
```

```
</div>

</div>

</nav>

</div>


</header>

<!--================Header Menu Area =================-->




<!--================Hero Banner Area Start =================-->

<section class="hero-banner d-flex align-items-center">

<div class="container text-center">

<h2>Frequently Asked Questions</h2>

<nav aria-label="breadcrumb" class="banner- breadcrumb"> <ol class="breadcrumb">

<li class="breadcrumb-item"><a href="#">Home</a></li>

<li class="breadcrumb-item active" aria-current="page">Blog Details</li> </ol>

</nav>

</div>

</section>

<!--================Hero Banner Area End =================-->
```

<!--================ Start Faq Area ================-->

<section class="faq_area area-padding-bottom">

<h4>Have any queries?</h4>

Below are a few questions that are frequently asked by our users
If your question is not listed below, you can contact us anytime!

7.2.2 Login Page

Index.php

<?php

define('TITLE',"FNDS");

include 'includes/header.php';

?>

<hr>

<h1>Fake News Detection System</h1>

Our smart Fake News Detector can classify news accurately as REAL or FAKE and thereby enable users to verify their news sources

Pied Piper is a smart fake news detection system that can accurately predict whether a news article is real or fake.

Pied Piper has an accuracy percentage of upto 85%.

You no longer have to worry about being tricked by fake news articles or messages.

Pied Piper uses a simple yet efficient User Interface , making it easy for users to use Pied Piper and identify fake news articles

<img src="f1.jpg">

<h6> In order to use Pied Piper, please login </h6>

<h6>If you are a new user please Signup !</h6>

<hr>

<?php

include 'includes/footer.php';

?>

signup.php

```
<?php
define('TITLE',"Signup | Franklin's Fine Dining"); include 'includes/header.php';

if(isset($_SESSION['userId']))
{
header("Location: index.php"); exit();
}
?>
<div id="contact">
<hr>
<h1>Signup</h1>
<?php

$userName = '';
$email = '';
if(isset($_GET['error']))
{
if($_GET['error'] == 'emptyfields')
{
echo '<p class="closed">*Fill In All The Fields</p>';
$userName = $_GET['uid'];
```

```
$email = $_GET['mail'];
}
else if ($_GET['error'] == 'invalidmailuid')
{
echo '<p class="closed">*Please enter a valid email and user name</p>';
}
else if ($_GET['error'] == 'invalidmail')
{
echo '<p class="closed">*Please enter a valid email</p>';
}
else if ($_GET['error'] == 'invaliduid')
{
echo '<p class="closed">*Please enter a valid user name</p>';
}
else if ($_GET['error'] == 'passwordcheck')
{
echo '<p class="closed">*Passwords donot match</p>';
}
else if ($_GET['error'] == 'usertaken')
{
echo '<p class="closed">*This User name is already taken</p>';
}
else if ($_GET['error'] == 'invalidimagetype')
{
```

```
echo '<p class="closed">*Invalid image type. Profile image must be a .jpg or .png
file</p>';
}
else if ($_GET['error'] == 'imguploaderror')
{
echo '<p class="closed">*Image upload error</p>';
}
```

profile.php

```
<?php
define('TITLE',"My Profile | Franklin's Fine Dining"); include 'includes/header.php';

if(!isset($_SESSION['userId']))
{
header("Location: index.php"); exit();
}
?>

<div style="text-align: center">
<img id="userDp" src=<?php echo "./uploads/".$_SESSION['userImg']; ?> >

<h1><?php echo strtoupper($_SESSION['userUid']);
```

```
?></h1> <hr>
</div>
<h3><?php echo strtoupper($_SESSION['f_name']) . " " . strtoupper($_SESSION['l_name']);
?></h3>
<p>
<?php
if ($_SESSION['gender'] == 'm')
{
echo 'Male';
}
else if ($_SESSION['gender'] == 'f')
{
echo 'Female';
}
?>
</p>
<h6><?php echo $_SESSION['headline']; ?></h6>
<p><?php echo $_SESSION['bio'];?></p>

<br><br><br><br>
<?php include 'includes/footer.php'; ?>
```

edit-profile.php

```
<?php



define('TITLE', "Edit Profile | Franklin's Fine Dining"); include 'includes/header.php';


if (!isset($_SESSION['userId']))
{
header("Location: index.php"); exit();
}



?>
<div style="text-align: center">
<img id="userDp" src=<?php echo "./uploads/".$_SESSION['userImg']; ?> >



<h1><?php echo strtoupper($_SESSION['userUid']);
?></h1> </div>



<?php
```

```
$userName = '';
$email = '';



if(isset($_GET['error']))
{
if($_GET['error'] == 'emptyemail')
{
echo '<p class="closed">*Profile email cannot be empty</p>';
$email = $_GET['mail'];

}
else if ($_GET['error'] == 'invalidmail')
{
echo '<p class="closed">*Please enter a valid email</p>';
}
else if ($_GET['error'] == 'emptyoldpwd')
{
echo '<p class="closed">*You must enter the current password to change it</p>';
}
else if ($_GET['error'] == 'emptynewpwd')
{
```

```
echo '<p class="closed">*Please enter the new password</p>';
}
else if ($_GET['error'] == 'emptyreppwd')
{
echo '<p class="closed">*Please confirm new password</p>';
}
else if ($_GET['error'] == 'wrongpwd')
{
echo '<p class="closed">*Current password is wrong</p>';
}
else if ($_GET['error'] == 'samepwd')
{
echo '<p class="closed">*New password cannot be same as old password</p>';
}
else if ($_GET['error'] == 'passwordcheck')
{
echo '<p class="closed">*Confirmation password is not the same as the new password</p>';
}
}
```

7.2.3 Feedback Page index.php

```
<?php include 'sendemail.php'; ?>



<!DOCTYPE html>

<html lang="en" dir="ltr">

<head>

<meta charset="utf-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Responsive Contact Form</title> <link rel="stylesheet" href="style.css">

<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/5.13.1/css/all.min.css">

</head>

<body>



<!--alert messages start-->

<?php echo $alert; ?>

<!--alert messages end-->



<!--contact section start-->

<div class="contact-section">

<div class="contact-info">
```

Darussalam,Nampally ,Hyderabad-500001

piedpiper708@gmail.com

+91 8074339455

Mon - Sat 9:00 AM to 5:00 PM

<h2>Feedback Form</h2>

<form class="contact" action="" method="post">

<input type="text" name="name" class="text-box" placeholder="Your Name" required>

<input type="email" name="email" class="text-box" placeholder="Your Email" required>
<textarea name="message" rows="5" placeholder="Your Message"

required></textarea>

<input type="submit" name="submit" class="send-btn" value="Send"> </form>

<!--contact section end-->

<script type="text/javascript"> if(window.history.replaceState){

window.history.replaceState(null, null, window.location.href);

}

</script>

```
</body>
</html>
```

sendemail.php

```
<?php
use PHPMailer\PHPMailer\PHPMailer;

require_once 'phpmailer/Exception.php'; require_once 'phpmailer/PHPMailer.php';
require_once 'phpmailer/SMTP.php';

$mail = new PHPMailer(true);

$alert = '';

if(isset($_POST['submit'])){
$name = $_POST['name'];
$email = $_POST['email'];
$message = $_POST['message']; try{
$mail->isSMTP();
$mail->Host = 'smtp.gmail.com';
```

```
$mail->SMTPAuth = true;

$mail->Username = 'piedpiper708@gmail.com'; // Gmail address which you want to use as SMTP server

$mail->Password = 'piedpiper123'; // Gmail address Password

$mail->SMTPSecure = PHPMailer::ENCRYPTION_STARTTLS;

$mail->Port = '587';



$mail->setFrom('piedpiper708@gmail.com'); // Gmail address which you used as SMTP server

$mail->addAddress('piedpiper708@gmail.com'); // Email address where you want to receive emails (you can use any of your gmail address including the gmail address which you used as SMTP server)

$mail->isHTML(true);

$mail->Subject = 'Message Received (Contact Page)';

$mail->Body = "<h3>Name : $name <br>Email: $email <br>Message : $message</h3>";



$mail->send();

$alert = '<div class="alert-success">

<span>Message Sent! Thank you for contacting us.</span> </div>';

}
```

7.3 Output Screenshots

Index.html

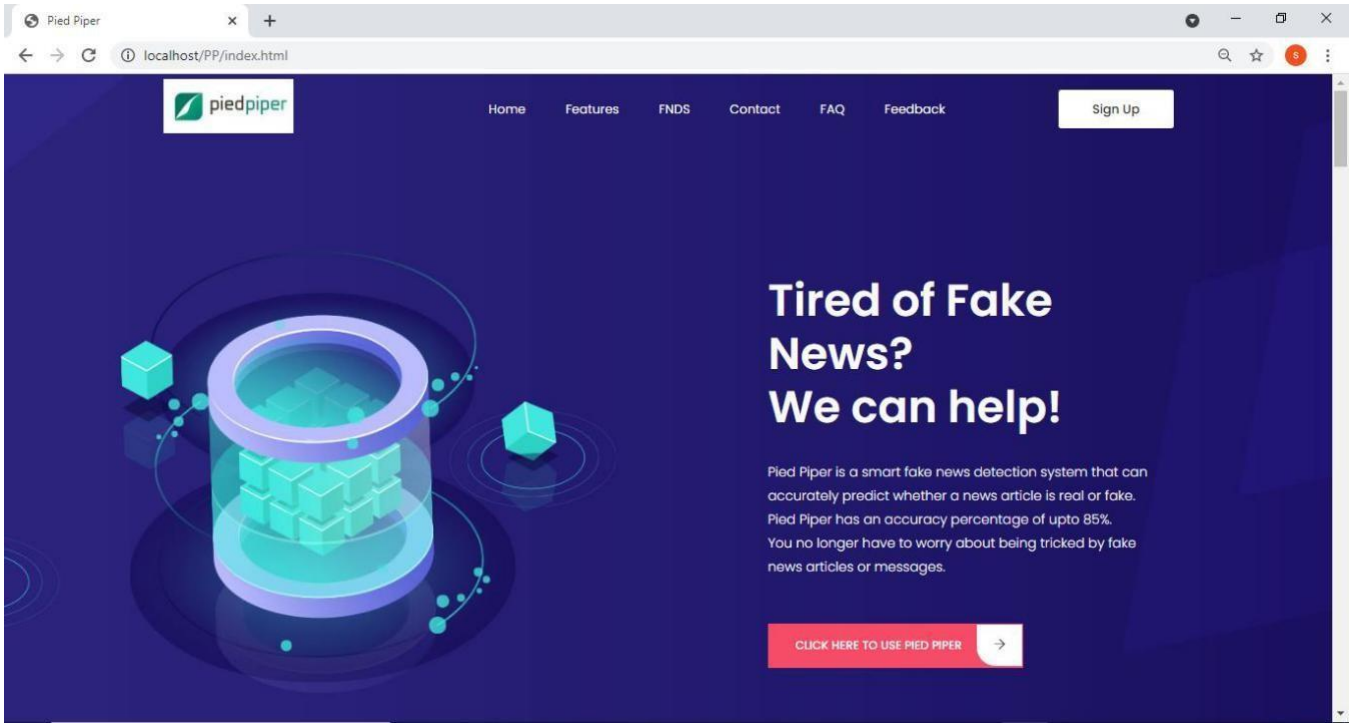


Fig (7.3.1) index.html

feature.html

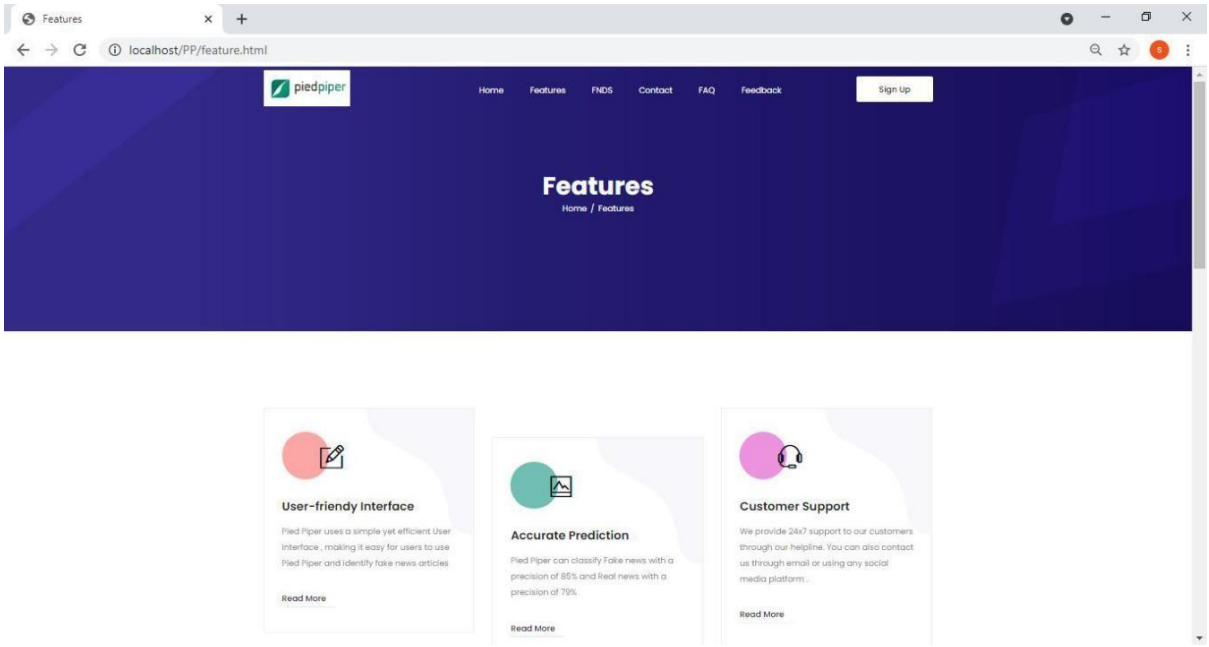


Fig (7.3.2) feature.html

contact.html

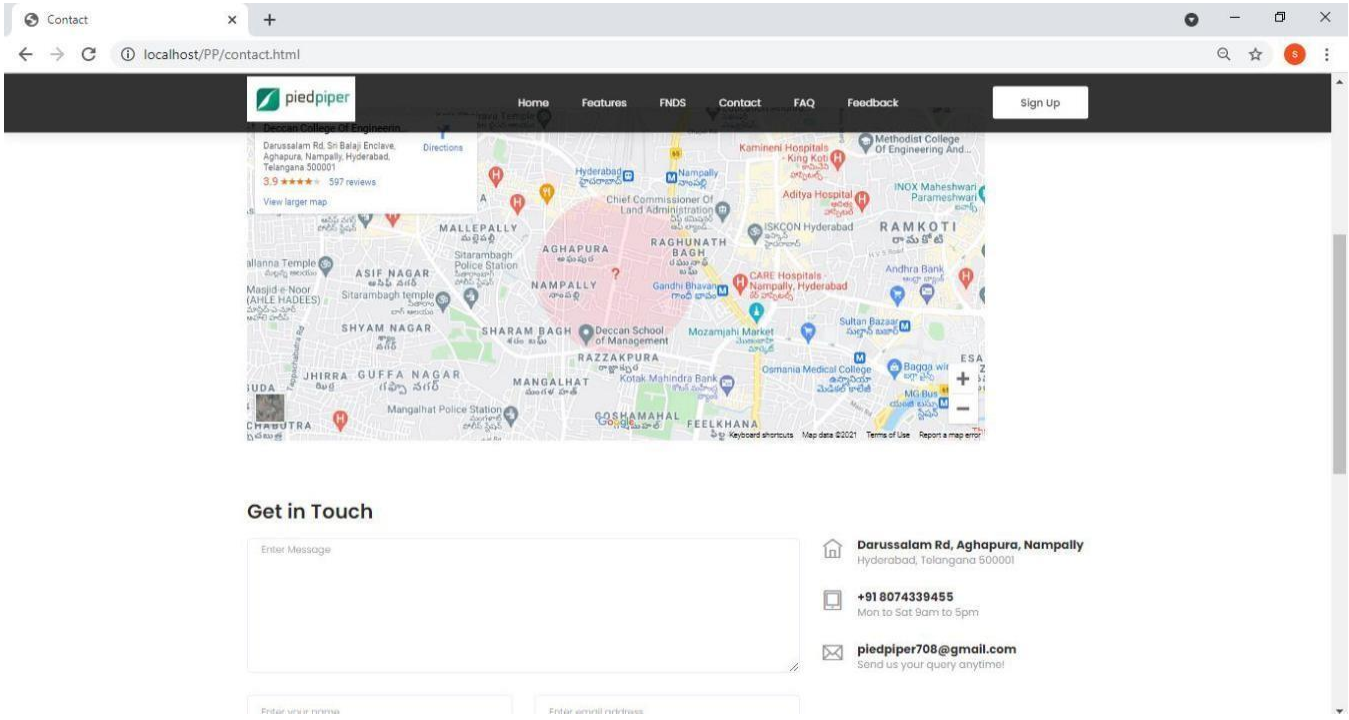


Fig (7.3.3) contact.html

FAQ.html

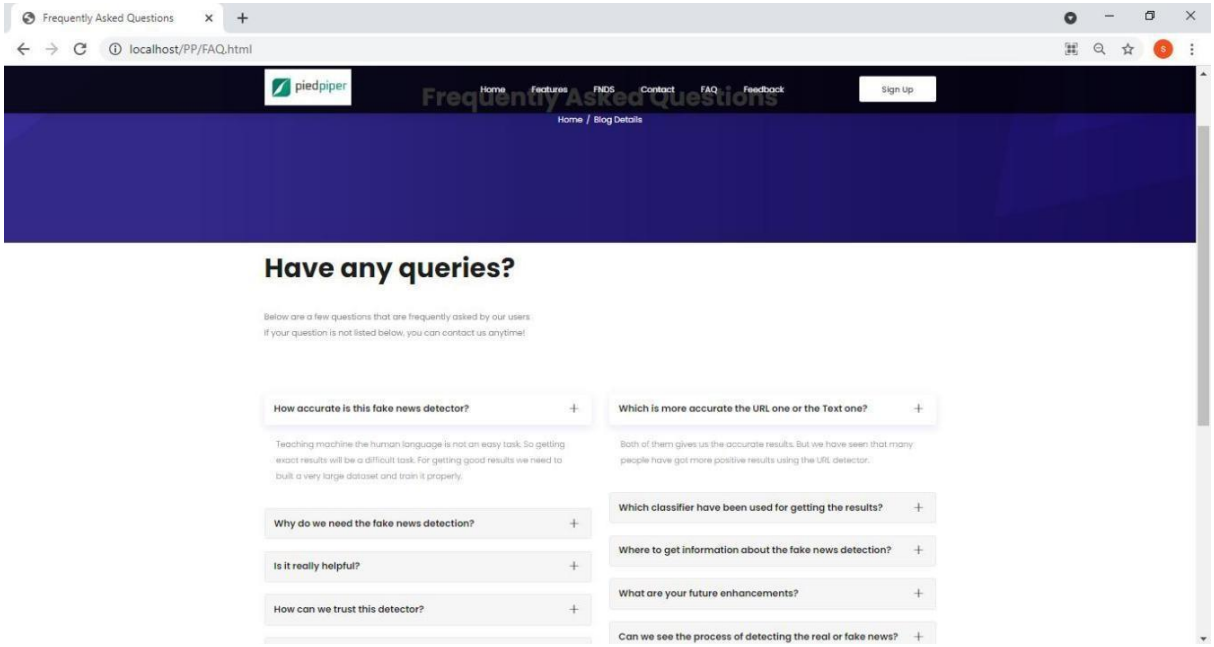


Fig (7.3.4) FAQ.html

Feedback Form

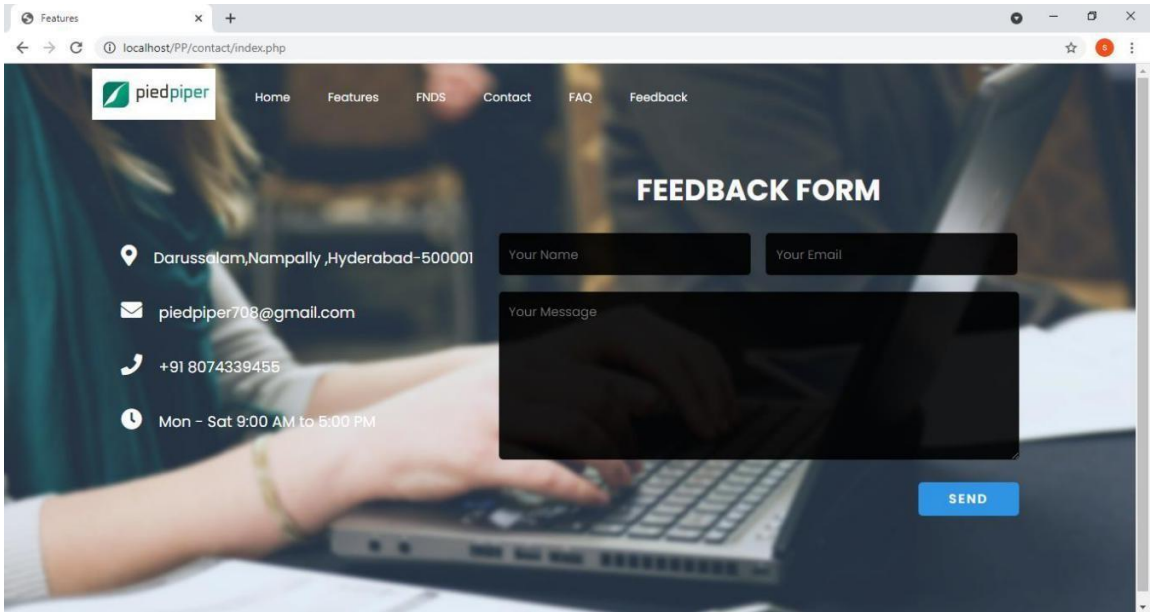


Fig (7.3.5) Feedback Form

Login Page

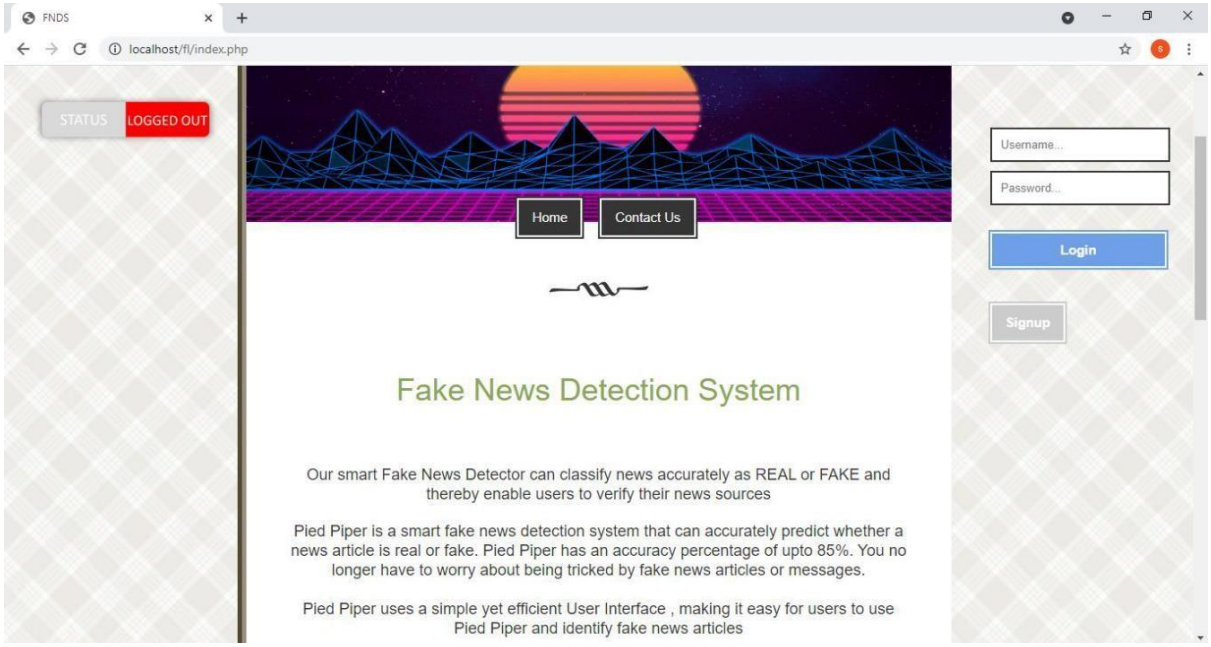


Fig (7.3.6) Login Page

Signup.php

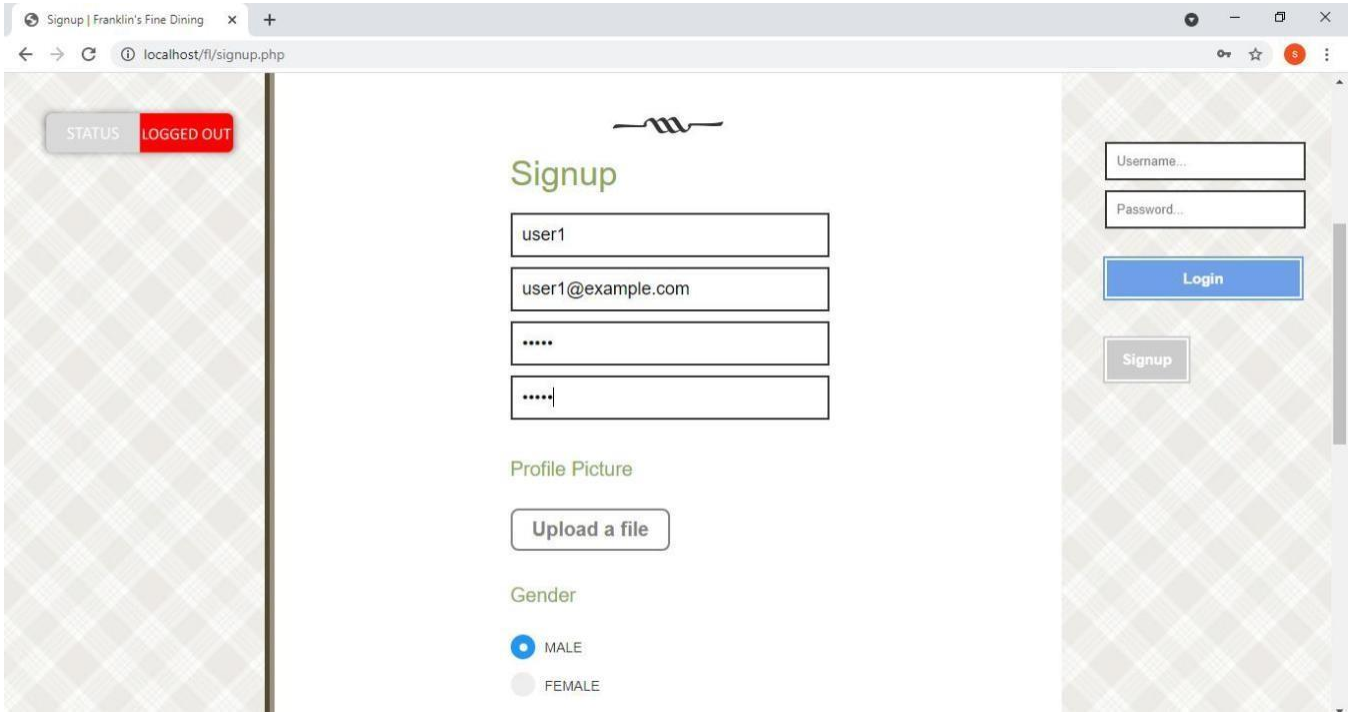


Fig (7.3.7) Signup Page

profile.php

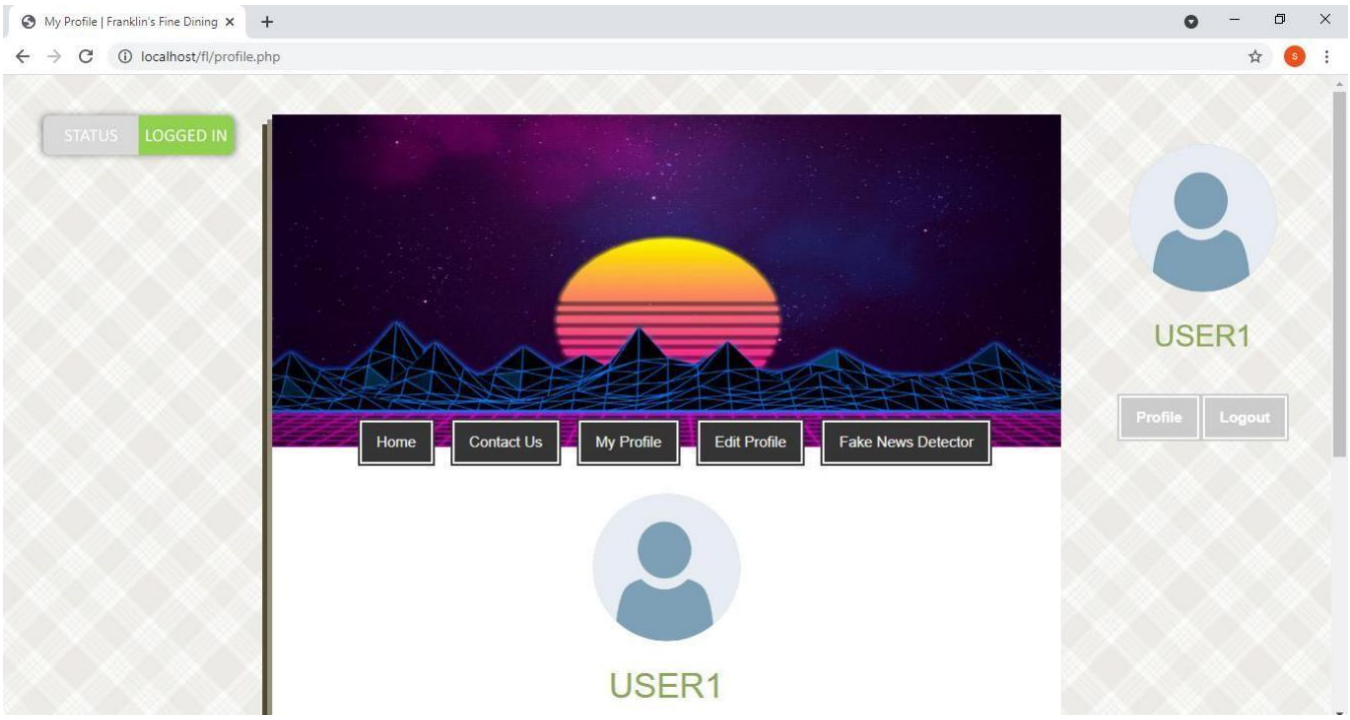


Fig (7.3.8) My Profile Page

edit-profile.php

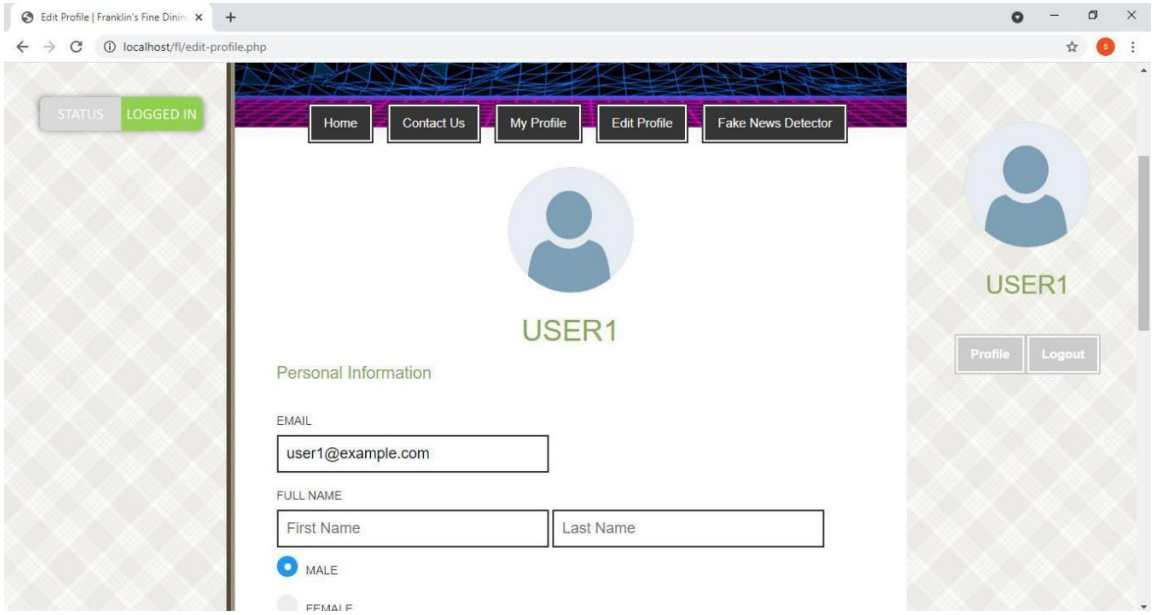


Fig (7.3.9) Edit Profile Page

Chapter 8

Testing

8.1 Testing Strategies and Methodologies

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, subassemblies, 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. Testing is the debugging program is one of the most critical aspects of the computer programming triggers, without programming that works, the system would never produce an output of which it was designed. Testing is best performed when user development is asked to assist in identifying all errors and bugs. The sample data are used for testing. It is not quantity but quality of the data used the matters of testing. Testing is aimed at ensuring that the system was accurately an efficiently before live operation commands.

• Testing objectives

The main objective of testing is to uncover a host of errors, systematically and with minimum effort and time. Stating formally, we can say, testing is a process of executing a program with intent of finding an error.

1) A successful test is one that uncovers an as yet undiscovered error.

2) A good test case is one that has probability of finding an error, if it exists.

3) The test is inadequate to detect possibly present errors.

4) The software more or less confirms to the quality and reliable standards.

• Levels of Testing

In order to uncover present in different phases we have the concept of levels of testing. Tests are grouped together based on where they are added in SDLC or the by the level of detailing they contain. In general, there are four levels of testing: unit testing, integration testing, system testing, and acceptance testing. The purpose of Levels of testing is to make software testing systematic and easily identify all possible test cases at a particular level.

There are many different testing levels which help to check behaviour and performance for software testing. These testing levels are designed to recognize missing areas and reconciliation between the development lifecycle states. In SDLC models there are characterized phases such as requirement gathering, analysis, design, coding or execution, testing, and deployment.

The Basic levels of Testing:

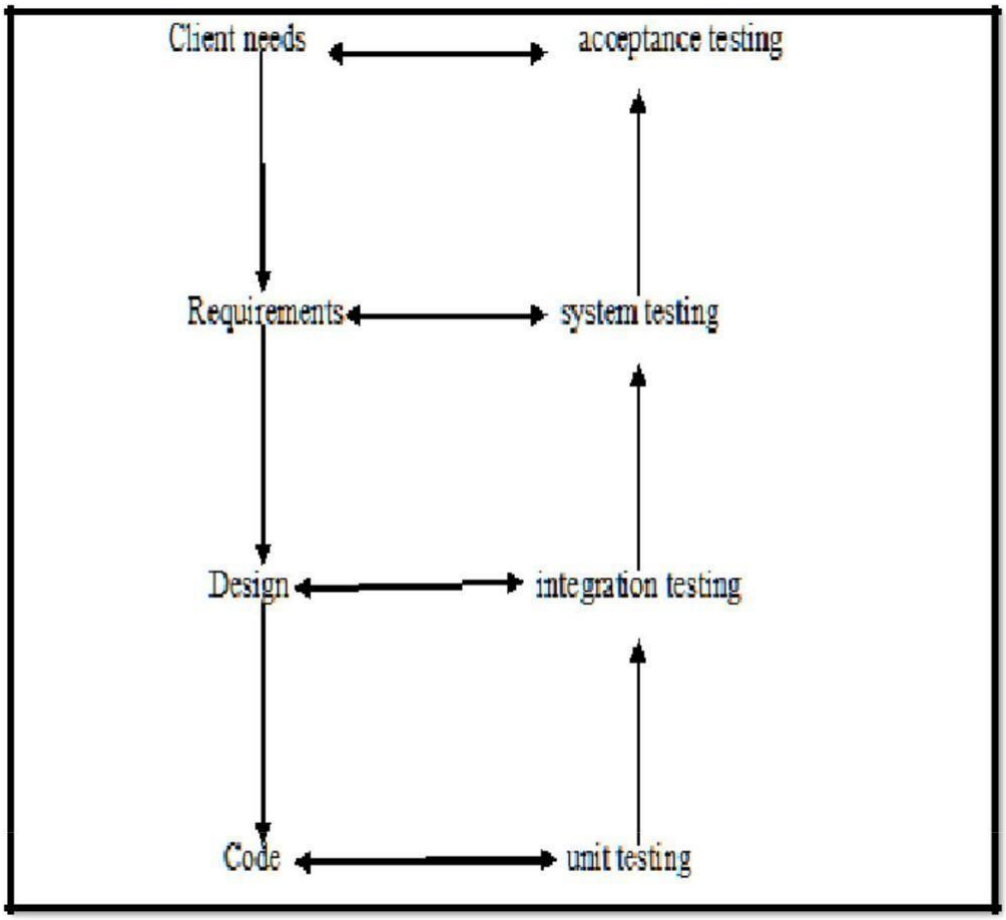


Fig (8.1.1): Levels of Testing

8.1.1 Code testing

This examines the logic of the program. For example, the logic for updating various sample data and with the sample files and directories were tested and verified.

8.1.2 Specification Testing

Executing this specification starting what the program should do and how it should performed under various conditions. Test cases for various situation and combination of conditions in all the modules are tested.

8.1.3 Unit testing

In the unit testing we test each module individually and integrate with the overall system. 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.

Unit testing focuses verification efforts on the smallest unit of software design in the module. This is also known as module testing. The module of the system is tested separately. This testing is carried out during programming stage itself. In the testing step each module is found to work satisfactorily as regard to expected output from the module. There are some validation checks for fields also. For example the validation check is done for varying the user input given by the user which validity of the data entered. It is very easy to find error debut the system. Each Module can be tested using the following two Strategies:

- ➢ Black Box Testing
- ➢ White Box Testing

8.1.4 Black Box Testing

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.

Fig (8.1.4.1): Black Box Testing

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. Black box testing is a software testing technique in which functionality of the software under test (SUT) is tested without looking at the internal code structure, implementation details and knowledge of internal paths of the software. This type of testing is based entirely on the software requirements and specifications. In Black Box Testing we just focus on inputs and output of the software system without bothering about internal knowledge of the software program. The above Black Box can be any software system you want to test. For example: an operating system like Windows, a website like Google, a database like Oracle or even your own custom application. Under Black Box Testing, you can test these applications by just focusing on the inputs and outputs without knowing their internal code implementation.

8.1.5 White Box Testing

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

White Box Testing is the testing of a software solution's internal coding and infrastructure. It focuses primarily on strengthening security, the flow of inputs and outputs through the application, and improving design and usability. White box testing is also known as clear, open, structural, and glass box testing.

It is one of two parts of the "box testing" approach of software testing. Its counter-part, black box testing, involves testing from an external or end-user type perspective. On the other hand, White box testing is based on the inner workings of an application and revolves around internal testing. The term "white box" was used because of the see-through box concept. The clear box or white box name symbolizes the ability to see through the software's outer shell (or "box") into its inner workings. Likewise, the "black box" in "black box testing" symbolizes not being able to see the inner workings of the software so that only the end-user experience can be tested. White box testing involves the testing of the software code for the following:

- ➢ Internal security holes
- ➢ Broken or poorly structured paths in the coding processes
- ➢ The flow of specific inputs through the code
- ➢ Expected output
- ➢ The functionality of conditional loops
- ➢ Testing of each statement, object and function on an individual basis

The testing can be done at system, integration and unit levels of software development. One of the basic goals of white box testing is to verify a working flow for an application. It involves testing a series of predefined inputs against expected or desired outputs so that when a specific input does not result in the expected output, you have encountered a bug.

8.1.6 System testing

Once the individual module testing is completed, modules are assembled and integrated to perform as a system. The top down testing, which began from upper level to lower level module, was carried out to check whether the entire system is performing satisfactorily. System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration-oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

There are three main kinds of System testing:

1) Alpha Testing

2) Beta Testing

3) Acceptance Testing

Alpha Testing: This refers to the system testing that is carried out by the test team with the Organization.

Beta Testing: This refers to the system testing that is performed by a selected group of friendly customers -

Acceptance Testing: This refers to the system testing that is performed by the customer to determine whether or not to accept the delivery of the system

8.2 Test Cases

The following cases were carried out and the respective results were encountered for each test case: -

8.2.1 Test Case for Classification of News Article URL (Naïve Bayes)

Description	Expected Output	Actual Output	Remark Pass/Fail	Date	Name of Creator
Enter URL of a news article	News Classified as REAL	News Classified as FAKE	Fail	18th Feb,2021	Syed Saifuddin
Enter URL of a news article	News Classified as REAL	News Classified as REAL after modifying the dataset	Pass	18th Feb,2021	Syed Saifuddin

Table 8.2.1: Test Case for Classification of News Article URL (Naïve Bayes)

8.2.2 Test Case for Classification of News Article Text (Max-Entropy)

Description	Expected Output	Actual Output	Remark Pass/Fail	Date	Name of Creator
Enter text of a news article	News Classified as REAL	News Classified as FAKE	Fail	20th Feb,2021	Atif Ali Khan
Enter text of a news article	News Classified as REAL	News Classified as REAL	Pass	20th Feb,2021	Atif Ali Khan

Table 8.2.2: Test Case for Classification of News Article Text (Max-Entropy)

8.2.3 Test Case for Feedback Form Submission

Description	Expected Output	Actual Output	Remark Pass/Fail	Date	Name of Creator
Enter name,email ID and feedback	Mail received with the contents of the form	No mail received	Fail	22th Feb,2021	Mohd Misbah Uddin
Enter name,email ID and feedback	Mail received with the contents of the form	After changing the mail settings, mail was received with the contents of the form	Pass	22th Feb,2021	Mohd Misbah Uddin

Table 8.2.3: Test Case for Feedback Form Submission

8.2.4 User Creation and Validation

Description	Expected Output	Actual Output	Remark Pass/Fail	Date	Name of Creator
Create a new user and try signing in	Login Success, User ID and password are validated	Incorrect ID/Password	Fail	24th Feb,2021	Atif Ali Khan
Create a new user and try signing in	Login Success, User ID and password are validated	After debugging the login code , login was a Success, User ID and password validated	Pass	24th Feb,2021	Atif Ali Khan

Table 8.2.4: User Creation and Validation

Chapter 9

Conclusion

Fake news is a major problem in today's modern world, where social media is a major news source for everyone. Since social media provides free access to news, it is very easy for fake news to spread and there exists no system to detect and curb fake news. Fake news spreads like wildfire , the users have no idea whether the news they are reading is REAL or FAKE.

This project provides a means for fake news to be detected by means of supervised machine learning algorithms such as Naïve Bayes and Max-Entropy Classifiers. These algorithms coupled with the datasets used provide upto 85% accuracy in classification of news articles as REAL or FAKE. It is very difficult to achieve 100% accuracy as the growth in information and data on the internet is exponential.

The accuracy of the classifier cannot be 100% , however the accuracy percentage can be increased by utilising a dataset that has a very large collection of REAL and FAKE news articles from around the world . The accuracy can also be increased by using a machine learning algorithm that can overcome the flaws of both Naïve Bayes and Max-Entropy Classifier.

Chapter 10

Future Enhancements

Fake news detector designed is able to classify news articles accurately . However it does not have 100% accuracy , it is possible to achieve an accuracy percentage of close to 100% by linking the classifier that houses a very large collection of REAL and FAKE articles , such as “Event-Registry”.

“Event-Registry” has a very large collection of news articles , however it is a paid service .New users are eligible for a 30-days free trial, after which they need to pay on a monthly basis in order to access it’s benefits.

The integration of Event-Registry can prove to be a tedious task due to the complexity of the collection , however once integrated it can boost the performance of the Fake News Detector significantly.

Chapter 11

References

1. Ammara Habib, Muhammad Zubair Asghar, Adil Khan, Anam Habib, AurangzebKhan, “False information detection in online content and its role in decision making: a ystematic literature review”, Springer-Verlag GmbH Austria, part of Springer Nature 2019.

2. Thota, Aswini; Tilak, Priyanka; Ahluwalia, Simrat; and Lohia, Nibrat (2018) "Fake News Detection: A Deep Learning Approach," SMU Data Science Review: Vol. 1 : No. 3 , Article 10.

3. Chaowei Zhang, Ashish Gupta, Christian Kauten, Amit V Deokar, Xiao Qin, “Detecting fake news for reducing misinformation risks using analytics approaches”, European Journal of Operational Research Elsevier 279 (2019).

4. Dinesh Kumar Vishwakarma, Deepika Varshney, Ashima Yadav, “Detection and veracity analysis of fake news via scrapping and authenticating the web search”, Cognitive Systems Research 58.

5. Hossein Jahankhani, Thulasirajh Jayaraveendran, and William Kapuku-Bwabw, “Improved Awareness on Fake Websites and Detecting Techniques”, ICGS3/e- Democracy 2011, LNICST 99.

6. https://www.hindawi.com/journals/complexity/2020/8885861/

7. https://ieeexplore.ieee.org/document/8546944

8. https://ieeexplore.ieee.org/abstract/document/920749

9. https://www.kdd.org/exploration_files/19-1-Article2.pdf

10. https://www.kaggle.com/c/fake-news

11. https://www.ijert.org/fake-news-detection-using-machine-learning-algorithms

12. https://matheo.uliege.be/bitstream/2268.2/8416/1/s134450_fake_news_detection_using_machine_learning.pdf

13. https://www.sciencedirect.com/science/article/pii/S2667096820300070

14. https://ieee-dataport.org/open-access/fnid-fake-news-inference-dataset

15. https://datasetsearch.research.google.com/search?query=Fake%20Content%20Detection&docid=L2cvMTFrZHF6dDF2ZA%3D%3D

16. https://www.aclweb.org/anthology/P17-2067/

17. https://www.researchgate.net/post/Fake_news_dataset_for_COVID_tweets_or_text