SAVITRIBAI PHULE PUNE UNIVERSITY

A PROJECT REPORT ON

Falsified News Detection using Deep Learning Approach

SUBMITTED TOWARDS THE
PARTIAL FULFILLMENT OF THE REQUIREMENTS OF

BACHELOR OF ENGINEERING(Computer Engineering)

BY

Ghanshyam Patil	Exam No: 71818487E
Ketan Patel	Exam No: 71818371B
Sudesh Pawar	Exam No: 71818502B
Piyush Pawar	Exam No: 71818499J

Under The Guidance of

Prof.S.P. Khedkar
Prof.B.F. More

DEPARTMENT OF COMPUTER ENGINEERING
Modern Education Society's College of Engineering
19, Late Prin. V.K. Joag Path, Wadia College Campus, Pune- 411001

Modern Education Society's College of Engineering,Pune-01

CERTIFICATE

This is to certify that the Project Entitled

Falsified News Detection using Deep Learning Approach

Submitted by

Ghanshyam Patil	**Exam No: 71818487E**
Ketan Patel	**Exam No: 71818371B**
Sudesh Pawar	**Exam No: 71818502B**
Piyush Pawar	**Exam No: 71818499J**

is a bonafide work carried out by Students under the supervision of Prof.S.P Khedkar and Prof.B.F.More and it is submitted towards the fulfillment of the requirement of Bachelor of Engineering (Computer Engineering).

Prof.S.P.Khedkar
Internal Guide
Dept. of Computer Engg.

Prof.B.F.More
Internal Guide
Dept. of Computer Engg.

Dr.(Mrs.) N. F. Shaikh
H.O.D
Dept. of Computer Engg.

Dr.S.S.Sarawade
Principal
MESCOE Pune.

Sign of Internal Examiner

Sign of External Examiner

PROJECT APPROVAL SHEET

A Project Title

Falsified News Detection using Deep Learning Approach

Is successfully completed by

Ghanshyam Patil	Exam No:71818487E
Ketan Patel	Exam No:71818371B
Sudesh Pawar	Exam No:71818502B
Piyush Pawar	Exam No:71818499J

at

DEPARTMENT OF COMPUTER ENGINEERING

Modern Education Society's College of Engineering,Pune

SAVITRIBAI PHULE PUNE UNIVERSITY,PUNE

ACADEMIC YEAR 2020-21

Prof.S.P.Khedkar Internal Guide Dept. of Computer Engg.	Prof.B.F.More Internal Guide Dept. of Computer Engg.

Dr.(Mrs.) N. F. Shaikh
H.O.D.
Dept. of Computer Engg.

Abstract

Technology is changing the way that we live our lives. With the advances in technology Social Media attains a lot of attention of the people around us. With this fake news for various commercial and political purposes has been appearing in large numbers and widespread in the online world.. Users can get infected easily by these online fake news, which has brought about tremendous effects on the society. An important goal is to stop the spread of rumors and focus on the correct, authenticated news articles.This paper focuses on techniques such as natural language processing (NLP) techniques for text analytics and train deep learning models for detecting fake news based on news title or news content. Using this technique,text pre-processing such as regular expression, tokenization, lemmatization and stop words removal are used before vectorizing them into N-gram vectors or sequence vectors using terms frequency inverse document frequency (TF-IDF) or one-hot encoding respectively. With the help of Machine learning and natural language processing, author tried to aggregate the news and later determine whether the news is real or fake using deep neural networks. Experimental evaluation yields the best performance using Term Frequency-Inverted Document Frequency (TF-IDF) as feature extraction technique.

Keywords- Fake News, Social Media, Natural Language Processing, Deep Learning, Neural Network

Acknowledgment

It gives us great pleasure and satisfaction in presenting the final project report on ‘Falsified News Detection using Deep Learning Approach’.

We have furthermore to thank our Guides Prof.S.P.Khedkar and Prof.B.F.More and Computer Department HOD Dr.(Mrs.) N. F. Shaikh to encourage us to go ahead and for continuous guidance.

We would like to thank all those, who have directly or indirectly helped us during of the work.

Ghanshyam Patil
Ketan Patel
Sudesh Pawar
Piyush Pawar
(B.E. Computer Engg.)

Contents

List of Figures

Chapter 1

Introduction

1.1 Project Title

Falsified News Detection using Deep Learning Approach.

1.2 Project Option

Internal project

1.3 Internal Guide

Prof.S.P.Khedkar and Prof.B.F.More

1.4 Problem Definition

Fake news also called as bluff news occupies large sphere of cyber security space today's date. Cyber technology's broad reach and quick outspread contributes to its risks. Publicity through such fake news on internet today has been adopted by institutions as well as individuals for various reasons. emotional news is created and spread through social media to realize intended end. On the other side, it may also require narration of a true fact however being purposely overestimated. This may also include labelling the webpages with ambiguous title or tag-lines in order to grab attention of readers. Such fake information may lead in committing offences, social unrest, financial frauds upon such misrepresentation, political gain, to extend number of readers, gain revenue related to click, etc. This may also affect the affect the importance of great journalism. The problem is to identify the genuineness of the news and online content and at the same time it is also important to identify the bots involved in spreading rumours.

1.5 Motivation

In Today's world, anybody can post the content over the internet. It is harmful for the society to believe on the rumors and pretend to be a news. The fake news can lead to mob lynching, riots and negative or wrong information. Fake news detection is made to

stop the rumors that are being spread through the various platforms whether it be social media or messaging platforms Fake news tries to stop such false information thereby protecting the society from this kind of violence. Satirical Cues are prevalent in false news they can help in detection.

1.6 Goals and Objectives

1. Most of the smart phone users prefer to read the news via social media over internet.
2. Lots of Falsified news is being circulated on social media
3. The need of an hour is to stop the rumors and focus on the correct, authenticated news articles.
4. The question is how to authenticate the difference.
5. The main objective with this project is to classify the news into fake or real.
6. With the help of Deep Learning and Neural Networks we can classify it.

1.7 Conferences/Journals where papers can be published

- **IJRTE**
- **Conferences/workshops in IITs/SPPU/Other Central Universities**
- **IJCT**

Chapter 2

Literature Survey

2.1 Review of Conference/Journal Papers supporting Project idea

Paper Name: Fake News Detection using Deep Learning[1]
Author: S. H. Kong, L. M. Tan, K. H. Gan and N. H. Samsudin

Description: This study aims to apply NLP techniques for text analytics and train deep learning models for detecting fake news based on news title or news content. Solution proposed in this study aims to be applied in real-world social media and eliminate the bad experience for user to receive misleading stories. For NLP techniques, text pre-processing such as regular expression, tokenization, lemmatization and stop words removal are used before vectorizing them into N-gram vectors or sequence vectors using term frequency inverse document frequency or one-hot encoding respectively. Then, TensorFlow is chosen as the framework to be used with built-in Keras libraries which are enough to build deep learning models. Results show that models trained with news content can achieve better performance with computation time being sacrificed while models trained with news title require less computation time to achieve good performance. Also, overall performance of models fed with N-gram vectors are slightly better than models fed with sequence vectors.

Paper Name: A Multi-semantics Classification Method Based on Deep Learning for Incredible Messages on Social Media[2]
Author: L. Wu, Y. Rao, H. Yu, Y. Wang and N. Ambreen

Description: This work mainly focuses on rumors or fake news and achieves some success to detect them. The existing problem is that messages have different types on social media, and rumors or fake news cannot represent all incredible messages. Based on this, the author divides messages on social media into five types based on three dimensions of information evaluation metrics. And a novel method is proposed based on deep learning for classifying the five types of incredible messages on social media.More specifically, they use attention mechanism to obtain deep text semantic features and strengthen emotional semantics features, meanwhile, construct universal metadata as auxiliary features, concatenating them for incredible messages classification. A series of experiments

on two representative real-world datasets demonstrate that the proposed method outperforms the state-of-the-art methods.

Paper Name: Detecting Misleading Information on COVID-19[3]
Author: M. K. Elhadad, K. F. Li and F. Gebali

Description: This paper addresses the problem of detecting misleading information specifically related to COVID-19. The authors propose a detection model that relies on the WHO, UNICEF, and the UN as sources of information, as well as epidemiological material collected from a range of fact-checking websites. Obtaining data from reliable sources should assure their validity. They use this collected ground-truth data to build a detection system that uses machine learning to identify misleading information. 10 ML Algorithms, with 7 feature extraction techniques, are used to construct a voting ensemble machine learning classifier. 5-fold cross-validation is performed to check the validity of the collected data and report the evaluation of twelve performance metrics. The evaluation results indicate the quality and validity of the data and their effectiveness in constructing models to detect misleading information.

Paper Name: Fake News Stance Detection Using Deep Learning Architecture (CNN-LSTM)[4]
Author: M. Umer, Z. Imtiaz, S. Ullah, A. Mehmood, G. S. Choi and B.W. On

Description: In this paper, To address the issue of fake news, a hybrid Neural Network architecture, that combines the capabilities of CNN and LSTM, is used with 2 different dimensionality reduction approaches, PCA and Chi-Square. This work proposed to employ the dimensionality reduction techniques to reduce the dimensionality of the feature vectors before passing them to the classifier. To develop the reasoning, this work acquired a dataset from the Fake News Challenges website which has four types of stances: agree, disagree, discuss, and unrelated. The nonlinear features are fed to PCA and chi-square which provides more contextual features for detection. The motivation of this research is to determine the relative stance of a news article towards its headline. The proposed model improves results by 4% and 20% in terms of Accuracy and F1 score. The experimental results show that PCA outperforms

than Chi-square and state-of-the-art methods with 97.8% accuracy.

Paper Name: Big Data and quality data for fake news and misinformation detection [5]
Author: F. T. Asr and M. Taboada

Description: Fake news has become an important topic of research in a variety of disciplines including linguistics and computer science. In this paper, authors explain how the problem is approached from the perspective of natural language processing, with the goal of building a system to automatically detect misinformation in news. The main challenge in this line of research is collecting quality data, i.e. instances of fake and real news articles on a balanced distribution of topics. They reviewed available datasets and repositories as a contribution of our lab to the community. They make available the full text of the news articles, together with veracity labels previously assigned based on manual assessment of the articles' truth content. They also performed a topic modelling experiment to elaborate on the gaps and sources of imbalance in currently available datasets to guide future efforts.

Paper Name: FakeDetector: Effective Fake News Detection with Deep Diffusive Neural Network[6]
Author: Jiawei, Zhang & Dong, Bowen & Yu, Philip

Description: This paper aims at investigating the principles, methodologies and algorithms for detecting fake news articles, creators and subjects from online social networks and evaluating the corresponding performance. This paper addresses the challenges introduced by the unknown characteristics of fake news and diverse connections among news articles, creators and subjects. This paper introduces a novel automatic fake news credibility inference model, namely FAKEDETECTOR. Based on a set of explicit and latent features extracted from the textual information, Authors builds a deep diffusive network model to learn the representations of news articles, creators and subjects simultaneously. Extensive experiments have been done on a real-world fake news dataset to compare FAKEDETECTOR with several state-of-the-art models, and the experimental results have demonstrated the effectiveness of the proposed model.

Paper Name: Fake news detection using deep learning models: A Novel Approach[7]
Author: Sachin Kumar,Rohan Asthana, Shashwat Upadhyay,Nidhi Upreti,Mohammad Akbar

Description: With the ever increase in social media usage, it has become necessary to combat the spread of false information and decrease the reliance of information retrieval from such sources. Social platforms are under constant pressure to come up with efficient methods to solve this problem because users' interaction with fake and unreliable news leads to its spread at an individual level. This spreading of misinformation adversely affects the perception about an important activity, and as such, it needs to be dealt with using a modern approach. In this paper, the authors collected 1356 news instances from various users via Twitter and media sources such as PolitiFact and create several datasets for the real and the fake news stories. Our study compares multiple state-of-the-art approaches such as convolutional neural networks (CNNs), long short-term memories (LSTMs), ensemble methods, and attention mechanisms. We conclude that CNN + bidirectional LSTM ensembled network with attention mechanism achieved the highest accuracy of 88.78%

Paper Name: Fake News Detection: A Deep Learning Approach[8]
Author: Aswini Thota, Priyanka Tilak,Simrat Ahluwalia,Nibrat Lohia

Description: Fake news is defined as a made-up story with an intention to deceive or to mislead. In this paper we present the solution to the task of fake news detection by using Deep Learning architectures. Gartner research predicts that "By 2022, most people in mature economies will consume more false information than true information". The exponential increase in production and distribution of inaccurate news presents an immediate need for automatically tagging and detecting such twisted news articles. However, automated detection of fake news is a hard task to accomplish as it requires the model to understand nuances in natural language. Moreover, majority of the existing fake news detection models treat the problem at hand as a binary classification task, which limits model's ability to understand how related or unrelated the reported news is when compared to the real news. To

address these gaps, we present neural network architecture to accurately predict the stance between a given pair of headline and article body. Our model outperforms existing model architectures by 2.5% and we are able to achieve an accuracy of 94.21% on test data.

Paper Name: FNDNet- A Deep Convolutional Neural Network for Fake News Detection[9]
Author: Rohit Kumar Kaliyar, Anurag Goswami, Pratik Narang, Soumendu Sinha

Description: With the increasing popularity of social media and web-based forums, the distribution of fake news has become a major threat to various sectors and agencies. This has abated trust in the media, leaving readers in a state of perplexity. There exists an enormous assemblage of research on the theme of AI strategies for fake news detection. In the past, much of the focus has been given on classifying online reviews and freely accessible online social networking-based posts. In this work, authors propose a deep convolutional neural network (FNDNet) for fake news detection. Instead of relying on hand-crafted features, the model (FNDNet) is designed to automatically learn the discriminatory features for fake news classification through multiple hidden layers built in the deep neural network. We create a deep convolutional neural network (CNN) to extract several features at each layer. They compare the performance of the proposed approach with several baseline models. Benchmarked datasets were used to train and test the model, and the proposed model achieved state-of-the-art results with an accuracy of 98.36% on the test data. This research will assist researchers in broadening the understanding of the applicability of CNN-based deep models for fake news detection.

Chapter 3

Software Requirement Specification

3.1 Introduction

3.1.1 Purpose & Scope of Document

A software requirements specification (SRS) is a document that is created when a detailed description of all aspects of the software to be built must be specified before the project is to commence. It is important to note that a formal SRS is not always written. In fact, there are many instances in which effort expended on a SRS might be better spent in other software engineering activities.

3.1.2 Scope of Project

Most of the smart phone users prefer to read the news via social media over internet and lots of falsified news is being circulated on social media. The need of the hour is to stop these rumors spread through such falsified news which is being circulated on the social media and focus on the correct, authenticated news articles. The question is how to authenticate the difference and establish the credibility of the news articles. The main scope of this project is to classify the news into fake or real. With the help of machine learning and support vector machine, we can classify the news easily thus completing the objective of the project.

3.1.3 Usage Scenario

Usage scenario of the project is in media sector .The project can be used by news aggregators or directly by audience to analyze the authencity by provinding input as news article.

User profiles

Actors: News Aggregators and News Audience.

Use-cases

Remote customers most frequently use the platform for verification of news from popular sources. The customers are not expected to have a high educational and proficiency level or technical expertise. Hence, the user interfaces is available in a popular international language such as English.

The Admin is expected to have a field appropriate college degree and experience as a System Admin and an additional experience in the IT field. He/She has the privilege to update information in the database and technical expertise in database management. The Admin does not directly interact with the web service but handle the back-end.
Third-party websites can use service in the form of API and can have use case for news analysis and analysis of news resources considering our model as reference.

3.1.4 Assumption & Dependencies

The data will be coming from the news aggregator which will fetch the news data from different trusted websites and sources. We are assuming that the data coming from the trusted sources is accurate. The whole of the output of the system depends on the accuracy of data.

3.2 Functional Requirements

3.2.1 Frontend Subsystem

Description

The Frontend Module of our Project is the part which will be visible to the user i.e. the User will exclusively interact with this module. The Frontend will be a GUI Web Application developed in an appropriate JavaScript Framework for visual aesthetics.
There will be an option to provide input (News Details and News Source) , this input will be passed to the backend and backend will perform the necessary computations on it and give the desired binary output which will be passed back to the frontend.
The Frontend will now display the output received from the backend in a visually formatted manner, with possibly insights why the output was marked as Fake or Real.

Stimulus/Response Sequences

The Frontend will be loaded when the user inputs the address/domain of the project and no additional stimulus is required to load this module.

Functional Requirements

REQ-1: JavaScript enabled Browser
REQ-2: Internet Connectivity.

3.2.2 Backend Subsystem

Description

In the Backend of our proposed project, we will be using Python (Version 3.6 or greater) and TensorFlow for development of the backend system.
Python would be used to load the Dataset and Pandas will be used to convert the dataset (which will be in csv format) into a format easily readable for the TensorFlow models (i.e. Dataframes).

The Dataset could be having some values missing, some incorrect information i.e. the dataset is not clean. Therefore, the data cleaning of the Dataset will be handled by the Python Backend. TensorFlow will be used to develop a Deep Learning model as proposed in the abstract of the project. This model is provided input from the Dataset during Training Phase and from the frontend of the web application during normal use.

Stimulus/Response Sequences

The Backend module will be loaded when the user inputs the news article and clicks on the Analyze button. The backend module is not loaded by default and is loaded only when the frontend makes a function call to the backend.

Functional Requirements

REQ-1: Support for .h5 Models (Available in all Modern Browsers)

3.3 External Interface Requirements

3.3.1 User Interfaces

As this project is a GUI Web Application there will be an option to provide input (News Details and News Source) to user, this input will be passed to the backend and backend will perform the necessary computations on it and give the desired binary output which will be passed back to the frontend .The Frontend will now display the output received from the backend in a visually formatted manner, with possibly insights why the output was marked as Fake or Real.

3.3.2 Hardware Interfaces

The Project itself does not use any Hardware Interface but still it requires a system to run itself. As it is a Machine Learning on Web Based Implementation, this project can be run on any device with internet support. Supported device types include Mobile Phones, Smart Television, Desktops, Laptops etc. Communication protocols to be used is HTTP as the web application will be hosted on a Web Server and not a Native Deployment.

3.3.3 Software Interfaces

Python 3.6, Tensorflow 2.0, BeautifulSoup, JavaScript, HTML, CSS

3.3.4 Communication Interfaces

The requirements associated with communications functions required by this project, include any web browser with JavaScript Support for running the model in backend. Define any pertinent message formatting. Communication standards that will be used is HTTP and there is no file transfer involved in this project so there is no need of FTP. There are no security issues as the project does not require handling of any sensitive or real-time data such as E-Mail or Password.

3.4 Non Functional Requirements:

3.4.1 Performance Requirement

The performance of the detection system must be well.The overall performance of the software will enable the users to work efficiently.Performance of pre-processing of data should be fast.Performance of the feature extraction and working of algorithm must be fast.

3.4.2 Safety Requirement

The application is designed in modules where fake news is detected in stages of the deep learning algorithm. This makes it easier to install and update new functionality if required.

3.4.3 Security Requirement

All data will be pre-processed smoothly with precision and the necessary feature extraction is done accurately so that the output of the algorithm shows accuracy.

3.4.4 Software Quality Attributes:

Our software has many quality attribute that are given below:-

1.Adaptability: This software is adaptable by all users.
2.Availability: This software is freely available to all users.The availability of the software is easy for everyone.
3.Maintainability: After the deployment of the project if any error occurs then it can be easily maintained by the software developer.
4.Reliability: The performance of the software is better which will increase the reliability of the Software.
5.User Friendliness: Since, the software is a GUI application; the output generated is much user friendly in its behavior.
6.Integrity: Integrity refers to the extent to which access to

software by unauthorized persons can be controlled.

7.Testability: The software will be tested considering all the aspects.

3.5 System Requirements

3.5.1 Database Requirements

Data objects that will be managed/manipulated by the software are described in this section. The database entities or files or data structures required to be described. For data objects details can be given as below

3.5.2 Software Requirements

[For User]: **Any Browser with JavaScript Connectivity, Internet Connectivity**

[For Development]: **TensorFlow, Python, Pandas,Numpy,Matplotlib, Flask, ReactJS.**

3.5.3 Hardware Requirements

[For User]: **Any Device with Internet Connectivity and ability to run websites.**

[For Development]: **GPU, CPU**

3.6 Analysis Model

We will be using Waterfall Model in our development and therefore, we are first finalizing the requirements,then we will analyze these before moving towards development and testing.

3.7 System Implementation Plan

Gantt Chart

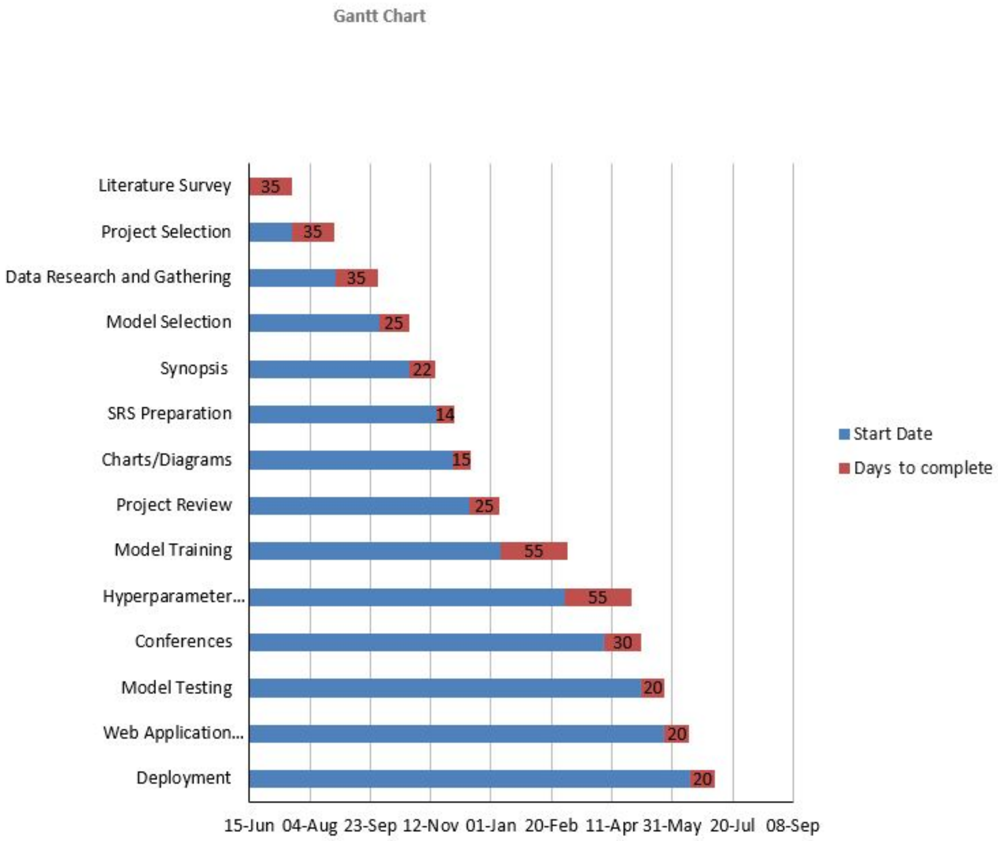


Figure 3.1: Gantt Chart

Chapter 4

System Design

4.1 System Architecture

The System will fetch the news articles from news aggregators in Training Phase and store it in the dataset. This dataset will then be used for training our deep learning model and then, this model will be exported as a .h5 model and will be used in our web application.

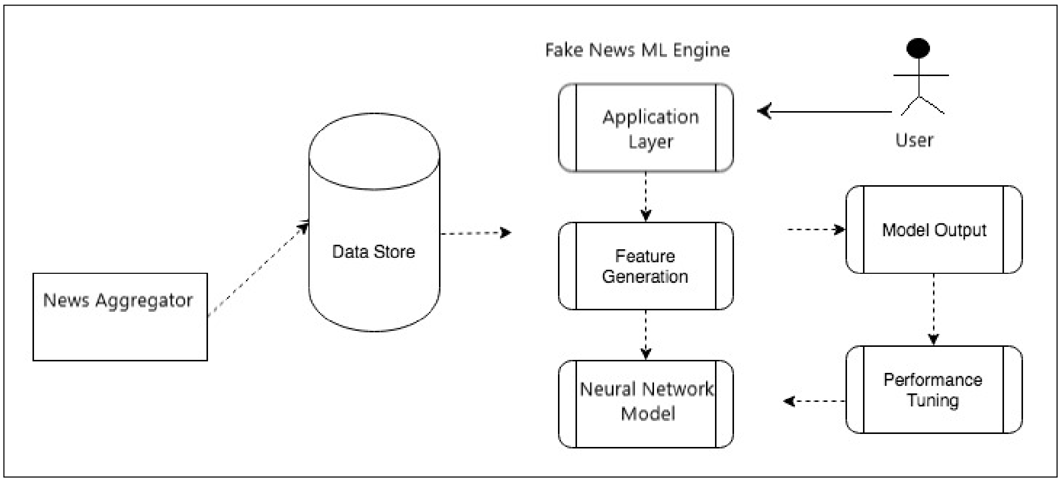


Figure 4.1: Architecture diagram

4.2 Data Flow Diagram

Level 0 Data Flow Diagram

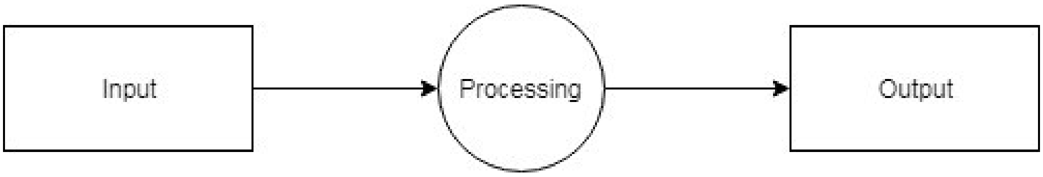


Figure 4.2: DFD Level 0 Diagram

Level 1 Data Flow Diagram

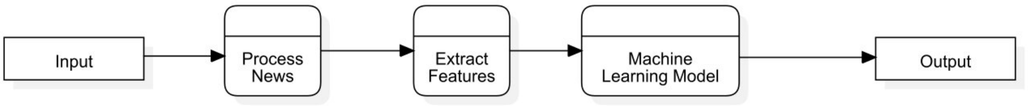


Figure 4.3: DFD Level 1 Diagram

Level 2 Data Flow Diagram

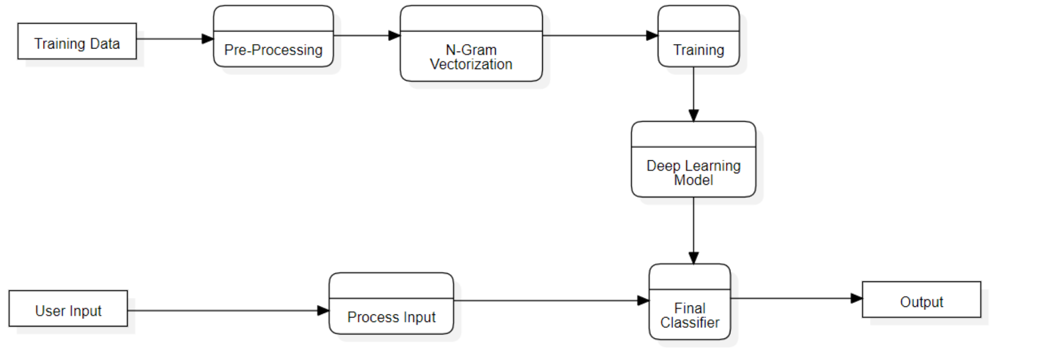


Figure 4.4: DFD Level 2 Diagram

4.3 Entity Relationship Diagrams

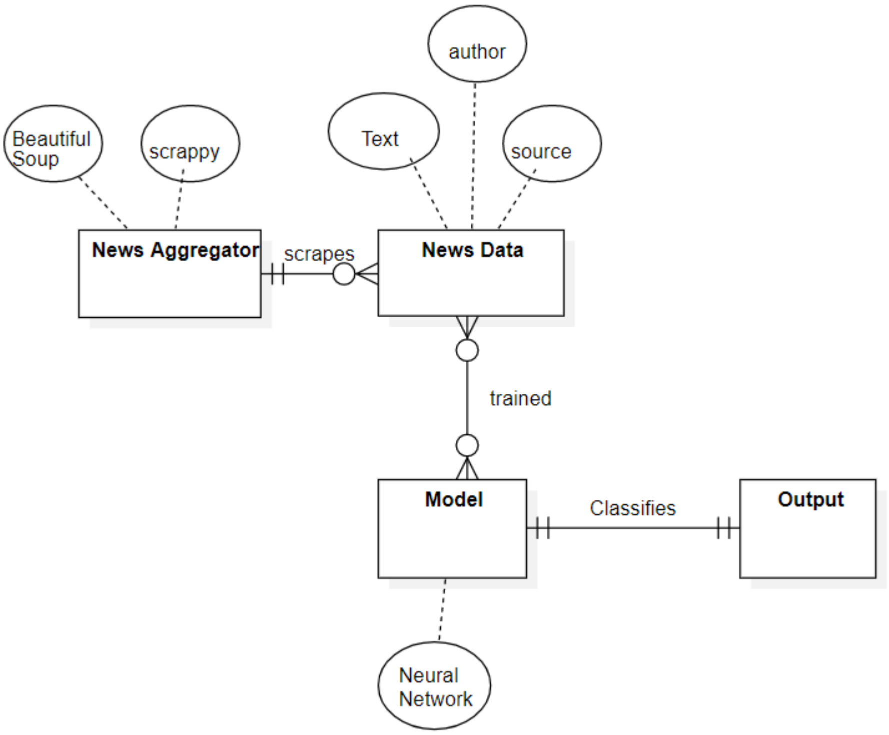


Figure 4.5: ER Diagram

4.4 UML Diagrams

4.4.1 Use Case Diagram

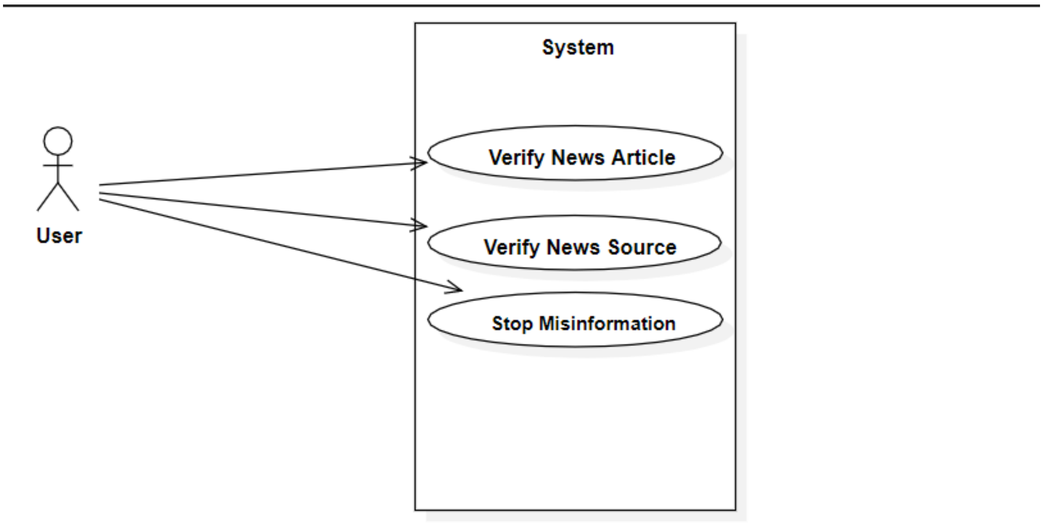


Figure 4.6: Use Case Diagram

4.4.2 Class Diagram

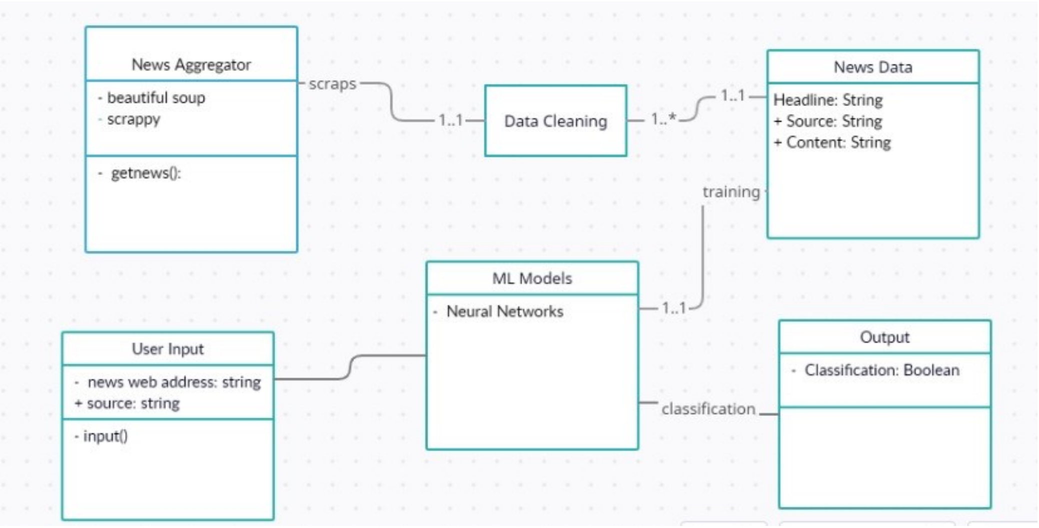


Figure 4.7: Class Diagram

4.4.3 Sequence Diagram

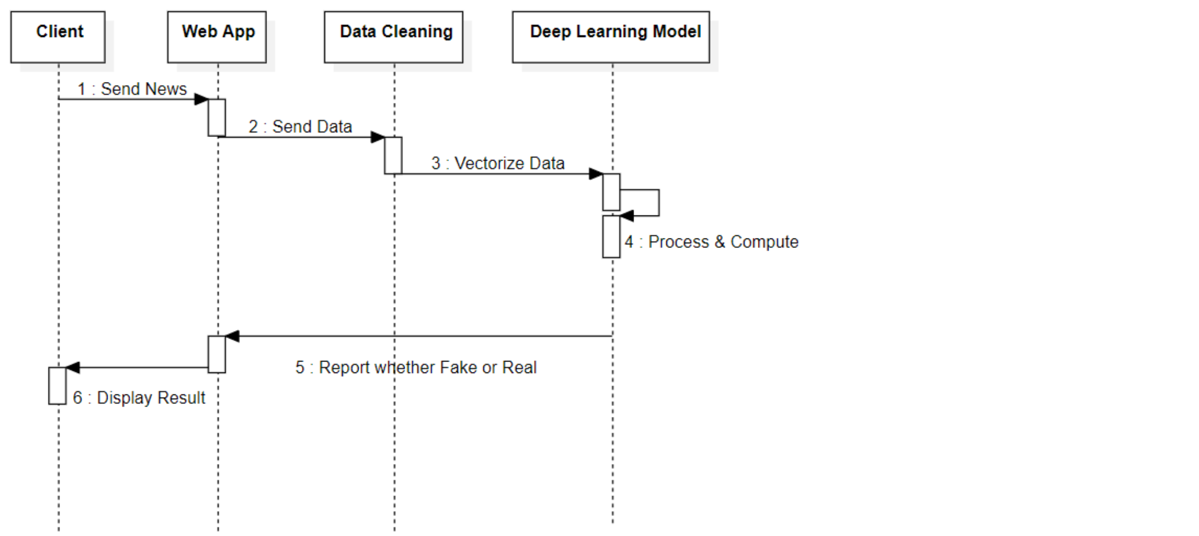


Figure 4.8: Sequence Diagram

4.4.4 State Transition Diagram

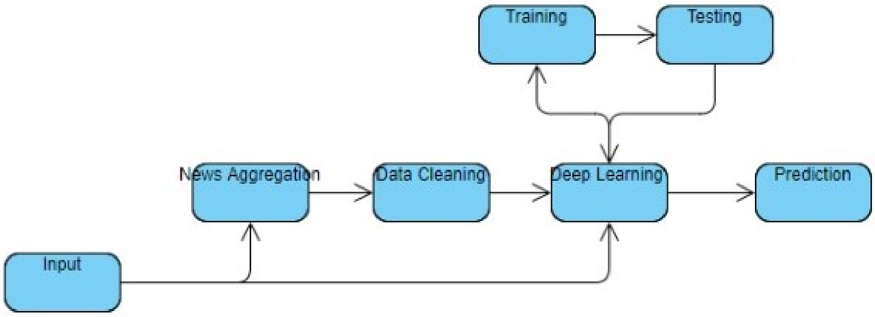


Figure 4.9: State Transition Diagram

4.4.5 Component Diagram

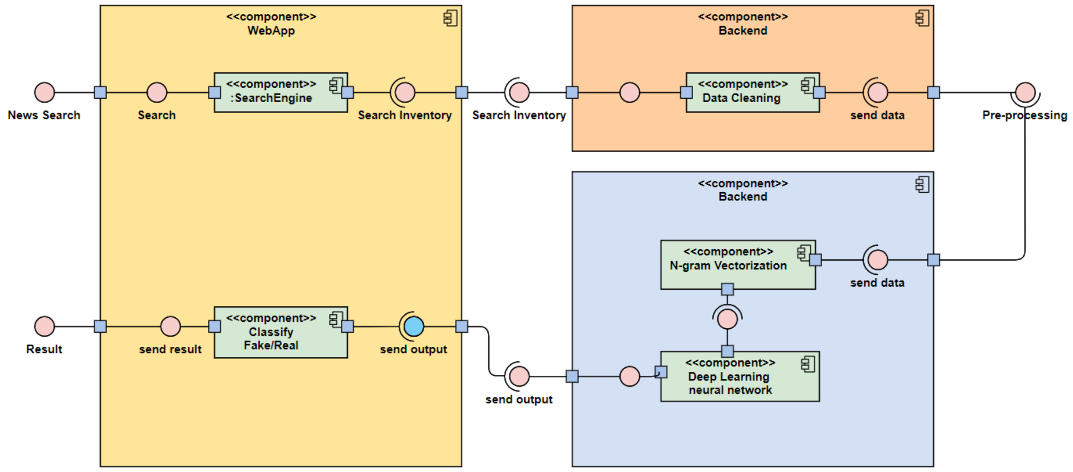


Figure 4.10: Component Diagram

4.4.6 Deployment Diagram

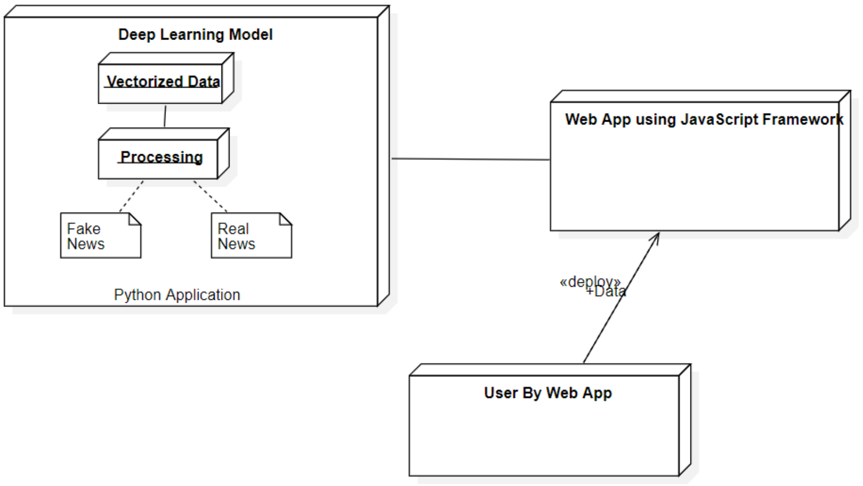


Figure 4.11: Deployment Diagram

4.4.7 Activity Diagram

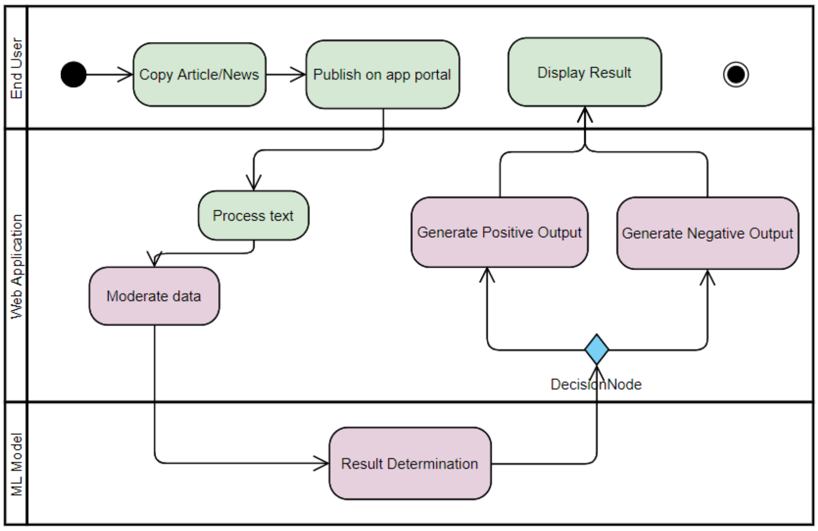


Figure 4.12: Activity Diagram

4.4.8 Package Diagram

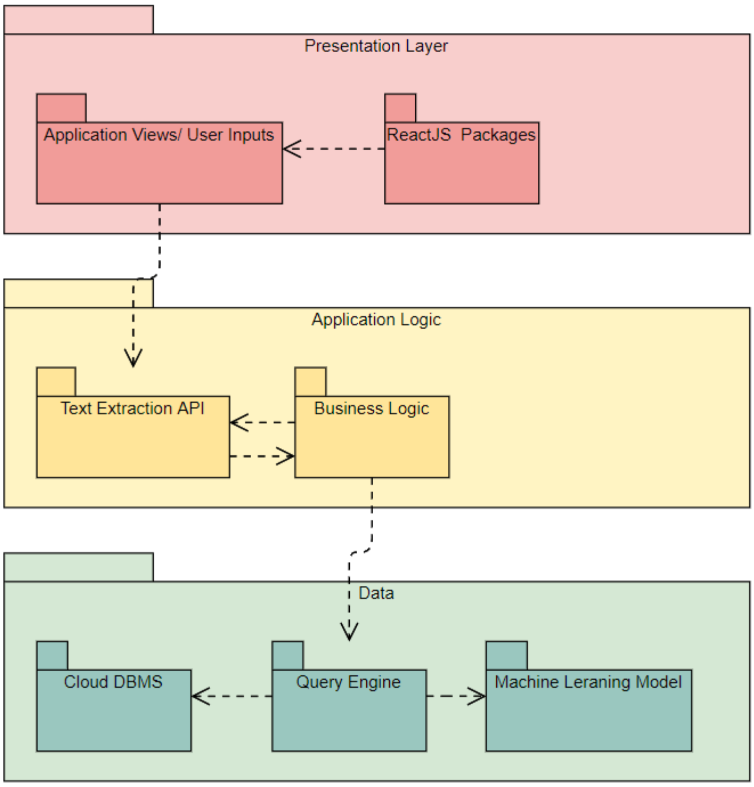


Figure 4.13: Activity Diagram

Chapter 5

Project Plan

5.1 Project Estimates

5.1.1 Reconciled Estimates

Compare estimate of software size with estimate of effort time. Bridging the gap between the two is called reconciliation of estimates.

Time Estimates
2 hrs/day for approx. 3 months

5.2 Risk Management w.r.t NP Hard analysis

There are lot of programs that dont run in polynomial time on a regular computer but do run in polynomial time on a non-deterministic Turing machine. These programs solve problems in NP, which stands for Non-Deterministic Polynomial time. If a problem is NP-Hard this means it can reduce any problem in NP to that problem. This means if I can solve that problem I can easily solve any problem in NP.
There are three types of classes provided for this are follows:
(a) NP-Hard (b) NP-Complete Class

A decision problem is in NP class if there is known as polyno-

mial time algorithm for a non-deterministic machine to get the answer. Problems known to be in P are trivially in NP the non-deterministic machine just never troubles itself to fork another process and acts just like a deterministic one. A problem is NP-hard if solving it in polynomial time would make it possible to solve all problems in class NP in polynomial time. Some NP-hard problems are also in NP (these are called NP-complete), some are not. If you could reduce an NP problem to an NP-hard problem and then solve it in polynomial time, you could solve all NP problems. Also, there are decision problems in NP-hard but are not NP-complete, such as the infamous halting problem.

5.2.1 Risk Identification

Project Risk Management includes the processes of conducting risk management Planning, identification, analysis, response planning, and controlling risk on a project. The objectives of project risk management are to increase the likelihood and impact of positive events, and decrease the likelihood and impact of negative events in the project. Project Risk identification is the most important process in the Risk Management Planning. Risk identification determines which risks might affect the project and documents their characteristics. However, we should not spend too much time in identifying risks. After the list is made, qualitative and quantitative analysis is done to figure out which risks you spend time and/or money on. In our project the requirements of end user is fully understood which minimizes the risk. To develop the software the development team is skilled and have appropriate knowledge about the tools which we are using to develop the software. Each team member is equally involved in the development of project in each stage. The number of peoples required for developing are sufficient. The requirement gathered is constant or stable which minimizes the risk of developing an inaccurate system.

5.2.2 Risk Analysis

The risks for the Project can be analyzed within the constraints of time and quality. Software Risk analysis a very important aspect of risk management. In this phase the risk is identified and then categorized. After the categorization of risk, the level, likelihood

(percentage) and impact of the risk is analyzed. Likelihood is defined in percentage after examining what are the chances of risk to occur due to various technical conditions.

5.3 Project Schedule

5.3.1 Estimation Diagram

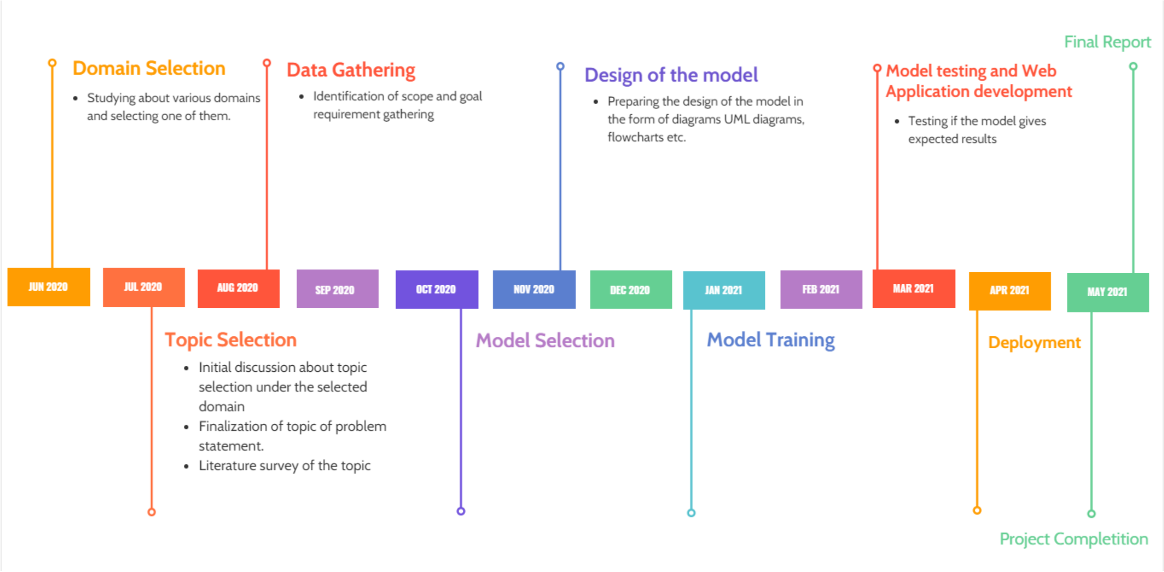

Final Report
Domain Selection
• Studying about various domains and selecting one of them.
Data Gathering
• Identification of scope and goal requirement gathering
Design of the model
• Preparing the design of the model in the form of diagrams UML diagrams, flowcharts etc.
Model testing and Web Application development
• Testing if the model gives expected results
JUN 2020
JUL 2020
AUG 2020
SEP 2020
OCT 2020
NOV 2020
DEC 2020
JAN 2021
FEB 2021
MAR 2021
APR 2021
MAY 2021
Topic Selection
• Initial discussion about topic selection under the selected domain
• Finalization of topic of problem statement.
• Literature survey of the topic
Model Selection
Model Training
Deployment
Project Completition

Figure 5.1: Estimation Diagram

5.3.2 Project task set

Major Tasks in the Project stages are:

- **Task 1: Choosing Project Area and Planning.**
- **Task 2: Selecting paper and literature survey**
- **Task 3: Project designing**
- **Task 4: Implementation**
- **Task 5: Execution and Testing**

5.4 Team Organization

Team consists of 4 members and proper planning mechanism are used and roles of each member are defined.

Distribution was as follows:

- **Ketan: Data Gathering and Pre-processing.**
- **Sudesh: Model Definition and Training**
- **Ghanshyam: Flask Integration and Literature Survey.**
- **Piyush: Frontend Designing and Documentation.**

Chapter 6

Project Implementation

6.1 Overview of Project Modules

6.1.1 Dataset Scraping

- The Core difference between existing implementations and proposed system is that existing systems are trained purely on Kaggle Datasets which are predominantly based on news by Western Media.
- There is a stark difference between the style of reporting in case of Western Media and Indian Media.
- Therefore, as the first module of our system, we are generating our own dataset by scrapping the prominent media houses and notoriously known fake news sites.

Tech Stack for the module: BeautifulSoup

6.1.2 Dataset Filtration Module

- Can also be called as Data Preprocessing Module.
- Dataset scrapped is raw data and not necessarily clean in nature, therefore our deep learning system will not be able to interpret anything from it.
- Therefore this module of the project has the objective of making the raw data into proper data which can be readily fed into the DL Model.

- **Based on our literature review, we have identified the NLP Operations required to be performed on the raw data and in this phase, we will be implementing that.**

Tech Stack for the module: NLTK

6.1.3 Neural Network Module

- **Once we have processed the dataset, we can pass that to our model.In this module, we will develop the Deep Learning System as proposed by referring the literature survey.**
- **From our survey, we know that we can get better results, by using both the news title and news content.**
- **N-Gram Vectorization and Sequence Vectors are two proposed methods which will be tested and the better one will be selected.**

Tech Stack for the module: TensorFlow

6.1.4 Frontend Module

- **Users cannot directly interact with the DL Model.**
- **Though theoretically it is possible but not user-friendly to do so.**
- **Therefore, we need an User Interface so that users can use the developed model easily , without any hassle.**
- **The UI would accept the news title and news content from user and display the predicted output to the user.**

Tech Stack for the module: React.js

6.1.5 Integration Module

- **Module 2 and 3 can be considered as backend and Module 4 as frontend.**
- **However, React Frontend cannot directly interact with the Tensorflow backend.**

- **Therefore, we need a separate module responsible for all communications between the frontend and backend.**
- **This module will pass the obtained input to the deep learning model and then send the processed output back to the frontend.**
- **Since the backend is predominantly Python, Flask is a viable option for this module.**

Tech Stack for the module: Flask

6.2 Tools and Technologies Used

- **Google Colab: It is an online Jupyter Notebook like environment for executing Python code in the browser. Its biggest advantage is that it requires no setup and runs completely in the cloud. Colab uses GPUs and TPUs in the backend for running the Python code hence it is completely suitable for data intensive ML, Deep Learning and Data Science applications. And the best part of it all is that Colab is totally free.**
- **NLTK: Natural Language Toolkit is a platform used for building Python programs that work with human language data for applying in statistical natural language processing (NLP). It contains text processing libraries for tokenization, parsing, classification, stemming, tagging and semantic reasoning. It also includes graphical demonstrations and a book which explains the principles behind the underlying language processing tasks that NLTK supports.**
- **Keras: It contains numerous implementations of commonly used neural-network building blocks such as layers, objectives, activation functions, optimizers, and a host of tools to make working with image and text data easier to simplify the coding necessary for writing deep neural network code. In addition to standard neural networks, Keras has support for convolutional and recurrent neural networks. It supports other common utility layers like dropout, batch normalization, and pooling. Keras allows users to productize deep models on smartphones,**

on the web, or on the JVM. It also allows use of distributed training of deep-learning models on clusters of GPU and TPU.

- Flask: Flask is a micro web framework written in Python. It is classified as a micro framework because it does not require particular tools or libraries. It has no database abstraction layer, form validation, or any other components where pre-existing third-party libraries provide common functions. However, it supports extensions that can add application features as if they were implemented in Flask itself. Extensions exist for object-relational mappers, form validation, upload handling, various open authentication technologies and several common framework related tools.
- React: React (also known as ReactJS) is an open-source front-end JavaScript library for building user interfaces or UI components. It is maintained by Facebook and a community of individual developers and companies. React can be used as a base in the development of single-page or mobile applications. However, React is only concerned with state management and rendering that state to the DOM, so creating React applications usually requires the use of additional libraries for routing, as well as certain client-side functionality.

6.3 Algorithm Details

6.3.1 N-Gram Vectorization

In the N-Gram Vectorization method, a document term matrix is generated for the document and each cell represents the count. The difference in the N-grams method is that the count represents the combination of adjacent words of length n in the title. Count vectorization is a special case of N-Gram where n=1.
To understand N-Gram Vectorization better consider example, “I love this article” has four words and n=4. So if n=2, i.e bigram, then the columns would be — [“I love”, “love this”, ‘this article”] and if n=3, i.e trigram, then the columns would be — [“I love this”, ”love this article”] and if n=4,i.e four-gram, then the column would be -[“I love this article”]. The n value is chosen based on optimal performance as per the intended task.

6.3.2 TF-IDF

In the TF-IDF method, a document term matrix is generated and each column represents a single unique word. The difference in the TF-IDF method is that each cell doesn't indicate the term frequency, but the cell value represents a weighting that highlights the importance of that particular word to the document. For calculating this importance, we use the following formula :

$$W_{x,y} = tf_{x,y^*}\ log\left(\frac{N}{df_x}\right)$$

$W_{x,y}$= Word x within document y

$tf_{x,y}$= frequency of x in y

df_x= number of documents containing x

N=total number of documents

Figure 6.1: TF-IDF Formula

6.3.3 Deep Learning Model

With all the pre-processed new titles and content in vectors form, Keras neural network models with some dense layers are built and trained using Tensorflow framework to perform classification task of detecting the fake news.

In the Keras neural network model, the layers 1, 3 and 5 are using RELU as the activation function with 32 nodes, 16 nodes and 8 nodes respectively.

In between these 3 layers, 2 dropout layers of 40% dropout rate are also added to the network. The purpose of adding dropout layers is used to avoid from overfitting by dropping some node and therefore, generalize better.

Then, the last layer of the Keras neural network model, also known as output layer, is added with sigmoid activation function so to ensure the output of the network is in binary format because there are only 2 possible outcomes from this project such that "0" refers to real news and "1" refers to fake news.

Chapter 7

Software Testing

7.1 Types of Tests Performed

- **Unit Testing**
- **Integration Testing**
- **Stress Testing**
- **GUI Testing**
- **Performance Testing**
- **Regression Testing**
- **Compatibility Testing**

7.1.1 Unit Testing

Purpose of this test is to validate that each unit of software performs as specified in SRS.

7.1.2 Integration Testing

The individual functional modules in the unit tests are integrated and tested as one.The NLP and Deep Learning Model, Flask Server, React Frontend and Processing modules are integrated and tested as a single complete web application to verify that it works as expected.

7.1.3 Stress Testing

The System is tested under workload.The speed,responsiveness and stability of a computer, network, software program or device is determined.

7.1.4 GUI Testing

The React Frontend is tested to see if it displays all the components properly or not. All components were found to be working properly and GUI Test was declared successful.

7.1.5 Model Performance Testing

Testing model performance is about testing the models with the test data/ unknown data and comparing the model performance in terms of parameters such as accuracy/recall/F1-Score etc., to that of predetermined accuracy with the model already built and moved into production.

7.1.6 Regression Testing

All modules and the final system were regression tested by introducing small changes in the code and then checking it's effect. No effects were found and therefore, the regression testing was declared as successful.

7.1.7 Compatibility Testing

Compatibility testing is a non-functional test to ensure our web application's compatibility within different environments such as various Operating Systems (Windows, Linux) and their various versions.Project is tested on various OS and browsers.

7.2 Test Cases & Test Results

ID	Type	Test Data	Expected Result	Actual Result	Status
TC1	Satire	Link1	Fake	Fake	PASS
TC2	Real News	Link2	True	True	PASS
TC3	Fake News	Link3	Fake	Fake	PASS
TC4	Media	Link4	Invalid	Invalid	PASS
TC5	Invalid URL	Link5	Invalid	Invalid	PASS

Link1: https://thefauxy.com/man-dies-after-taking-pakvac-imran-khan-claims-vaccine-working-perfectly-fine-since-corona-virus-also-died-inside-mans-body/

Link2: https://indianexpress.com/article/india/coronavirus-india-live-updates-lockdown-news-india-covid-vaccine-7353818/

Link3: https://www.vice.com/en/article/m7apnn/your-cock-is-mine-now-hacker-locks-internet-connected-chastity-cage-demands-ransom

Link4: https://www.youtube.com/c/DannyMullenOfficial

Link5: ww.theonioncom/fbi-says-chauvin-matches-profile-of-blue-uniformed-kill-1846732261

Chapter 8

Results

8.1 Outcomes

8.1.1 Confusion Matrix

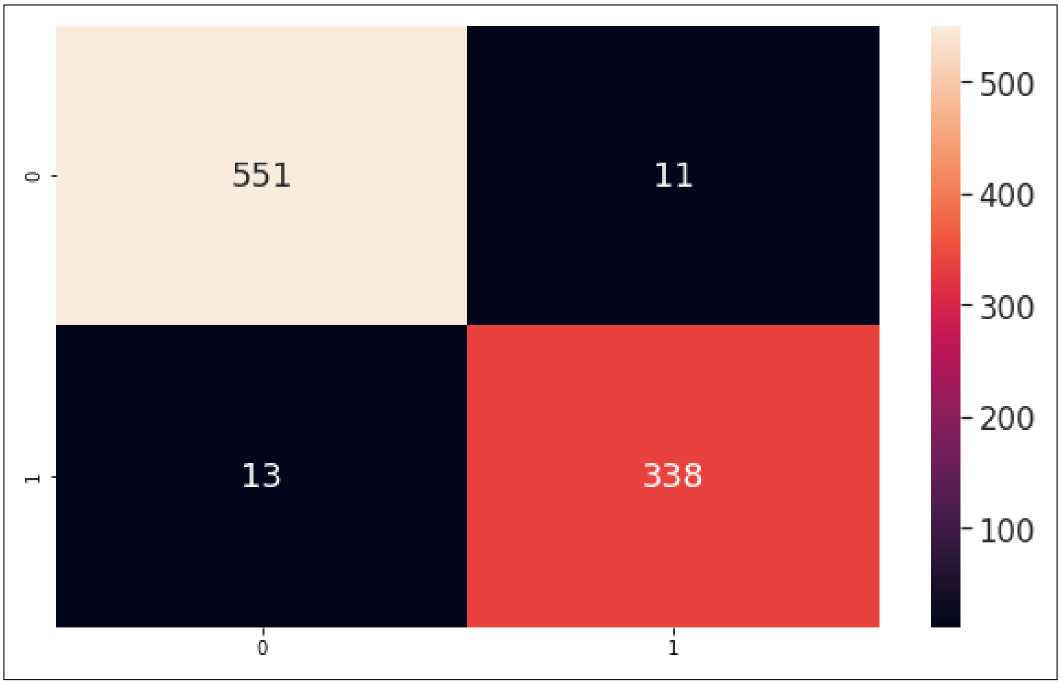
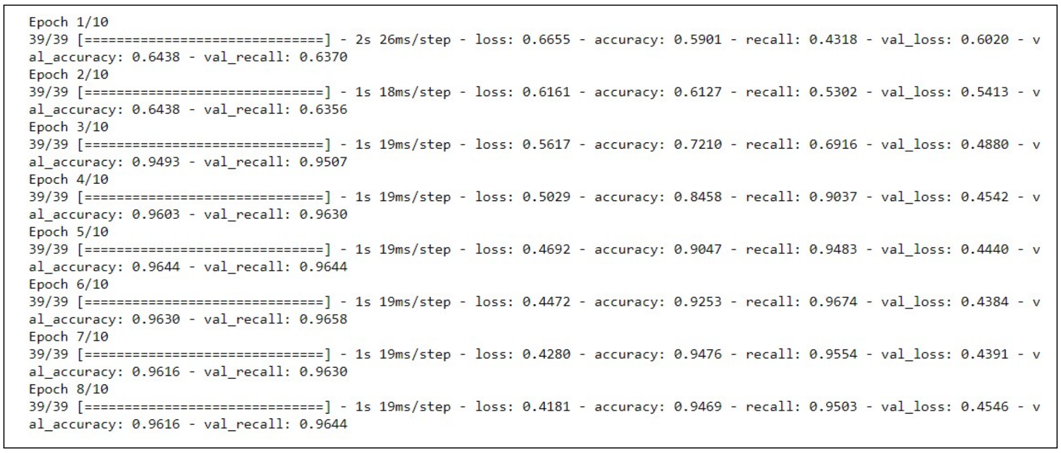


Figure 8.1: Confusion Matrix

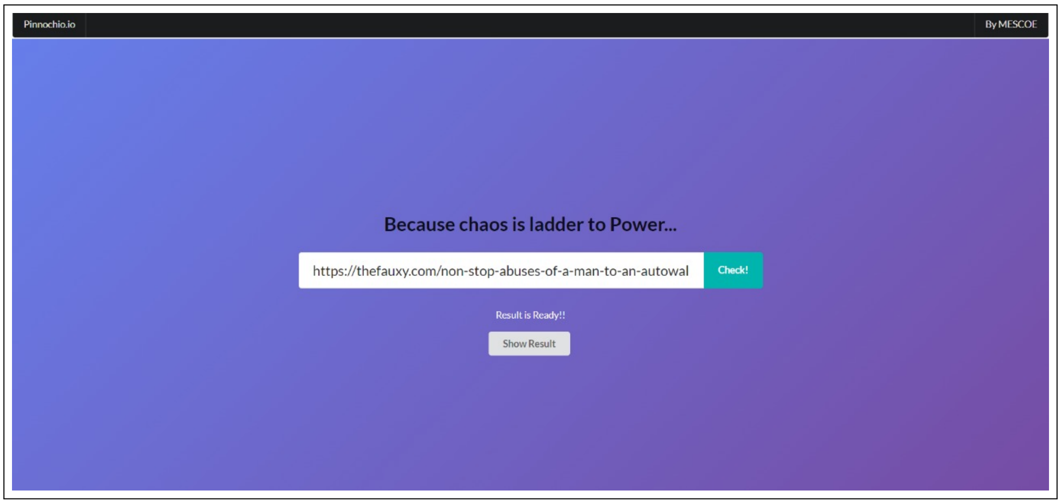
8.1.2 Training Epochs



```
Epoch 1/10
39/39 [==============================] - 2s 26ms/step - loss: 0.6655 - accuracy: 0.5901 - recall: 0.4318 - val_loss: 0.6020 - v
al_accuracy: 0.6438 - val_recall: 0.6370
Epoch 2/10
39/39 [==============================] - 1s 18ms/step - loss: 0.6161 - accuracy: 0.6127 - recall: 0.5302 - val_loss: 0.5413 - v
al_accuracy: 0.6438 - val_recall: 0.6356
Epoch 3/10
39/39 [==============================] - 1s 19ms/step - loss: 0.5617 - accuracy: 0.7210 - recall: 0.6916 - val_loss: 0.4880 - v
al_accuracy: 0.9493 - val_recall: 0.9507
Epoch 4/10
39/39 [==============================] - 1s 19ms/step - loss: 0.5029 - accuracy: 0.8458 - recall: 0.9037 - val_loss: 0.4542 - v
al_accuracy: 0.9603 - val_recall: 0.9630
Epoch 5/10
39/39 [==============================] - 1s 19ms/step - loss: 0.4692 - accuracy: 0.9047 - recall: 0.9483 - val_loss: 0.4440 - v
al_accuracy: 0.9644 - val_recall: 0.9644
Epoch 6/10
39/39 [==============================] - 1s 19ms/step - loss: 0.4472 - accuracy: 0.9253 - recall: 0.9674 - val_loss: 0.4384 - v
al_accuracy: 0.9630 - val_recall: 0.9658
Epoch 7/10
39/39 [==============================] - 1s 19ms/step - loss: 0.4280 - accuracy: 0.9476 - recall: 0.9554 - val_loss: 0.4391 - v
al_accuracy: 0.9616 - val_recall: 0.9630
Epoch 8/10
39/39 [==============================] - 1s 19ms/step - loss: 0.4181 - accuracy: 0.9469 - recall: 0.9503 - val_loss: 0.4546 - v
al_accuracy: 0.9616 - val_recall: 0.9644
```

Figure 8.2: Training Model

8.2 Screenshots

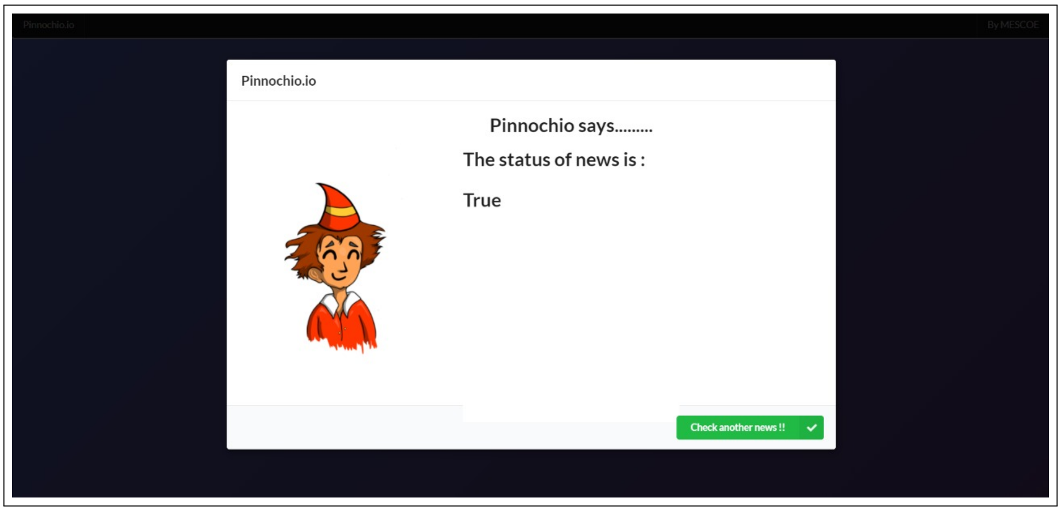


Because chaos is ladder to Power...

https://thefauxy.com/non-stop-abuses-of-a-man-to-an-autowal

Result is Ready!!

Figure 8.3: Homepage of Web App



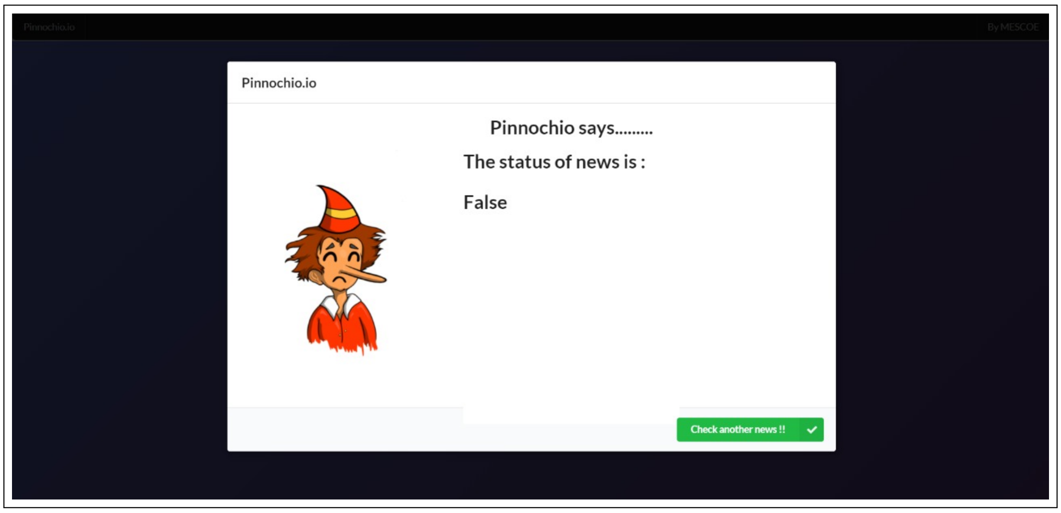
Pinnochio.io

Pinnochio says.........

The status of news is :

True

Figure 8.4: True Prediction



Pinnochio.io

Pinnochio says.........

The status of news is :

False

Figure 8.5: False Prediction

Chapter 9

Other Specifications

9.1 Advantages

1. Social networking sites contain false information and people are often fallen prey to it.

2. Detecting those fake contents on these sites will safeguard the authenticity of the information available online.

3. Detecting Fake news that is forwarded can prevent the spread of rumors.

4. Better Output than Vanilla Machine Learning Models.

5. Easier to detect red flags than manual methods.

6. Emphasis on avoiding biased treatment towards any news sources.

9.2 Limitations

1. This model uses the N-gram approach which fails to model more complicated contextual dependencies of the text.

2. Accuracy can be further improved.

3. As the rate of producing news is increasing day by day so it is challenging to effectively detect fake news.

4. Detection of Sarcasm is not 100% accurate.

Chapter 10

Conclusions

10.1 Conclusions

Many people consume news from social network instead of mainstream news media. However, social media has also been used to spread misinformation, which has bad impact on individuals and communities.

In this paper, an innovative model for falsified news detection using deep learning algorithm & N-Gram Vectorization has been proposed. This model takes news events as an input and based on the features observed by deep learning model, it predicts whether news is false or true and displays it to the user via a web app interface.

10.2 Future Work

**Since the entire project has been developed using design patterns which emphasize on modularity of the individual components, we have managed to develop the backend in such a form that it can be used by any website/application without any issues. The Backend accepts URL of articles in JSON format and returns the result in JSON , making the entire backend as lucid as a REST API.
Therefore, we believe that there is a great scope of application of our model in a lot of applications.
The Model can be further optimized once GPT-3 is publicly available for use. Graph based Neural Networks can also be looked towards as one of the options.**

10.3 Applications

1. **For accreditation of authenticity of media houses.**
2. **Users can verify news correctness.**
3. **In Forensic Investigation of Cases.**
4. **In Systems designed to stop spreading of Misinformation**

Chapter 11

References

[1] S. H. Kong, L. M. Tan, K. H. Gan and N. H. Samsudin, "Fake News Detection using Deep Learning," 2020 IEEE 10th Symposium on Computer Applications Industrial Electronics (ISCAIE), Malaysia, 2020, pp. 102-107, doi: 10.1109/ISCAIE47305.2020.9108841. .

[2] L. Wu, Y. Rao, H. Yu, Y. Wang and N. Ambreen, "A Multisemantics Classification Method Based on Deep Learning for Incredible Messages on Social Media," in Chinese Journal of Electronics, vol. 28, no. 4, pp. 754-763, 7 2019, doi: 10.1049/cje.2019.05.002.

[3] M. K. Elhadad, K. F. Li and F. Gebali, "Detecting Misleading Information on COVID-19," in IEEE Access, vol. 8, pp. 165201-165215, 2020, doi: 10.1109/ACCESS.2020.3022867.

[4] M. Umer, Z. Imtiaz, S. Ullah, A. Mehmood, G. S. Choi and B. -W. On, "Fake News Stance Detection Using Deep Learning Architecture (CNN-LSTM)," in IEEE Access, vol. 8, pp. 156695-156706, 2020, doi: 10.1109/ACCESS.2020.3019735.

[5] F. T. Asr and M. Taboada, "Big Data and quality data for fake news and misinformation detection", Big Data Soc., vol. 6, no. 1, pp. 1-14, May 2019.

[6] Jiawei, Zhang & Dong, Bowen & Yu, Philip. (2020). FakeDetector: Effective Fake News Detection with Deep Diffusive Neural Network. 1826-1829. 10.1109/ICDE48307.2020.00180.

[7] Kumar, S, Asthana, R, Upadhyay, S, Upreti, N, Akbar, M. Fake news detection using deep learning models: A novel approach. Trans Emerging Tel Tech. 2020; 31:e3767.

[8] 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.

[9] R. K.Kaliyar, A.Goswami, P.Narang, S. Sinha, FNDNet – A deep convolutional neural network for fake news detection, Cognitive Systems Research,Vol. 61, 2020,Pg 32-44, ISSN 1389-0417.

[10] S. Girgis, E. Amer and M. Gadallah, "Deep Learning Algorithms for Detecting Fake News in Online Text," 2018 13th Intl Conference on Computer Engineering and Systems (ICCES), 2018, pp. 93-97, doi: 10.1109/ICCES.2018.8639198.

[11] Agarwal, A., Mittal, M., Pathak, A. et al. Fake News Detection Using a Blend of Neural Networks: An Application of Deep Learning. SN COMPUT. SCI. 1, 143 (2020).

Appendix A

Problem Statement Feasibility Assessment

A.1 Theory

A.1.1 NP-Hard problem

- NP-Complete and NP-Hard are: A decision problem is in P if there is a known polynomial-time algorithm to get that answer. The collection of all problems that can be solved in polynomial time is called P. That is, a decision question is in P if there exist an exponent k and an algorithm for the question that runs in time O(nk) where n is the length of the input.

- A decision problem is in NP if there is a known polynomial-time algorithm for a Non-deterministic machine to get the answer. The estimation cannot be solved in fixed time or we cannot define their execution complexity with a mathematical algorithm, are called as Non-Deterministic polynomial problems. Therefore, the problem becomes a decision problem, so it is NP.

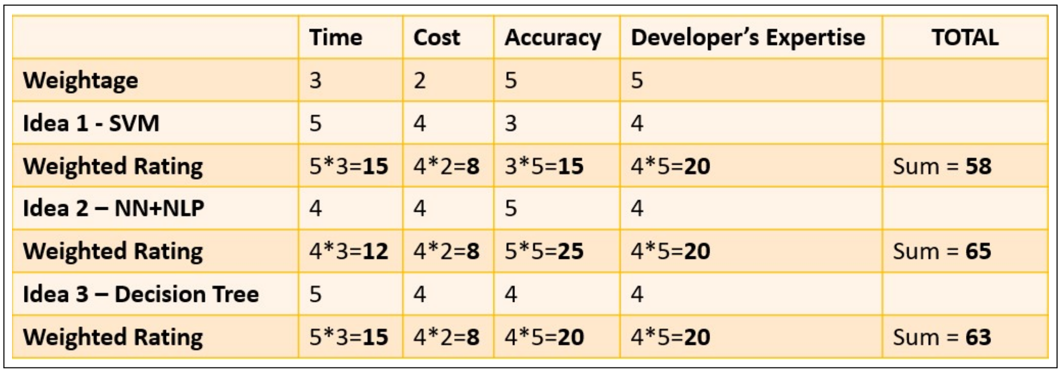
A.1.2 NP-Complete

- The collection of all problems that can be solved in polynomial time using non- deterministic is called NP. That is, a decision question is in NP if there exists an exponent k and a non-deterministic algorithm for the question that for all hints runs in time O (nk) where n is the length of the input.

- **Detection of bumps and finding ghat complexities with evaluation of road can be done in polynomial time but it requires infinite time for DB interaction.**

- **All project algorithms can be determined in polynomial time but requires indefinite time for db interaction. Hence all Database handling projects are NP-COMPLETE.**

A.2 IDEA Matrix



	Time	Cost	Accuracy	Developer's Expertise	TOTAL
Weightage	3	2	5	5	
Idea 1 - SVM	5	4	3	4	
Weighted Rating	5*3=**15**	4*2=**8**	3*5=**15**	4*5=**20**	Sum = **58**
Idea 2 – NN+NLP	4	4	5	4	
Weighted Rating	4*3=**12**	4*2=**8**	5*5=**25**	4*5=**20**	Sum = **65**
Idea 3 – Decision Tree	5	4	4	4	
Weighted Rating	5*3=**15**	4*2=**8**	4*5=**20**	4*5=**20**	Sum = **63**

Figure A.1: Matrix

Thus after the Brainstorming session and generation of IDEA Matrix, we concluded that going forward with the idea of implementing Neural Network would be in best interest's of the project.

Appendix B

Details of Paper Publication

B.1 Paper 1: Review Paper

1. **Paper Title: Review of Falsified News Detection using Deep Learning**
2. **Name of the Conference/Journal where paper submitted : IJRASET**
3. **Paper accepted/rejected : Accepted with Revisions**
4. **Review comments by reviewer 1 :**
   - **Abstract is comprehensive and important and essential information of the article is included,**
   - **Proposed work is good.**
5. **Review comments by reviewer 2 :**
   - **Paper formatting and presentation is up to the mark.**
   - **Add justification of why rectification is need for layer 1,3,5.**
   - **Explain about dropout rate of the layers.**
6. **Corrective actions (if any) :**
   - **All Asked Revisions were addressed.**
   - **The revised manuscript was submitted and the paper was accepted.**

B.2 Paper 2: Research Paper

1. **Paper Title: Falsified News Detection using Deep Learning**
2. **Name of the Conference/Journal where paper submitted : IEEE Asiancon 2021**
3. **Paper accepted/rejected : Submitted**
4. **Review comments by reviewer 1 : NA**
5. **Review comments by reviewer 2 : NA**
6. **Corrective actions if any : NA**

Appendix C

Plagiarism Report

Paper 1

ORIGINALITY REPORT

11%	4%	9%	1%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Sheng How Kong, Li Mei Tan, Keng Hoon Gan, Nur Hana Samsudin. "Fake News Detection using Deep Learning", 2020 IEEE 10th Symposium on Computer Applications & Industrial Electronics (ISCAIE), 2020 Publication	6%
2	www.ijircce.com Internet Source	2%
3	Anjali Jain, Avinash Shakya, Harsh Khatter, Amit Kumar Gupta. "A smart System for Fake News Detection Using Machine Learning", 2019 International Conference on Issues and Challenges in Intelligent Computing Techniques (ICICT), 2019 Publication	1%
4	Mohamed K. Elhadad, Kin Fun Li, Fayez Gebali. "Detecting Misleading Information on COVID-19", IEEE Access, 2020 Publication	1%
5	www.springerprofessional.de Internet Source	

1 %

6	ieeexplore.ieee.org Internet Source	<1 %
7	www.coursehero.com Internet Source	<1 %
8	"ECAI 2020", IOS Press, 2020 Publication	<1 %
9	link.springer.com Internet Source	<1 %

Exclude quotes	Off	Exclude matches	Off
Exclude bibliography	On		