PURBANCHAL UNIVERSITY

DEPARTMENT OF COMPUTER ENGINEERING
KHWOPA ENGINEERING COLLEGE
LIBALI-08, BHAKTAPUR

A MID TERM REPORT ON
Fake News Detection

Project work submitted in partial fulfillment of the requirements for the degree of Bachelor of Engineering in Computer Engineering(Eight Semester)

Submitted by

Raj Duwal	(760329)
Ram Simachhwa	(760331)
Rohan Shingkhwal	(760334)
Shivansh Duwadi Sharma	(760339)
Sujeet Chakradhar	(760344)

Khwopa Engineering College
Libali-08, Bhaktapur
July 05, 2024

Acknowledgement

Any attempt at any level cannot be satisfactorily completed without the support and guidance of learned people. We would like to express our immense gratitude to the Department of Computer Engineering, Khwopa Engineering College for providing all the necessary resource and guideline that encouraged us to come up with this report. Our special thanks to Er. Bikash Chawal (HoD) for providing us a golden opportunity to work on our project 'Fake News Detection' and rendering us all the possible help during the development, implementation and presentation work.

Special thanks to all our colleagues of our section who have rendered their whole hearted support at all times for successful completion of this project directly as well as indirectly by reviewing and suggesting the improvements.

Raj Duwal	(760329)
Ram Simachhwa	(760331)
Rohan Shingkhwalr	(760334)
Shivansh Duwadi Sharma	(760339)
Sujeet Chakradhar	(760344)

Abstract

In recent years, due to the booming development of online social networks, fake news for various commercial and political purposes has been appearing in large numbers and widespread in the online world. With deceptive words, online social network users can get infected by these online fake news easily, which has brought about tremendous effects on the offline society already. An important goal in improving the trustworthiness of information in online social networks is to identify the fake news timely.

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. Information preciseness on Internet, especially on social media, is an increasingly important concern, but web-scale data hampers, ability to identify, evaluate and correct such data, or so called "fake news," present in these platforms. In this paper, we propose a method for "fake news" detection and ways to apply it on Facebook, one of the most popular online social media platforms. This method uses Naive Bayes classification model to predict whether a post on Facebook will be labeled as real or fake. The results may be improved by applying several techniques that are discussed in the paper. Received results suggest, that fake news detection problem can be addressed with machine learning methods

Contents

List of Tables

List of Figures

List of Abbreviation

Abbreviations	Meaning
CNN	Convolutional Neural Network
LSTM	Long Short Term Memory
SVM	Support Vector Machine
AI	Artificial Intelligence
UI	User Interface

Chapter 1

Introduction

1.1 Background Introduction

These days fake news is creating different issues from sarcastic articles to a fabricated news and plan government propaganda in some outlets. Fake news and lack of trust in the media are growing problems with huge ramifications in our society. Obviously, a purposely misleading story is fake news but lately blathering social media's discourse is changing its definition. Some of them now use the term to dismiss the facts counter to their preferred viewpoints.

The importance of disinformation within American political discourse was the subject of weighty attention, particularly following the American president election. The term 'fake news' became common parlance for the issue, particularly to describe factually incorrect and misleading articles published mostly for the purpose of making money through page views. In this paper, it is sought to produce a model that can accurately predict the likelihood that a given article is fake news. Facebook has been at the epicenter of much critique following media attention. They have already implemented a feature to flag fake news on the site when a user sees it; they have also said publicly they are working on to distinguish these articles in an automated way. Certainly, it is not an easy task. A given algorithm must be politically unbiased since fake news exists on both ends of the spectrum and also give equal balance to legitimate news sources on either end of the spectrum. In addition, the question of legitimacy is a difficult one. However, in order to solve this problem, it is necessary to have an understanding on what Fake News

1.2 Motivation

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

1.3 Problem Definition

The advent of digital technologies and the rise of social media platforms have revolutionized the way information is disseminated, creating unprecedented opportunities for communication and knowledge sharing. However, this digital landscape has also facilitated the rapid spread of misinformation and fake news, posing significant challenges to the integrity of public discourse, democratic processes, and societal trust.

Fake news, characterized by intentionally deceptive or misleading content presented as factual information, has become a pervasive phenomenon in online spaces. From fabricated stories shared on social media platforms to manipulated images and videos circulated through various digital channels, the proliferation of fake news undermines the credibility of legitimate sources, erodes public trust, and distorts public perceptions of reality.

Existing methods for detecting fake news often rely on manual fact-checking processes or simplistic algorithms that are limited in their effectiveness, scalability, and ability to adapt to evolving forms of deception. Manual fact-checking is time-consuming, resource-intensive, and unable to keep pace with the sheer volume and velocity of misinformation circulating online. On the other hand, simplistic algorithms may struggle to differentiate between genuine and fabricated content, leading to high rates of false positives or negatives.

To address these challenges, there is an urgent need for advanced algorithms and systems capable of accurately detecting and combating fake news in real-time. Leveraging techniques from natural language processing, machine learning, data mining, and computational linguistics, these solutions should be designed to analyze textual, visual, and contextual cues to distinguish between legitimate and deceptive content effectively.

Moreover, these algorithms should be adaptable and robust enough to handle the evolving tactics employed by purveyors of fake news, including the use of sophisticated language, manipulation techniques, and adversarial attacks. By integrating these advanced detection mechanisms into digital platforms, social media networks, and online news outlets, we can empower users, fact-checkers, and content moderators with reliable tools to identify and mitigate the spread of misinformation.

Ultimately, the development and deployment of effective fake news detection systems are essential for preserving the integrity of online information ecosystems, safeguarding democratic values, and fostering informed public discourse. By combating the spread of fake news, we can uphold the principles of truth, transparency, and trustworthiness in the digital age.

1.4 Objectives

The main objective of this project is:

- To examine the problems and significance's related to the spread of fake news by applying various machine learning algorithms to different datasets to distinguish between real and fake news, aiming to mitigate the societal impact of misinformation through AI-driven pattern recognition.

1.5 Scope and Applications

The major scope of the project is to assist visually impaired people and the system is able to:

- Social Media Platforms: Integrating detection algorithms to flag or remove deceptive content, enhancing user trust and safety.
- Online News Outlets: Verifying news authenticity, fact-checking, and maintaining journalistic integrity.
- Fact-Checking Organizations: Systematically assessing news accuracy, debunking misinformation, and providing reliable information.
- Government Agencies: Monitoring disinformation campaigns, protecting national security, and promoting public awareness.
- Educational Institutions: Teaching critical thinking, media literacy, and digital citizenship to evaluate information critically.
- Research and Development: Advancing algorithms, datasets, and evaluation metrics to combat misinformation effectively.

Chapter 2

Literature Review

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

2.1 MEDIA RICH FAKE NEWS DETECTION: A SURVEY

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

2.2 WEAKLY SUPERVISED LEARNING FOR FAKE NEWS DETECTION ON TWITTER

The problem of automatic detection of fake news in social media, e.g., on Twitter, has recently drawn some attention. Although, from a technical perspective, it can be regarded as a straight-forward, binary classification problem, the major challenge is the collection of large enough training corpora, since manual annotation of tweets as fake or non-fake news is an expensive and tedious endeavor. In this paper, we discuss a weakly supervised approach, which automatically collects a large-scale,but very noisy training dataset comprising hundreds of thousands of tweets. During collection, we automatically label tweets by their source, i.e., trustworthy or untrustworthy source, and train a classifier on this dataset. We then use that classifier for a different classification target, i.e., the classification of fake and non- fake tweets. Although the labels are not accurate according to the new classification target (not all tweets by an untrustworthy source need to be fake news, and vice versa), we show that despite this unclean inaccurate dataset, it is possible to detect fake news with an F1 score of up to 0.9

2.3 FAKE NEWS DETECTION IN SOCIAL MEDIA

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

2.4 AUTOMATIC ONLINE FAKE NEWS DETECTION COMBINING CONTENT AND SOCIAL SIGNALS

The proliferation and rapid diffusion of fake news on the Internet highlight the need of automatic hoax detection systems. In the context of social networks, machine learning (ML) methods can be used for this purpose. Fake news detection strategies are traditionally either based on content analysis (i.e. analyzing the content of the news) or - more recently - on social context models, such as mapping the news diffusion pattern. In this paper, we first propose a novel ML fake news detection method which, by combining news content and social context features, outperforms existing methods in the literature, increasing their already high accuracy by up to 4.8%. Second, we implement our method within a Facebook Messenger chatbot and validate it with a real-world application, obtaining a fake news detection accuracy of 81.7%. In recent years, the reliability of information on the Internet has emerged as a crucial issue of modern society. Social network sites (SNSs) have revolutionized the way in which information is spread by allowing users to freely share content. As a consequence, SNSs are also increasingly used as vectors for the diffusion of misinformation and hoaxes. The amount of disseminated information and the rapidity of its diffusion make it practically impossible to assess reliability in a timely manner, highlighting the need for automatic hoax detection systems.As a contribution towards this objective, we show that Facebook posts can be classified with high accuracy as hoaxes or non-hoaxes on the basis of the users who "liked" them. We present two classification techniques, one based on logistic regression, the other on a novel adaptation of boolean crowdsourcing algorithms. On a dataset consisting of 15,500 Facebook posts and 909,236 users, we obtain classification accuracies exceeding 99% even when the training set contains less than 1% of the posts. We further 12 show that our techniques are robust: they work even when we restrict our attention to the users who like both hoax and non-hoax posts. These results suggest that mapping the diffusion pattern of information can be a useful component of automatic hoax detection systems.

2.5 THE SPREAD OF FAKE NEWS BY SOCIAL BOTS

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

2.6 MISLEADING ONLINE CONTENT

Tabloid journalism is often criticized for its propensity for exaggeration, sensationalization, scare-mongering, and otherwise producing misleading and low quality news. As the news has moved online, a new form of tabloidization has emerged:click baiting. Clickbait refers to “content whose main purpose is to attract attention and encourage visitors to click on a link to a particular web page clickbait, n.d. and has been implicated in the rapid spread of rumor and misinformation online. This paper examines potential methods for the automatic detection of clickbait as a form of deception. Methods for recognizing both textual and non-textual clickbaiting cues are surveyed, leading to the suggestion that a hybrid approach may yield best results.

Big Data Analytics and Deep Learning are two high focus of data science. Big Data has become important as many organizations both public and private have been collecting massive amounts of domain-specific information, which can contain useful information about problems such as national intelligence, cyber security, fraud detection, marketing, and medical informatics. Companies such as Google and Microsoft are analyzing large volumes of data for business analysis and decisions, impacting existing and future technology. Deep Learning algorithms extract high-level, complex abstractions as data representations through a hierarchical learning process. Complex abstractions are learnt at a given level based on relatively simpler abstractions formulated in the preceding level in the hierarchy. A key benefit of Deep Learning is the analysis and learning of massive amounts of unsupervised data, making it a valuable tool for Big Data Analytics where raw data is largely unlabeled and un-categorized. In the present study, we explore how Deep Learning can be utilized for addressing some important problems in Big Data Analytics, including extracting complex patterns from massive volumes of data, semantic indexing, data tagging, fast information retrieval, and simplifying discriminating tasks. We also investigate some aspects of Deep Learning research that need further exploration to incorporate specific challenges introduced by Big Data Analytics, including streaming data, high-dimensional data, scalability of models, and distributed computing. We conclude by presenting insights into relevant future works by posing some questions, including defining data sampling criteria, domain adaptation modeling, defining criteria for obtaining useful data abstractions, improving semantic indexing, semi - supervised learning, and active learning.

Chapter 3
PROJECT MANAGEMENT

3.1 Team Members

Our team consists of following members:

1. Raj Duwal (760329)
2. Ram Simachhwa (760331)
3. Rohan Shingkhwal (760334)
4. Shivansh Sharma Duwadi (760339)
5. Sujeet Chakradhar (760344)

3.2 Project Scheduling

Our project schedule is shown using GANTT Chart as time frame consisting of 8 weeks as shown below:

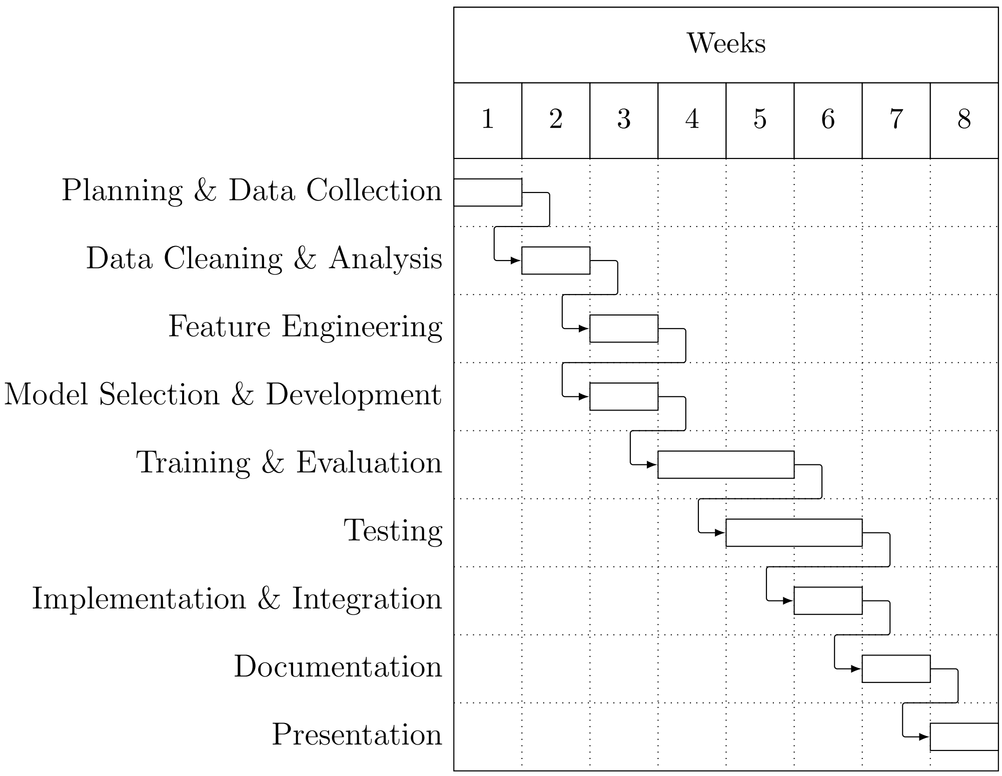

Figure 3.1: Gantt chart showing project timeline

Chapter 4
Requirement Analysis

4.1 Software Requirement

Software requirement for our prepared system includes:

a. Python

b. Visual Studio Code

c. Google Colaboratory

d. Github

e. Kaggle

f. Overleaf

g. Django

h. Slack

4.2 Hardware Requirement

Our prepared system of Fake News Detection requires following hardware requirements:

a. System - Pentium IV

b. Speed - 2.4GHZ

c. Hard disk - 40GB

d. Monitor - 15VGA color

e. RAM - 512MB

4.3 Functional Requirement

The functional requirement for the prepared systems are:

a. The system must be able to distinguish real and fake news.

b. It must not disturb and produce unwanted commands instructions.

4.4 Non-Functional Requirement

These are essential for the better performance of the system. The points below focus on the non-functional requirement of the system prepared.

4.4.1 Reliability

Different test metrices is conducted to test the accuracy of the system.

4.4.2 Maintainability

The System is breakdown into multiple sub module so that we could know where the problem is and maintain the sub module.

4.4.3 Performance

The object detection require only single process. So, it will be able to provide better performance.

4.4.4 Accessibility

Since the system is deployed in webserver. The system is easily accessible.

Chapter 5

Feasibility Study

On the basis of outcome of initial investigation, feasibility takes place. The main goal of feasibility study is not to resolve the problem but to accomplish the scope. In the process of feasibility study, the cost and the benefits are estimated with greater accuracy.

The technical feasibility of our project 'Fake News Detection' as it will be coded in python and later will be integrated as a part of website. And as such, any device that has a browser is able to access the website and explore its functionalities. Various python frameworks will beused and the website will be coded with a simple design.

5.1 Technical Feasibility

Use of open source tools and scalable tools have helped in building this application more robust and flexible hence benefiting in technical feasibility.

5.2 Economic Feasibility

The development tools were all free so it was economically feasible to develop the website. The current website can be deployed in very less amount of money. The investment needed to deploy this website is very less.

5.3 Legal Feasibility

Everything is open sourced and licensed to develop the application. It is perfectly deploy able in current market legality.

5.4 Space Feasibility

Depending on the complexity of the project, it will appropriately size. This can be improved.

5.5 Time Feasibility

The overall time to build this project will be the length of one semester. This perfectly matches the timing criteria of the project.

Chapter 6

Methodology

6.1 Software Development Approach

Prototype model is a software development model where instead of freezing the requirements before design or coding can proceed, a throwaway prototype is built to understand the requirements. The prototype are usually not complete systems and many of the details are not built in the prototype. The goal is to provide a system with overall functionality. In this model, we create the prototype of the actual system, update the requirements and again rebuild the system until the final requirements are met.

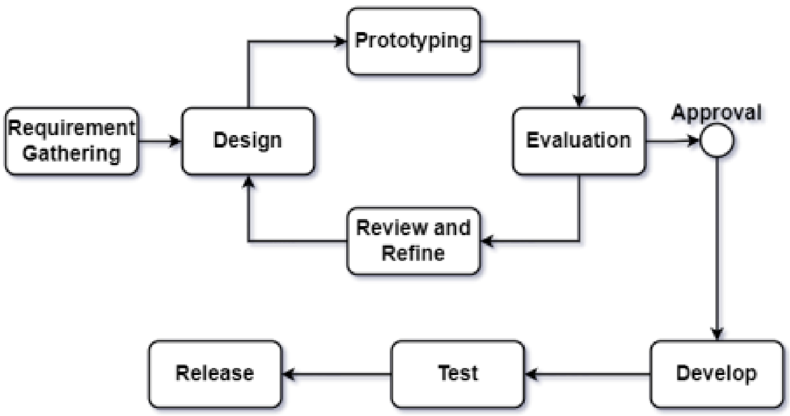

Figure 6.1: Prototype Model for Software Development

6.1.1 Requirements gathering and analysis

A prototyping model starts with requirement analysis. In this phase, the requirements of the system are defined in detail. During the process, the users of the system are interviewed to know what is their expectations from the system. In this phase, business analysts and other individuals responsible for collecting the requirements and discussing the need for the product, meet the stakeholders or clients.

6.1.2 Quick Design

The second phase is a preliminary design or a quick design. In this stage, a simple design of the system is created. However, it is not a complete design. It gives a brief idea of the system to the user. The quick design helps in developing the prototype.

6.1.3 Build a Prototype

In this phase, an actual prototype is designed based on the information gathered from a quick design. It is a small working model of the required system.

6.1.4 Initial user evaluation

In this stage, the proposed system is presented to the client for an initial evaluation. In this phase, the initial prototype is deployed and is accessible to clients for its use. Clients review or evaluate the prototype and they provide their feedback to the requirements gathering and development teams. It helps to find out the strength and weaknesses of the working model. Comment and suggestions are collected from the customer and provided to the developer.

6.1.5 Refining Prototype

If the user is not happy with the current prototype, you need to refine the prototype according to the user's feedback and suggestions. This phase will not over until all the requirements specified by the user are met. Once the user is satisfied with the developed prototype, a final system is developed based on the approved final prototype.

6.1.6 Implement Product and Maintain

Once the final system is developed based on the final prototype, it is thoroughly tested and deployed to production. The system undergoes routine maintenance for minimizing downtime and prevent large-scale failures.

6.2 Description of Work Flow

6.2.1 Data Collection

The process of collecting data for AI training involves several key steps. First, the objective of the project is defined. Then, relevant data sources are identified, and the data is gathered from these sources. The collected data is preprocessed to ensure its quality and relevance for the task at hand. Annotation or labeling may be applied to the data to provide necessary information for training. If needed, the dataset is augmented to increase its size or diversity.

Once the dataset is ready, it is split into training, validation, and testing sets to assess the performance of the AI model. Throughout this process, privacy regulations and ethical considerations must be strictly adhered to, ensuring that sensitive information is protected. Maintaining data quality is crucial, and documentation of the collected data is important for efficient training and future reference.

In this specific project, the datasets will be sourced from popular websites such as Kaggle.com, as well as specific datasets like CelebA, among other relevant sources.

6.2.2 Data Preprocessing

Clean the collected data by removing HTML tags, special characters, punctuation, and stop words. Convert the text to lowercase and handle any missing or duplicate entries.

6.2.3 Feature Extraction

Extract relevant features from the preprocessed text, such as:

- Word frequencies: Count the occurrence of each word in the text.
- TF-IDF (Term Frequency-Inverse Document Frequency): Calculate the importance of a word in the context of the entire dataset.
- Linguistic patterns: Identify specific linguistic patterns or stylistic features associated with fake news.

6.2.4 Labelling

Label the data as "fake" or "real" based on ground truth labels, expert assessment, or reliable fact-checking sources.

6.2.5 Model Selection

6.2.5.1 Logistic Regression

Logistic Regression is a statistical method used for binary classification tasks, where the target variable (or output) is categorical and has only two possible outcomes, typically represented as 0 and 1. It's named "logistic" because it models the probability of the input belonging to a particular class using the logistic function.

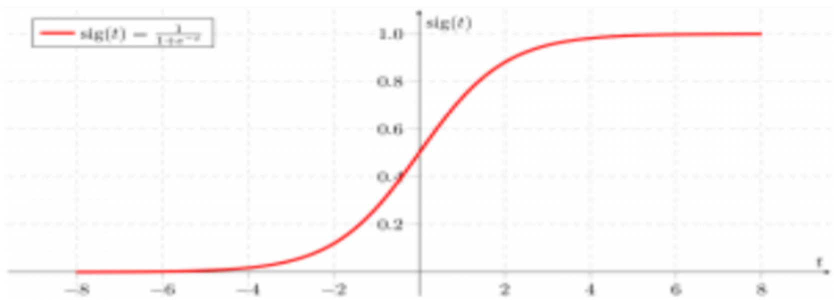

Figure 6.2: Logistic Regression

1. Model Representation:
   In Logistic Regression, we start with a linear equation, similar to linear regression. However, instead of directly predicting the output, we pass the linear combination of input features through a logistic (or sigmoid) function to obtain the probability of the input belonging to a particular class.

   Mathematically, the linear equation looks like this:

$z = b_0 + b_1 * x_1 + b_2 * x_2 + \ldots\ldots + b_n * x_n$

where z is the linear combination of input features and coefficients $(b_0, b_1, \ldots., b_n)$ and $(x_1, x_2, \ldots\ldots, x_n)$ are the input features.

2. Logistic (Sigmoid) Function:
The logistic function (also known as the sigmoid function) is used to map the linear combination z to a value between 0 and 1, representing the probability of the input belonging to the positive class (class1).

The logistic unction is defined as:
$P(y = 1|x) = 1/(1 + e^{-z})$

Where P(y = 1—x) is the probability of the input x belonging to class 1, and e is the base of the natural logarithm.

3. Decision Boundary:
Logistic regression predicts the class label based on whether the predicted probability (P(y = 1—x)) is greater than or equal to a predefined threshold (usually 0.5). If the probability is greater than the threshold, the input is classified as belonging to class ; otherwise, it's classified as belonging to class 0.

4. Training:
During training, the model learns the optimal values of the coefficients $(b_0, b_1, \ldots., b_n)$ that minimize the error between the predicted probabilities and the actual class labels. This is typically done using optimization algorithms like gradient descent.

5. Interpretability:
One of the advantages of Logistic Regression is its interpretability. Since it models probabilities, we can interpret the coefficients to understand the impact of each input feature on the likelihood of the input belonging to a particular class.

6.2.5.2 Support Vector Machine

Support Vector Machine (SVM) is another powerful algorithm commonly used for fake news detection. SVM is a supervised learning algorithm that can be applied to both classification and regression tasks. Here's how SVM can be used for fake news detection:

1. Decision Function:
The decision function of the SVM classifier is given by:
$f(x) = sign(\Sigma_{i=1}^{N} \alpha_i y_i K(x, x_i) + b)$
Where:
α_i are the learned Lagrange multipliers (dual coefficients).
b is the bias term.
$K(x, x_i)$ is the kernel function, which can be linear $(K(x, x_i) = x^T x_i)$ polynomial, radial basis function (RBF), etc.

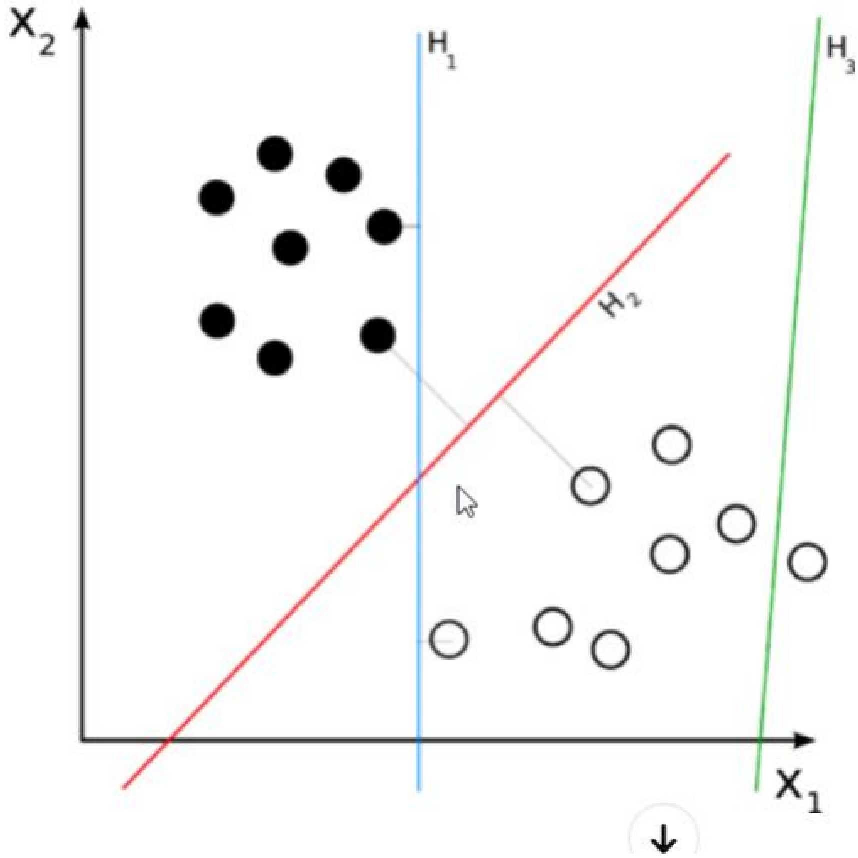

Figure 6.3: SVM for Linearly Seperable Data

2. Optimization Problem:
   The SVM classifier aims to find the hyperplane that separates the data with the maximum margin, which can be formulated as the following optimization problem:
   $min_{\alpha} 1/2 \Sigma_{i=1}^{N} \Sigma_{j=1}^{N} \alpha_i \alpha_j y_i y_j K(x_i, x_j) - \Sigma_{i=1}^{N} \alpha_i$
   Subject to:
   $0 \leq \alpha_i \leq C, for i = 1, ..., N$
   $\Sigma_{i=1}^{N} \alpha_i y_i = 0$
   where C is the regularization parameter.

3. Prediction:
   To prdict the class label of a new sample x, we use the decision function $f(x0$ as described above.

4. Kernel Trick:
   The kernel trick allows SVMs to implicitly map the input features into higher-dimensional space without explicitly computing the transformed features. This is achieved by replacing the inner product $x_T x_i$ with a kernel function $K(x, x_i$.
   By solving the optimization problem and obtaining the optimal values of α_i and b, we can construct the decision function f(x) to classify new samples.

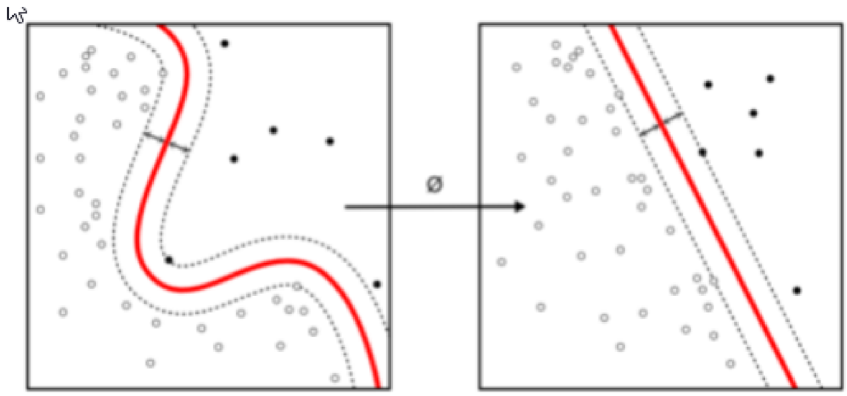

Figure 6.4: SVM for Non-Linearly Seperable Data

6.2.5.3 Long Short Term Memory(LSTM)

LSTMs are a type of recurrent neural network (RNN) architecture designed to process and analyze sequential data, such as text. They excel in capturing long-range dependencies and remembering important information over long sequences.

For fake news detection, LSTMs process news articles token by token, considering the sequential nature of the text. They contain memory cells with gates that control the flow of information, allowing them to selectively update and retrieve relevant information while discarding irrelevant details.

During training, LSTMs learn from labeled examples of news articles, adjusting their parameters to minimize prediction errors. They learn hierarchical representations of the text data, capturing both low-level features (individual words) and high-level semantic relationships (overall meaning).

Once trained, LSTMs can classify new news articles as real or fake by analyzing their sequential input data and producing a prediction based on the entire article. This prediction is typically based on the LSTM's output at the last time step or by aggregating information from multiple time steps.

Overall, LSTMs are effective for fake news detection due to their ability to understand the context and nuances of news articles, making them powerful tools for analyzing and identifying patterns indicative of fake news.

- x_t as the input at time step t,
- h_{t-1} as the previous hidden state (also known as the cell state) at time step t - 1,
- c_{t-1} as the previous cell state at time step t - 1,
- h_t as the new cell state at time step t, and
- c_t as the new cell state at time step t.

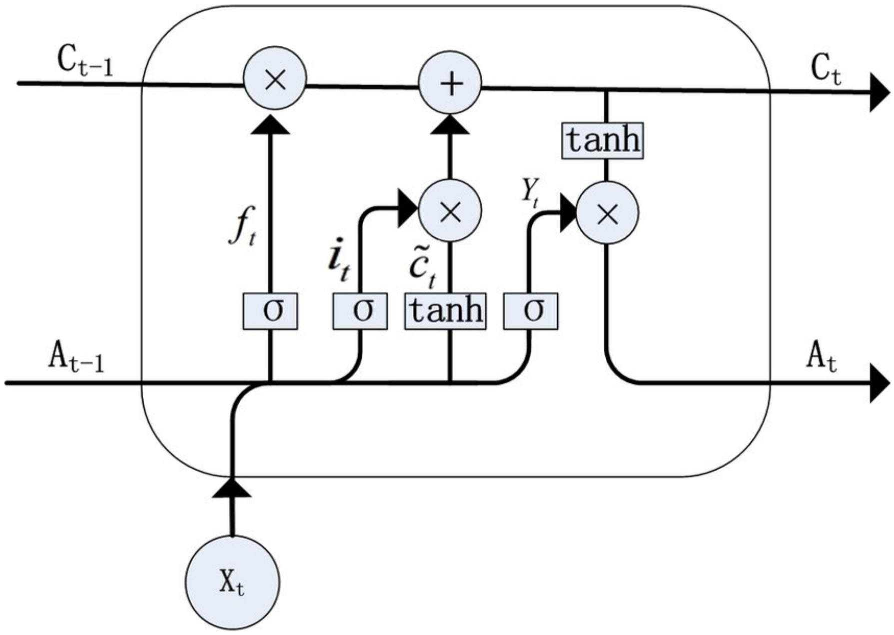

Figure 6.5: Long Short Term Memory

The LSTM cell consists of several gates and operations:

1. Forget Gate:
   Determines what information from the previous cell state c_{t-1} should be discarded.
   $f_t = \sigma(W_f.[h_{t-1}, x_t] + b_f)$

2. Input Gate:
   Determines what new information should be stored in the cell state.
   $i_t = \sigma(W_i.[h_{t-1}, x_t] + b_i)$
   $\overline{c_t}$=tanh(W$_c$.$[h_{i-1}, x_t] + b_i)$

3. Update cell State:
   Combines the information from the forget gate and input gate to update the cell state
   $c_t = f_t.c_{t-1} + i_t.\overline{c_t}$

4. Output Gate:
   Determines what information from the cell state should be output as the new hidden state.
   $o_t = \sigma(W_o.[h_{t-1}, x_t] + b_o)$
   $h_t = o_t.tanh(c_f)$

Where:

- σ is the sigmoid function,
- W_f, W_i, W_c, W_o are weight matrices for forget gate, input gate, input modulation and output gate respectively,

- W_f, W_i, W_c, W_o are weight matrices for forget gate, input gate, input modulation and output gate respectively,
- b_f, b_i, b_c, b_o are bias terms,
- $[h_{t-1}, x_t]$ denotes the concatenation of the previous hidden state and the current input, and
- $\overline{c_t}$ represents the candidate cell state.

These equations describe the operations performed within an LSTM cell at each time step t. The forget gate, input gate, and output gate control the flow of information, while the cell state c_t and hidden state h_t capture the long-term dependencies and output of the LSTM cell, respectively.

6.2.5.4 Bidirectional Encoder Representations from Transformers

To begin, the process starts with gathering a dataset comprising labeled news articles, where each article is categorized as either real or fake. This dataset serves as the foundation for training and evaluating the fake news detection model.

Next, a pre-trained BERT model is fine-tuned using the labeled dataset. Fine-tuning involves adjusting the model's parameters to specialize in the task of distinguishing between real and fake news articles. This step allows BERT to learn the contextual nuances and linguistic patterns indicative of fake news.

Once the fine-tuning process is complete, the model's performance is evaluated using a separate test dataset. Metrics such as accuracy, precision, recall, and F1-score are calculated to assess how well the model can classify news articles as real or fake.

After evaluation, the trained BERT model is deployed for inference. When a new news article is submitted for classification, it undergoes preprocessing (tokenization, padding), and then the processed text is fed into the trained model. The model predicts whether the article is real or fake based on its learned knowledge.

Optionally, additional post-processing steps or validation checks may be performed on the model's predictions. This could involve analyzing confidence scores, cross-referencing with external sources, or applying rules-based filters to flag ambiguous cases for human review.

Finally, the fake news detection system undergoes iterative improvement. Continuous monitoring of the system's performance allows for adjustments and enhancements. This iterative process may include collecting more labeled data, fine-tuning the model with different hyperparameters, or incorporating additional features or techniques to enhance the system's effectiveness in detecting fake news.

6.2.6 Model Evaluation

We will assess the performance of our implemented models. We will evaluate its performance using appropriate metrics such as accuracy, precision, F1-score, and

Range Score. You can use techniques like cross-validation to assess the model's generalization performance on unseen data.

6.2.7 Model Deployment

Following the optimization phase, incorporating continuous evaluation and feedback, our next step involves deploying the model and constructing the user interface (UI). For this purpose, we have chosen to utilize Django for UI development and Streamlit for deployment. Django will serve as the framework for building a robust and interactive user interface, offering features. Streamlit will facilitate the deployment process by providing a seamless platform for hosting the model and UI, ensuring smooth integration and accessibility for users.

Chapter 7

System Design and Architecture

7.1 Use Case Diagram

The Use Case Diagram of the prepared inference system.

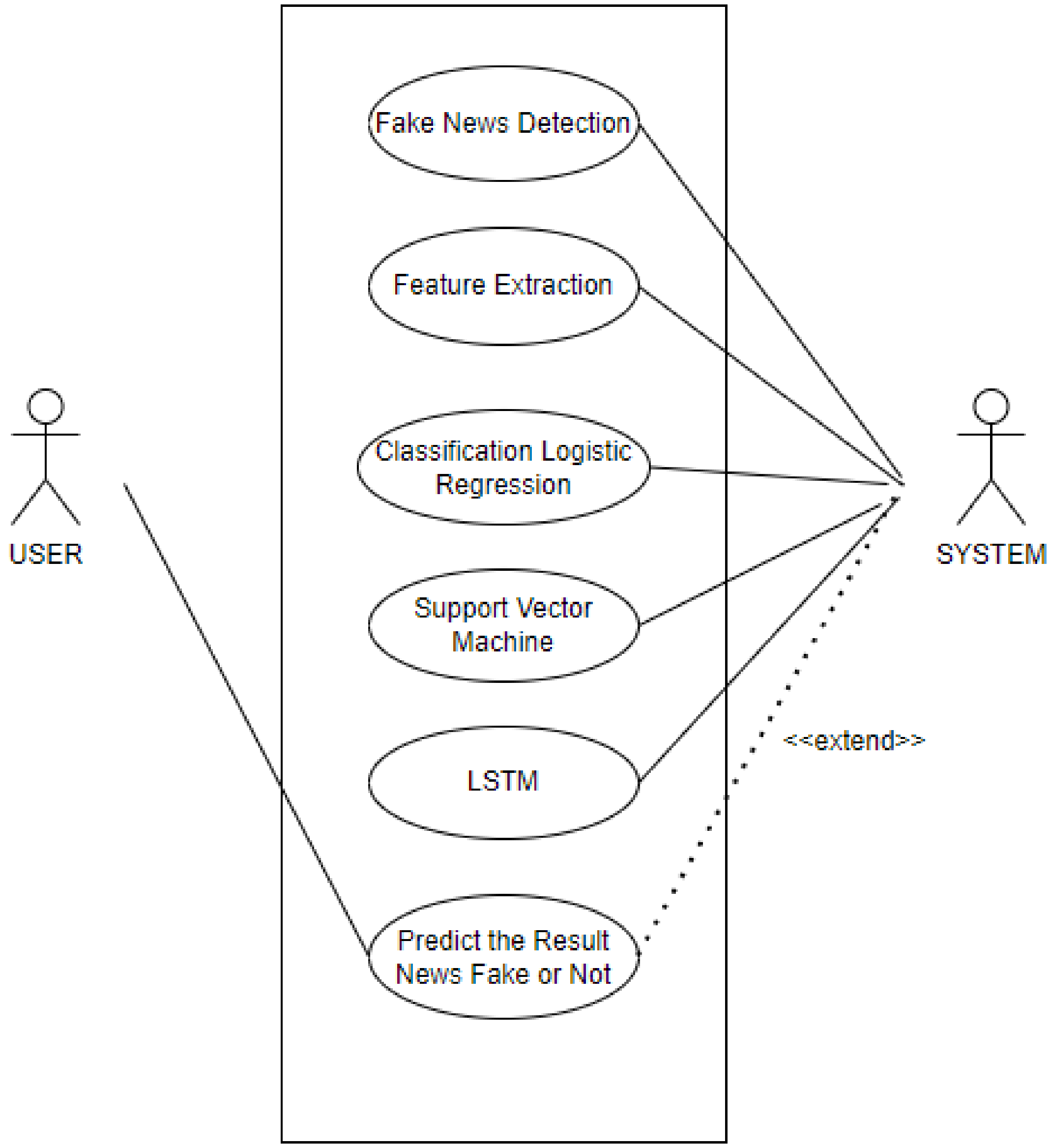

Figure 7.1: Use Case Diagram

7.2 Context Diagram

The Context Diagram shows the top level picture of the system.

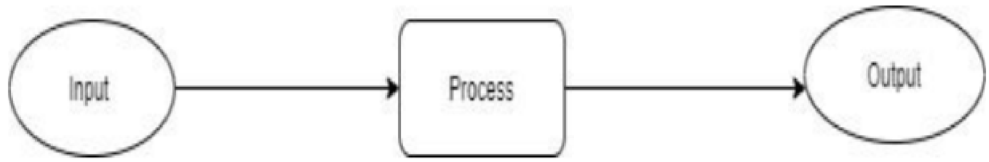

Figure 7.2: Context Diagram of Inference System

7.3 Data Flow Diagram

The Data Flow Diagram shows the flow of the data between the subsystem of the inference system. The level-1 data flow diagram of the inference system is shown below:

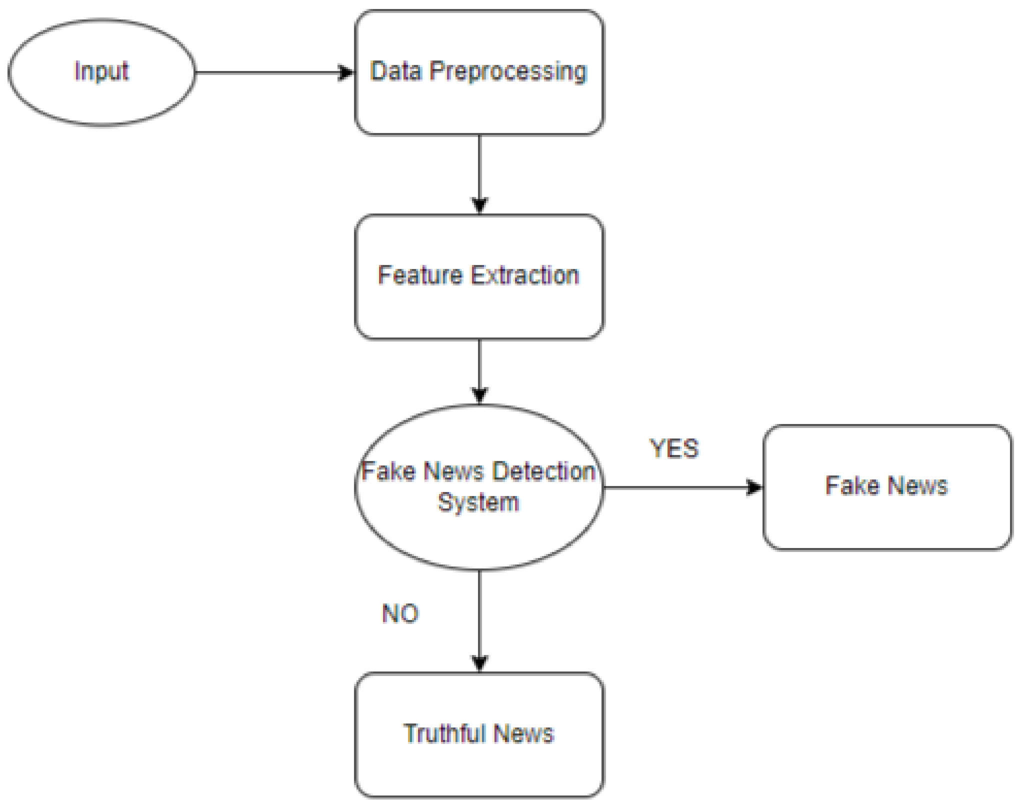

Figure 7.3: Level-1 Data Flow Diagram of Inference System

7.4 Block Diagram

We developed Fake news detection system which takes news as input and provide its actuality.

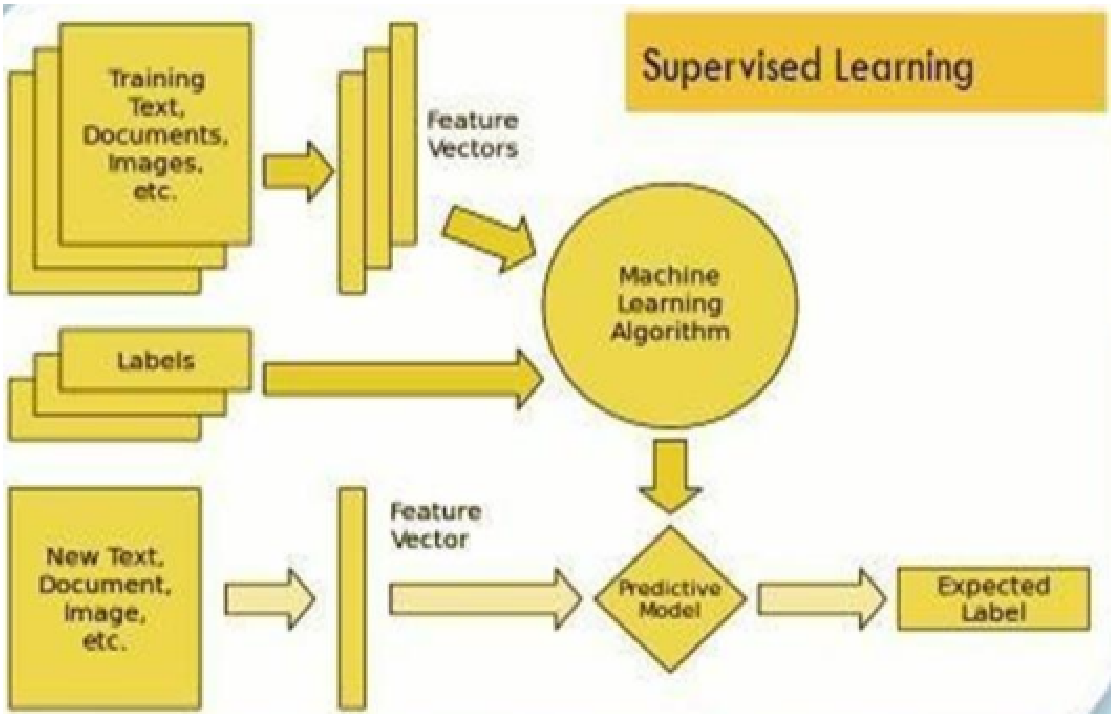

Figure 7.4: Block Diagram of Inference System

Chapter 8

Results and Discussion

8.1 Work Done

The project involved training and evaluating several machine learning models on datasets containing both fake and true news articles. Three primary models were utilized: LSTM (a type of recurrent neural network), BERT (a powerful transformer-based model), and SVM (Support Vector Machine). Each model was trained to classify news articles as either real or fake based on their content.

8.2 Work to be Done

- Building a Logistic Regression model.
- Evaluation and Comparison between these models.
- Scraping News from websites to find out similar news.

8.3 Result

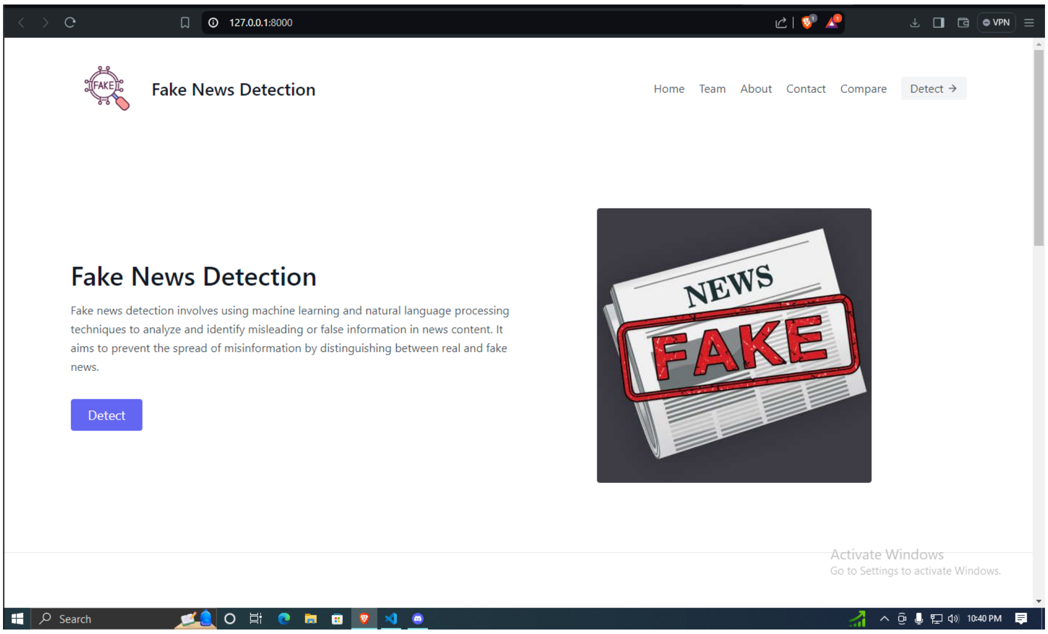

Figure 8.1: Home Page

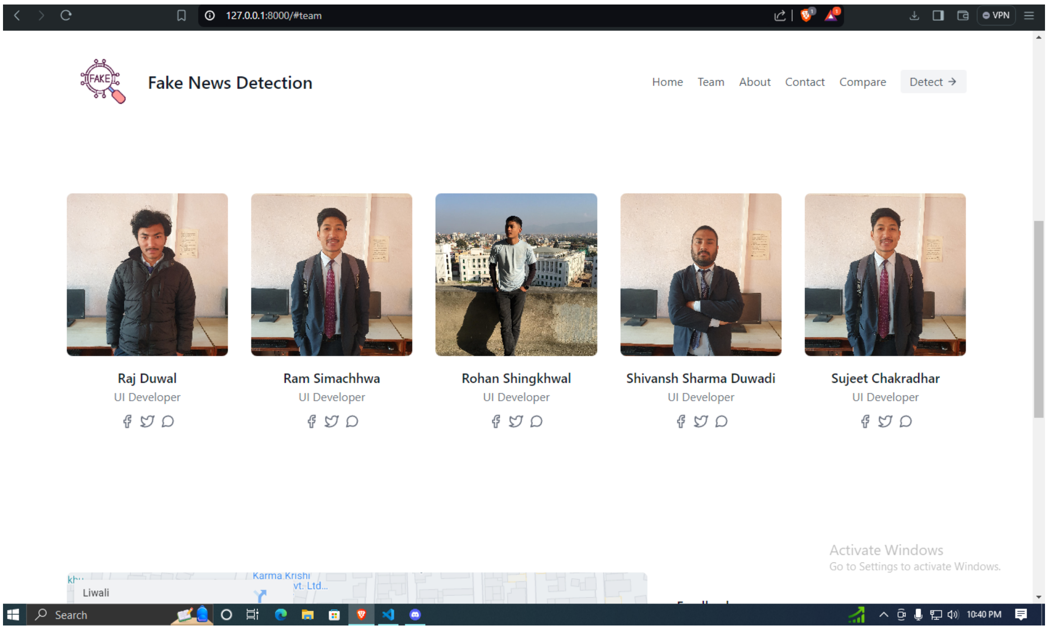

Figure 8.2: Teams Page

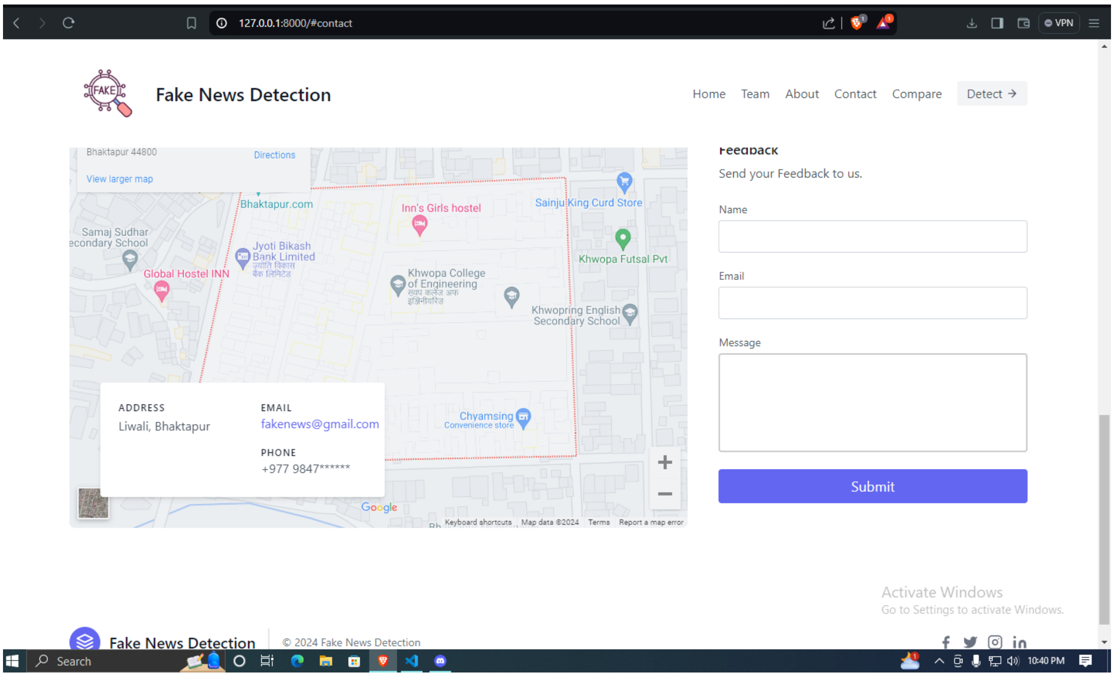

Figure 8.3: Contact Page

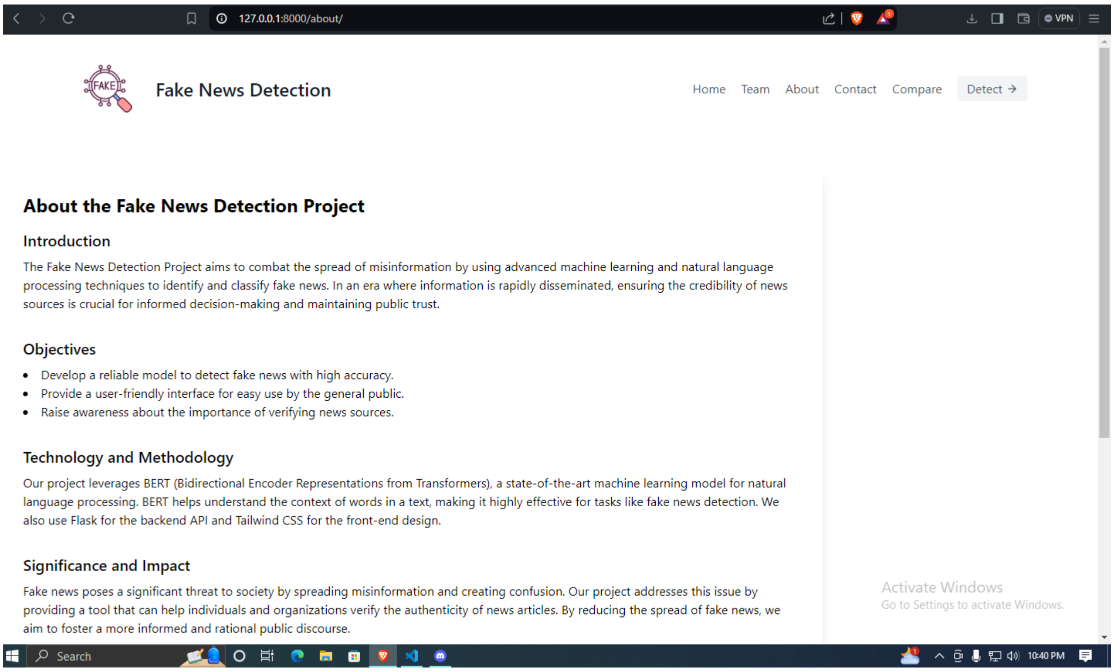

Figure 8.4: About Page

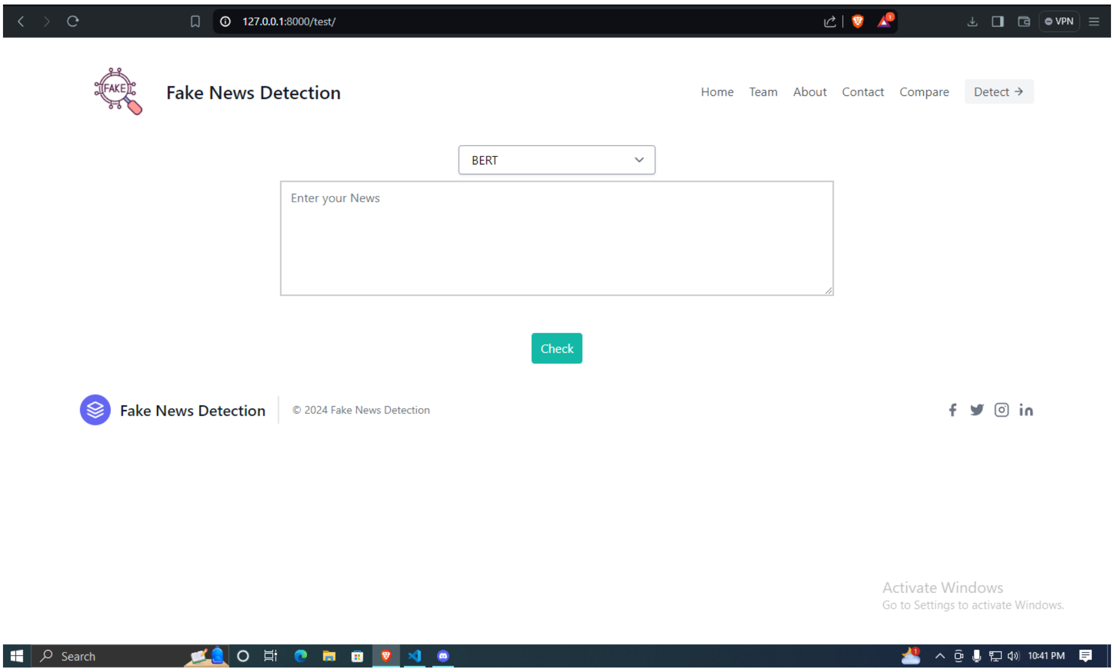

Figure 8.5: Detect Page (1)

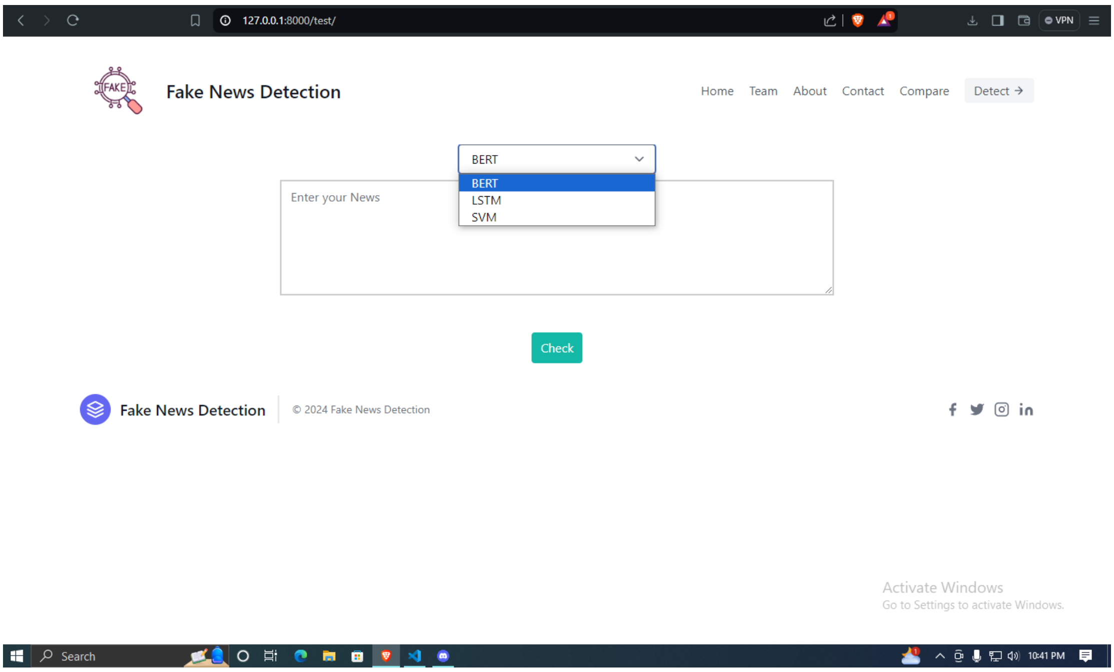

Figure 8.6: Detect Page (2)

Bibliography

[1]. Parikh, S. B., & Atrey, P. K. (2018, April). Media-Rich Fake News Detection: A Survey. In 2018 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR) (pp. 436-441). IEEE.

[2]. Conroy, N. J., Rubin, V. L., & Chen, Y. (2015, November). Automatic deception detection: Methods for finding fake news. In Proceedings of the 78th ASIS&T Annual Meeting: Information Science with Impact: Research in and for the Community (p. 82). American Society for Information Science.

[3]. Helmstetter, S., & Paulheim, H. (2018, August). Weakly supervised learning for fake news detection on Twitter. In 2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM) (pp. 274-277). IEEE. [4]. Stahl, K. (2018). Fake News Detection in Social Media.

[4]. Della Vedova, M. L., Tacchini, E., Moret, S., Ballarin, G., DiPierro, M., & de Alfaro, L. (2018, May). Automatic Online Fake News Detection Combining Content and Social Signals. In 2018 22nd Conference of Open Innovations Association (FRUCT) (pp. 272-279). IEEE.

[5] Tacchini, E., Ballarin, G., Della Vedova, M. L., Moret, S., & de Alfaro, L. (2017). Some like it hoax: Automated fake news detection in social networks. arXiv preprint arXiv:1704.07506.

[6]. Shao, C., Ciampaglia, G. L., Varol, O., Flammini, A., & Menczer, F. (2017). The spread of fake news by social bots. arXiv preprint arXiv:1707.07592, 96-104.

[7]. Chen, Y., Conroy, N. J., & Rubin, V. L. (2015, November). Misleading online content: Recognizing clickbait as false news. In Proceedings of the 2015 ACM on Workshop on Multimodal Deception Detection (pp. 15-19). ACM