Deception Detection

Mini-project Report

Submitted in partial fulfillment of the requirements

For the of

B.Tech.

Computer Science & Engineering (AIML)

by

Babita Sheoran (22AM1013)

Mythri Patel (22AM1125)

Aman Singh (22AM1023)

Nirav Kadam (22AM1075)

Supervisor

Dr. Shubhangi Ghate

D Y PATIL
DEEMED TO BE
UNIVERSITY
RAMRAO ADIK
INSTITUTE OF TECHNOLOGY
NAVI MUMBAI

Department of Computer & Science Engineering (AIML)
Ramrao Adik Institute of Technology,
Sector 7, Nerul , Navi Mumbai
(Under the ambit of D. Y. Patil Deemed to be University) April 25, 2025

D Y PATIL
DEEMED TO BE
UNIVERSITY
RAMRAO ADIK
INSTITUTE OF TECHNOLOGY
NAVI MUMBAI

Ramrao Adik Institute of Technology
(Under the ambit of D. Y. Patil Deemed to be University)
Dr. D. Y. Patil Vidyanagar,Sector 7, Nerul, Navi Mumbai 400 706.

Certificate

This is to certify that, the Mini-project titled

"Deception Detection "

is a bonafide work done by

Babita Sheoran (22AM1013)
Mythri Patel (22AM1125)
Aman Singh (22AM1023)
Nirav Kadam (22AM1075)

and is submitted in the partial fulfillment of the requirement for the degree of

Bachelor of Technology

Computer Science & Engineering (AIML)
to the
D. Y. Patil Deemed to be University.

Dr. Shubhangi Ghate Supervisor	Dr. Sandeep Sangle Project Coordinator
Dr. Sangita Chaudhari Head of Department	Dr. Mukesh Patil Principal

Dissertation Approval for B.Tech

This is certify the dissertation entitled **"Deception Detection"** is a bonafide work done by **Babita Sheoran (22AM1013), Mythri Patel (22AM1125), Aman Singh (22AM1023), Nirav Kadam, (22AM1075)** under the supervision of **Dr. Shubhangi Ghate**. This dissertation has been approved for the award of Mini Project in Computer Science and Engineering, D. Y. Patil Deemed to be University.

Examiners:

1:_____

2:_____

Supervisor:

1:_____

2:_____

Date : _____

Place : _____

Declaration

I declare that this written submission represents my ideas in my own word and where others ideas or words have been included, I have adequately cited and referenced the original source. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. I understand that any violation of above will be cause for disciplinary action by institute and can also evoke penal action from the source which have thus not been properly cited or from whom proper permission has not been taken when needed.

_____

_____

_____

_____

(Name of student and Roll No.)

Date : _____

Acknowledgement

We take this opportunity to express my profound gratitude and deep regards to my guide Dr. Shubhangi Ghate for her exemplary guidance, monitoring and constant encouragement throughout the completion of this report. We are truly grateful to her efforts to improve my understanding towards various concepts and technical skills required in our project. The blessing, help and guidance given by her time to time shall carry us a long way in the journey of life on which we are about to embark. We take this privilege to express my sincere thanks to Dr. Mukesh D. Patil, Principal, RAIT, D. Y. Patil deemed to be University for providing the much necessary facilities. We are also thankful to Dr. Sangita Chaudhari, Head of Department of Computer Science and Engineering, Dr. Sandeep Sangle, Mini Project Coordinator, for their generous support. Last but not the least we would also like to thank all those who have directly or indirectly helped us in completion of this project report.

Contents

Chapter 1

Introduction

In today's fast paced digital world, the internet has become our go-to source for news and information. With just a few clicks, we gain access to countless articles, blogs, and social media updates. However, this easy access comes with a major drawback the rapid spread of misinformation and fake news. Whether it's politically driven, clickbait for profit, or simply misleading content, fake news can shape public opinion, influence elections, and even incite panic.

Deception detection, particularly in the context of fake news, has emerged as a crucial field in recent years. It focuses on identifying whether a piece of information is genuine or deceptive by analyzing text patterns, language cues, source credibility, and more. With the rise of artificial intelligence and natural language processing, we now have the tools to automate the detection process and reduce human bias and workload.

This project aims to explore and implement techniques to detect deceptive content using machine learning and NLP-based approaches. By building a model that can differentiate between truthful and misleading articles, we take a step toward a more informed and trustworthy online environment.

1.1 Literature review

Over the past few years, fake news has evolved from being a buzzword to a serious global concern. As the internet became a primary source of information, the line between factual reporting and fabricated content began to blur.

Several studies have highlighted that fake news articles often use emotionally charged language, exaggerated claims, and unreliable sources. Techniques such as TF-IDF (Term Frequency–Inverse Document Frequency), word embeddings, and bag-of-words models have been widely adopted for feature extraction. On the classification side, algorithms like Logistic

Regression, Naive Bayes, Random Forest, and even deep learning models like LSTMs and Transformers have shown promising results in detecting deceptive patterns.

1.2 Motivation

The increasing influence of social media and digital platforms has made it incredibly easy to share news across the globe. While this connectivity has many benefits, it also opens the door for the rapid spread of fake or misleading information. We've all seen sensational headlines or viral posts that later turned out to be completely false. The consequences of such misinformation can be serious damaging reputations, misleading public health decisions, or swaying political outcomes.

What motivated this project is the growing need to combat fake news through technology. Instead of relying solely on manual fact checking which is time consuming and prone to human error we wanted to explore how machine learning and natural language processing could help us automatically flag deceptive content, making online spaces more trustworthy for everyone.

1.3 Objective

The main objective of this project is to develop a system that can effectively detect and classify fake news articles. The goals include:

Collecting and preparing a reliable dataset of real and fake news samples.

Analyzing linguistic and statistical features that distinguish truthful content from deceptive writing.

Building and training machine learning models that can automatically identify fake news with high accuracy.

Evaluating the model's performance using standard metrics like accuracy, precision, recall, and F1 score.

Ultimately, the project seeks to demonstrate how artificial intelligence can be a powerful tool in promoting digital truth and transparency.

1.4 Scope for Work

This project focuses on textual deception detection in online news articles. It does not address multimedia content such as images or videos. The scope includes:

Data preprocessing, feature extraction, and exploratory data analysis of news datasets.

Implementation of NLP techniques to understand the text structure and tone.

Application of various machine learning algorithms like Logistic Regression, Random Forest, and possibly deep learning models for improved accuracy.

Analysis of the model's results and comparison between different approaches.

While this project is an academic exploration, the concepts and methodologies can be extended to real-world applications, such as integration into browser plugins, news apps, or social media moderation tools.

1.5 Organization of Report

The report details the various components of the software, including the coding languages used. It aims to demonstrate the various environment variables used for the software and the way in which the software works.

It also showcases some of the screen snips along with the code snippet. This report explains the front-end and back-end codes, database software used, environment variables, authentication software and API declaration.

It also aims to highlight the applications of the software and the future scope, along with the additional implementations that can be done on it, along with references.

Chapter 2

Literature Survey

Over the past few years, fake news has become a serious problem across the globe. Researchers from different fields have come together to understand not only how fake news spreads, but also how it affects people and how we can detect it more effectively. Here are some key studies that helped shape our understanding of deception in news.

In 2014, Meital Balmas examined what happens when people are exposed to fake news regularly. The study found that repeated exposure especially from multiple sources can actually change how people feel about politics. It often leads to feelings of confusion, cynicism, and helplessness, showing that fake news doesn't just misinform, it can damage trust in the system altogether.

Michele Banko and her team (2007) focused on extracting important facts from web pages using an automated system. While their work wasn't specifically about fake news, it created tools that are incredibly useful today for pulling out key information from online content which is a huge part of how modern fake news detectors work.

In 2016, Alessandro Bessi and Emilio Ferrara looked into how social bots automated accounts on platforms like Twitter played a role in the U.S. presidential election. Their research revealed that bots were responsible for spreading a large amount of political misinformation, often making fake stories go viral before anyone could stop them.

Another important study by Prakhar Biyani and colleagues (2016) tackled the issue of clickbait those dramatic, curiosity pulling headlines like "You Won't Believe What Happened Next!" They developed a model that could detect these types of misleading titles based on how informal or exaggerated the language was.

Jonas Blom and Kenneth Hansen (2015) took a different angle by analyzing how news headlines are written. They noticed that fake or misleading headlines often leave out important details on purpose, forcing readers to click. This tactic, known as forward reference, is very common in clickbait and fake news.

Chapter 3

Methodology

To build a reliable fake news detection system, we followed a structured and iterative process that combines natural language processing (NLP) with machine learning (ML) techniques. Each step was carefully designed to help the model understand, learn, and predict whether a given news article is real or fake.

Programming Language : Python is chosen due to its extensive libraries and support for machine learning.

Machine Learning Libraries Scikit-learn for implementing machine learning algorithms. NLTK for text preprocessing and NLP tasks.

Feature Extraction Techniques TF-IDF (Term Frequency-Inverse Document Frequency): Measures word importance in a document relative to a dataset. Word Embeddings (Word2Vec): Captures semantic relationships between words.

Evaluation Metrics Accuracy: Measures overall correctness. F1-score: Balances precision and recall for a better assessment of model performance.

3.1 Working of the program

The Deception Detection model uses a machine learning approach to classify news articles as either real or fake. The workflow is as follows:

Dataset Loading - The dataset contains news entries with attributes such as ID, title, author, text, and a label (1 = Fake, 0 = Real). It is read into a DataFrame using Pandas.

Data Preprocessing - Missing values are handled by replacing them with empty strings. Title and author fields are merged to enrich the content. Stopwords are removed using NLTK, and words are reduced to their root form using PorterStemmer (stemming).

Feature Extraction - The cleaned content is converted into numerical format using TF-IDF Vectorizer for model training.

Model Training - The preprocessed text is split into features (X) and labels (Y). A Logistic Regression model is trained to classify the news as real or fake. Model evaluation is done using accuracy metrics.

3.2 Starting the program

Here's how to start and run the program step-by-step in Google Colab:

Install required libraries (if not already available): python !pip install nltk scikit-learn pandas

Download NLTK stopwords: python import nltk nltk.download('stopwords')

Run the cells sequentially:
- Import all dependencies
- Load the dataset
- Apply preprocessing: merging, cleaning, stemming.
- Extract features using TfidfVectorizer.
- Train the logistic regression model and evaluate it.

3.3 Navigation of the program

Step 1: Frontend

- Users paste a news article into a text box and click a "Check News" button.
- JavaScript sends the article to the backend using a POST request.

Step 2: Backend(Flask API):

- Flask receives the JSON containing the article's content at the /predict endpoint.
- The model processes the input and predicts whether it's real or fake.
- The result is returned as a JSON object: json

Step 3: Frontend

- Parses the response and displays the result dynamically on the webpage.

This structured navigation ensures smooth interaction and a real-time response mechanism between the UI and the model.

3.4 Flowchart

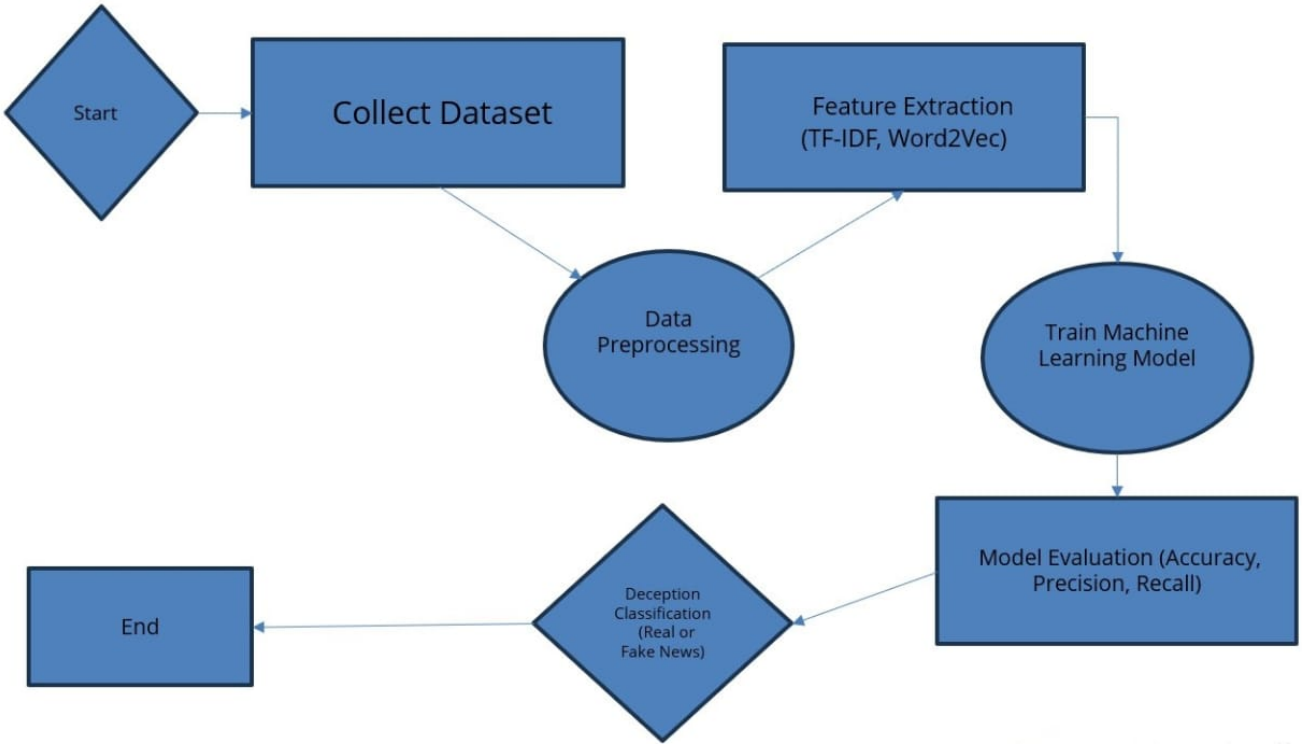

Figure 1: Flowchart of deception detection

3.5 Algorithm

Step 1: Start the Program
Begin by initializing the necessary libraries such as NumPy, Pandas, Scikit-learn, and Natural Language Toolkit (NLTK). Also, prepare the environment for data preprocessing and model building.

Step 2: Load the Dataset
Import the dataset containing labeled news articles (e.g., real or fake). This data may include titles, body text, authorship, and source information.

Step 3: Preprocess the Data
Clean and prepare the data for analysis. This involves:
Converting text to lowercase
Removing punctuation and special characters
Eliminating stop words (like “the”, “is”, “at”)
Tokenizing the text (splitting it into words)
Applying stemming or lemmatization to reduce words to their root form

Step 4: Convert Text to Numerical Features
Since machine learning models require numerical input, convert the processed text into feature vectors

Step 5: Split the Dataset
Divide the dataset into training and testing sets—typically 80

Step 6: Train the Machine Learning Model
Use a classification algorithm to train the model on the training set:Logistic Regression

Step 7: Evaluate the Model
Test the trained model on the testing dataset.

Step 8: Predict on New Inputs
Allow the model to predict the label (real/fake) of new articles entered by a user. Display the result with appropriate messaging.

Step 9: End the Program
After prediction and evaluation, the program ends. Optionally, the model can be saved for future use or integrated into a simple web interface.

3.6 Hardware Requirements

To run this project efficiently, only a modest hardware setup is needed, as the primary tasks involve data processing, training lightweight machine learning models, and text analysis.

3.7 Software Requirements

The software setup for this project is designed to be lightweight, open-source, and beginner-friendly. It allows for seamless development, testing, and deployment of a machine learning model to detect fake news articles. Below is a breakdown of the essential software components and why they were chosen.

1. Operating System: Windows 10/11, Linux (Ubuntu 20.04 or later), or macOS The project is platform-independent and can run on any standard modern OS. We primarily worked in a Windows environment, but all tools used are cross-compatible.

2. Python (Version 3.7 or higher): Python is widely used in the field of data science and machine learning due to its simplicity, readability, and massive library support.

3. Integrated Development Environment (IDE) Options:

Jupyter Notebook: Great for interactive coding, visualizing outputs, and step-by-step experimentation.

VS Code / PyCharm: Ideal for larger codebases, script-based execution, and debugging.

4. Python Libraries and Frameworks
Here are the main libraries used in the project:

pandas: For reading datasets, data cleaning, and tabular manipulation. It simplifies working with structured data.

numpy: Provides support for numerical operations and arrays, which are used throughout the ML pipeline.

scikit-learn: The core machine learning library used to train, evaluate, and test classification models such as Logistic Regression, Random Forest, and Naive Bayes.

nltk or spaCy: For text preprocessing and natural language processing tasks like tokenization, stop word removal, and lemmatization.

matplotlib and seaborn: Used for plotting confusion matrices, visualizing performance metrics, and displaying trends in data.

Chapter 4

Result Analysis

Chapter 5

5.1 Future Scope

While our fake news detection system shows promising results in identifying deceptive content based on text, there is still significant room for growth and improvement. As misinformation tactics evolve, so must the technologies used to combat them. Below are some directions in which this project can be expanded in the future:

1. Multimodal Fake News Detection: Currently, our system focuses only on textual content. However, fake news often comes with images, videos, or misleading headlines that can amplify its impact. Incorporating image analysis, video verification, and even audio sentiment detection could help build a more robust and comprehensive solution.

2. Real-Time Detection: Integrating the model into a browser extension or social media filter can enable real-time detection of deceptive content as users browse the internet. This would provide instant feedback and help users make more informed decisions.

3. Language and Regional Expansion: Most fake news detectors are trained on English datasets. Expanding the model to understand and classify content in multiple regional languages would make it more inclusive and effective, especially in diverse linguistic regions like India.

4. Source Credibility Scoring: A future system could not only analyze the article's content but also assess the credibility of the source, flagging lesser-known or suspicious websites and ranking articles based on trustworthiness.

5.2 Conclusion

Working on this project has been both eye opening and incredibly relevant to the times we live in. Every day, we're surrounded by news some true, some misleading, and some intentionally fake. And as we scroll through social media or browse headlines online, it's not always easy to tell the difference. That's what inspired us to take on this challenge, to build something that helps people make better informed decisions in a world full of noise.

Through this project, we explored how technology specifically machine learning and natural language processing can be used to spot fake news based on the way it's written. From cleaning messy text to training models that learn how deceptive articles are structured, every step of the process taught us something new. And while the final system isn't perfect, it's a solid starting point one that shows how even a simple model can make a real difference.

More importantly, this project helped us understand the responsibility that comes with creating tools like this. It's not just about writing code or getting good accuracy; it's about helping people trust the information they see, and in a way, helping protect the truth.

We know there's still a long way to go fake news is constantly evolving but we're proud of what we've built so far. And with more data, better models, and real world integration, we truly believe that systems like ours can become valuable tools in the fight against misinformation.

Chapter 6

References

[1] Sadia Afroz, Michael Brennan, and Rachel Greenstadt. Detecting hoaxes, frauds, and deception in writing style online. In *ISSP'12*.

[2] Hunt Allcott and Matthew Gentzkow. Social media and fake news in the 2016 election. Technical report, National Bureau of Economic Research, 2017.

[3] Solomon E. Asch and H. Guetzkow. Effects of group pressure upon the modification and distortion of judgments. *Groups, leadership, and men*, pages 222–236, 1951.

[4] Meital Balmas. When fake news becomes real: Combined exposure to multiple news sources and political attitudes of inefficacy, alienation, and cynicism. *Communication Research*, 41(3):430–454, 2014.

[5] Michele Banko, Michael J Cafarella, Stephen Soderland, Matthew Broadhead, and Oren Etzioni. Open information extraction from the web. In *IJCAI'07*.

[6] Alessandro Bessi and Emilio Ferrara. Social bots distort the 2016 us presidential election online discussion. *First Monday*, 21(11), 2016.

[7] Prakhar Biyani, Kostas Tsioutsiouliklis, and John Blackmer. "8 amazing secrets for getting more clicks": Detecting clickbaits in news streams using article informality. In *AAAI'16*.

[8] Jonas Nygaard Blom and Kenneth Reinecke Hansen. Click bait: Forward-reference as lure in online news headlines. *Journal of Pragmatics*, 76:87–100, 2015.

[9] Paul R Brewer, Dannagal Goldthwaite Young, and Michelle Morreale. The impact of real news about fake news: Intertextual processes and political satire. *International Journal of Public Opinion Research*, 25(3):323–343, 2013.

[10] Carlos Castillo, Mohammed El-Haddad, Jürgen Pfeffer, and Matt Stempeck. Characterizing the life cycle of online news stories using social media reactions. In *CSCW'14*.