DAYANANDA SAGAR UNIVERSITY

KUDLU GATE, BENGALURU – 560068

Bachelor of Technology
in
COMPUTER SCIENCE AND TECHNOLOGY

Project Phase-II Report

(20CT4802)

TruthTracker-Combatting Misinformation and Deepfakes Online

By

DHANUSH S - ENG20CT0010
DHARSHAN K - ENG19CT0010
BHOOMIKA S - ENG20CT0007
KEERTHI A REDDY - ENG20CT0012

Under the supervision of

Prof. Chithambarathanu M
Assistant Professor
Department of Computer Science and Technology

SCHOOL OF ENGINEERING

Department of Computer Science & Technology

Kudlu Gate, Bengaluru – 560068

Karnataka, India

CERTIFICATE

This is to certify that the work titled "**TruthTracker-Combatting Misinformation and Deepfakes Online**" is carried out by **Dhanush S (ENG20CT0010), Dharshan K (ENG19CT0010), Bhoomika S (ENG20CT0007), Keerthi A Reddy (ENG20CT0012)** Bonafide students of Bachelor of Technology in Computer Science and Technology at the School of Engineering, Dayananda Sagar University, Bengaluru in partial fulfillment for the award of degree in Bachelor of Technology in Computer Science and Technology, during the year **2023-2024**.

Prof. Chithambarathanu M
Assistant Professor, Dept. of CST,
School of Engineering,
Dayananda Sagar University.

Dr. M Shahina Parveen
Chairperson CST,
School of Engineering,
Dayananda Sagar University.

Department of Computer Science & Technology
Kudlu Gate, Bengaluru – 560068
Karnataka, India

DECLARATION

We, **Dhanush S (ENG20CT0010), Dharshan K (ENG19CT0010), Bhoomika S (ENG20CT0007), Keerthi A Reddy (ENG20CT0012)** are students of the seventh semester B.Tech in Computer Science and Technology, at School of Engineering, Dayananda Sagar University, hereby declare that the project phase-II project titled "**TruthTracker-Combatting Misinformation and Deepfakes Online**" has been carried out by us and submitted in partial fulfillment for the award of degree in Bachelor of Technology in Computer Science and Technology during the academic year **2023-2024**.

Student Signatures

Name1 : Bhoomika S

USN : ENG20CT0007

Name2 : Dharshan K

USN : ENG19CT0010

Name3 : Dhanush S

USN : ENG20CT0010

Name4 : Keerthi A Reddy

USN : ENG20CT0012

Place : Bengaluru

Date :

ACKNOWLEDGEMENT

It is a great pleasure for us to acknowledge the assistance and support of many individuals who have been responsible for the successful completion of this project work.

First, we take this opportunity to express our sincere gratitude to School of Engineering, Dayananda Sagar University for providing us with a great opportunity to pursue our Bachelor's degree in this institution.

We would like to thank **Dr. Udaya Kumar Reddy K R, Dean, School of Engineering & Technology, Dayananda Sagar University** for his constant encouragement and expert advice.

It is a matter of immense pleasure to express our sincere thanks to **Dr. M Shahina Parveen, Chairperson**, **Computer Science and Technology**, **Dayananda Sagar University,** for providing right academic guidance that made our task possible.

We would like to thank our **Project Coordinators Prof. Bhaskar Venugopalan** and **Prof. Chithambarathanu M** as well as all the staff members of Computer Science and Technology for their support. We are also grateful to our family and friends who provided us with every requirement throughout the course.

We would like to thank our guide **Prof. Chithambarathanu M**, **Assistant Professor**, **Dept. of Computer Science and Technology**, **Dayananda Sagar University**, for sparing his valuable time to extend help in every step of our project work, which paved the way for smooth progress and fruitful culmination of the project.

We would like to thank one and all who directly or indirectly helped us in the Project work.

ABSTRACT

TruthTracker is a machine learning-based system designed to combat the growing issue of misinformation and deepfakes online. The project consists of two primary components: a fake news detection system and a deepfake image detection system.

The fake news detection system analyzes news articles to determine their authenticity. It utilizes various machine learning models, including Logistic Regression, Decision Tree, Random Forest, and Gradient Boosting. The process involves loading and preprocessing datasets, converting text data into numerical form using TF-IDF vectorization, and training machine learning models for classification. This system also allows users to manually input and test news articles.

The deepfake image detection system focuses on identifying manipulated images, starting with face detection using MTCNN, followed by feature extraction using the InceptionResnetV1 model. To improve interpretability, GradCAM highlights the image areas influencing the model's decision. TruthTracker's deep learning model, combining ResNetV1.5 CNN and LSTM-based RNN, effectively detects deepfake faces, achieving high accuracy. Comprehensive testing confirms the model's ability to halt deepfake spread. A user-friendly interface, powered by Gradio, ensures easy interaction with the system.

Keywords: misinformation, fake news, deepfakes, machine learning, text preprocessing, face detection, feature extraction, MTCNN, InceptionResnetV1, TF-IDF, Logistic Regression, Decision Tree, Random Forest, Gradient Boosting, GradCAM, user interface, Gradio.

LIST OF FIGURES

TABLE OF CONTENTS

CHAPTER 1
INTRODUCTION

In the digital age, misinformation and deepfake technology pose significant challenges to online information integrity. Misinformation, often in the form of fake news, can influence public opinion, disrupt social harmony, and erode trust in legitimate news sources. Deepfake technology, which creates hyper-realistic but fabricated videos and images, blurs the line between reality and fiction, raising concerns about privacy, security, and ethical implications.

To address these issues, TruthTracker, a machine learning-based system, is designed to detect and combat fake news and deepfake content. The system consists of two main components: a fake news detection module and a deepfake image detection module.

TruthTracker is a system that uses machine learning techniques to detect and combat fake news and deepfake content. Its primary goal is to identify and mitigate the spread of false information on the internet.

1.1 Definition of Fake News Articles

Fake news articles are fabricated news stories created to deceive readers and manipulate public opinion. They often contain false or misleading information and are disseminated through various channels, including social media and email.

1.2 Fake News Detection

The fake news detection module of TruthTracker analyzes textual content to determine its authenticity. By utilizing machine learning classifiers, this module processes and evaluates news articles, classifying them as either genuine or fake. Leveraging labeled news article datasets and techniques such as text preprocessing and TF-IDF vectorization, the module

aims to provide accurate and reliable results, ensuring a robust defense against misinformation.

1.3 Definition of Deepfake Images

Deepfake images are fabricated images created using deep learning techniques. These images often involve manipulating existing images or videos to create hyper-realistic but entirely fabricated content.

1.4 Deepfake Image Detection

The deepfake detection module of TruthTracker identifies manipulated images by combining facial detection, feature extraction, and explainability techniques. Using advanced methods such as MTCNN for face detection and InceptionResnetV1 for feature extraction, the system effectively analyzes facial features to detect deepfakes. GradCAM enhances the interpretability of the model's decisions, and a user-friendly interface developed with Gradio allows seamless interaction with the system, making it accessible for users to verify image authenticity.

CHAPTER 2

LITERATURE REVIEW

[1]. Fake News Detection Using a Logistic Regression Model and Natural Language Processing Techniques, 2023

Journal:

Research Square

Authors:

Johnson Adeleke Adeyiga, Philip Gbounmi Toriola, Temitope Elizabeth Abioye, Adebisi Esther Oluwatosin, Oluwasefunmi 'Tale Arogundade

Problem mentioned:

In this paper, the spread of fake news is a major concern, affecting democracy, journalism, and people's daily lives. Traditional fact-checking methods are not enough to handle the large amount of information, and automated detection systems are needed.

Tools Used:

Data Handling: Kaggle dataset, CSV files
Pre-processing: Python, NLTK
Feature Extraction: TF-IDF Vectorizer
Modeling: Logistic Regression, KNN, Passive Aggressive, Naïve Bayes
Evaluation: Accuracy, Precision, Recall, F1 Score
Implementation: Google Colab, Flask
Front-end: HTML, CSS, JavaScript

Results and Discussion:

The results showed that Logistic Regression performed well in detecting fake news, with high precision, recall, and F1 score. The confusion matrix analysis revealed that the model correctly identified most cases, with some errors. The ROC curve, precision-recall curve, and learning curve also demonstrated the model's effectiveness and ability to generalize.

Knowledge Acquired:

This study found that Logistic Regression is effective in detecting fake news, as shown by its strong performance in precision, recall, and F1 score. The analysis of evaluation metrics and visualizations provided insights into the model's behavior and generalizability. The comparison with other studies highlighted the superiority of Logistic Regression in combating misinformation.

[2]. Fake News Detection using Machine Learning: A Comprehensive Analysis, 2022

Journal/Conference:

Journal of Management and Service Science

Authors:

Nidhi Singh Kushwaha, Pawan Singh

Problem Statement and Solution:

The spread of fake news is a major concern, and this paper proposes a solution using machine learning techniques. The system has three components: a static component for training classifiers, a dynamic component for online fact-checking, and a component for authenticating URLs. The system was built using Python, Django, and web scraping techniques.

Algorithm Used:

The paper uses machine learning algorithms, including Logistic Regression, Decision Tree Classifier, and Random Forest Classification, to detect fake news. These algorithms are compared across three stages of machine learning strategies. The paper also explores four different ML algorithms, including Multinomial Naïve Bayes and Passive Aggressive Classifier, to train classifiers to predict the classification of text.

Result and Discussion:

The paper presents the implementation of machine learning algorithms using Count Vectors and Tf-Idf vectors. The results show that Logistic Regression achieved 80% accuracy after optimization, while the dynamic system using Passive Aggressive Classifier demonstrated high precision, recall, F1-score, and accuracy at 93%. The developed fake news detection system uses NLP and machine learning techniques to verify information online and authenticate websites.

Knowledge Acquired:

This study contributes to the field of fake news detection by showing the effectiveness of machine learning algorithms in distinguishing between real and fake news articles. The paper demonstrates the potential of machine learning in addressing the challenges posed by misinformation on social media and digital platforms. Additionally, the paper highlights the importance of continuous updates to the dataset through web crawling to enhance the accuracy of fake news detection systems over time.

[3]. Fake News Classification Using Random Forest and Decision Tree (J48), 2020

Journal:

ResearchGate

Authors:

Reham Jehad Al-Shammari, Suhad A. Yousif

Problem mentioned:

The spread of fake news is a major concern, and this paper addresses the problem of detecting fake news, especially in social media and politics. The authors highlight the challenges associated with identifying fake news, including the limited availability of benchmark datasets and the rapid influx of news publications.

Algorithm Used:

The paper uses two classification algorithms for detecting fake news: Decision Tree (specifically the J48 algorithm) and Random Forest. These algorithms are chosen for their effectiveness in classification tasks and their ability to handle both numerical and categorical data.

Results and Discussion:

The experiment showed that the decision tree classifier achieved 89.11% accuracy, surpassing the random forest's 84.97%. Preprocessing significantly improved the accuracy of both classifiers. The decision tree's success is due to its feature importance, while random forest's strength lies in handling large datasets. Compared to prior studies, the method using the

decision tree outperformed in accuracy. Overall, the decision tree excels for this fake news dataset, and preprocessing significantly boosts classification results.

Knowledge Acquired:

This study contributes to the field of fake news detection by showing the effectiveness of machine learning algorithms in distinguishing between real and fake news articles. The paper demonstrates the potential of Decision Tree and Random Forest algorithms in addressing the challenges posed by misinformation on social media and digital platforms. Additionally, the paper highlights the importance of preprocessing in improving the accuracy of fake news detection systems.

[4]. A smart System for Fake News Detection Using Machine Learning, 2019

Journal:

ResearchGate

Authors:

Anjali Jain, Avinash Shakya, Harsh Khatter, Amit Kumar Gupta

Problem mentioned:

The problem addressed in this paper is the detection of fake news, particularly through social media platforms, using machine learning and natural language processing techniques. The proposed model aims to accurately classify news articles as either real or fake, thereby mitigating the harmful effects of misinformation on society.

Algorithm Used:

The paper uses two classification algorithms for detecting fake news: Naive Bayes Classifier and Support Vector Machine (SVM).

Results and Discussion:

The proposed model outperformed four existing approaches, achieving up to 93.50% accuracy. The implementation and results section compared the proposed model with existing approaches using Python programming in R studio. The Naive Bayes classifier and SVM were found to be effective in identifying fake news articles.

Knowledge Acquired:

This study contributes to the field of fake news detection by comparing the effectiveness of different algorithms in identifying fake news articles. The paper demonstrates the potential of Naive Bayes classifier and SVM in addressing the challenges posed by misinformation on social media and digital platforms. Additionally, the paper highlights the importance of accurately detecting fake news and suggests avenues for future research to improve detection methods and enhance prototype efficiency and user interface.

[5]. Deepfake detection, 2024

Journal:

Journal of Emerging Technologies and Innovative Research (JETIR)

Authors:

Dr. P.Sruthi, Dr.T.Bhaskar, Bommagalla Ankitha, Duggirala Mercy Sunada, Yellanur Bhargavi, Yerra Manasa

Problem mentioned:

The spread of fake news is a major concern, and this paper addresses the problem of detecting fake news, especially in social media and politics. The authors highlight the challenges associated with identifying fake news, including the limited availability of benchmark datasets and the rapid influx of news publications.

Algorithm Used:

The paper uses a ResNetV1.5 CNN for feature extraction and LSTM networks for analyzing temporal dynamics in video frames. It also employs pretrained models like ResNetV1.5 and VGGFace2, along with MTCNN for facial image analysis and GradCAM for visualizing important areas, to enhance the accuracy and transparency of deepfake detection.

Results and Discussion:

The study's results show that the deepfake detection system effectively distinguishes between real and altered images with high accuracy. The research concludes that integrating advanced deep learning architectures significantly enhances deepfake detection capabilities,

highlighting the importance of ongoing development to counteract evolving deepfake technologies and maintain digital media integrity.

Knowledge Acquired:

This study contributes to the field of deepfake detection by demonstrating the potential of deep learning, particularly CNNs and LSTMs, in analyzing spatial and temporal features of deepfake images. The paper highlights the importance of diverse datasets like VGGFace2 for training models, fine-tuning pretrained models such as ResNetV1.5 for better accuracy, and advanced techniques like MTCNN for face detection and GradCAM for model interpretation. The study underscores the use of performance metrics like AUC and precision-recall curves to evaluate model effectiveness and emphasizes the need for ongoing development to counter evolving deepfake technologies and protect digital media integrity.

[6]. Deepfake Detection using Convolutional Neural Networks, 2023

Journal:

FUW TRENDS IN SCIENCE & TECHNOLOGY JOURNAL

Authors:

Agu, Edward .O.; Dennis, Samuel Toochukwu

Problem mentioned:

The paper proposes a deep learning model to detect deepfake videos, utilizing an EfficientNet B6 classifier trained on the FaceForensics++ dataset. The authors emphasize the importance of ongoing development in countering the threat of deepfake manipulation.

Algorithm Used:

The proposed deepfake detection model integrates several algorithms for accurate classification. Initially, MTCNN (Multi-Task Cascaded Convolutional Neural Networks) is employed to precisely identify and align facial features in video frames. These aligned faces are then processed through transfer learning on an EfficientNet B6 model, fine-tuned to discern between real and manipulated images. Dropout regularization mitigates overfitting, and convolution operations extract discriminative features for classification.

Results and Discussion:

The deepfake detection model, integrating MTCNN for face alignment and EfficientNet B6 for classification, yields promising results without specifying accuracy percentages. During training and testing, the model achieves effective discrimination between real and manipulated images.

Knowledge Acquired:

This study contributes to the field of deepfake detection by demonstrating the potential of CNNs and EfficientNet B6 in detecting deepfake content. The outlined methodology covers preprocessing, feature extraction, training, and evaluation. The study's results emphasize the model's effectiveness without mentioning accuracy. Future work could enhance accuracy through hyperparameter tuning and dataset expansion.

[7]. Comparative Analysis of Deepfake Image Detection Method Using Convolutional Neural Network, 2021

Journal:

ResearchGate

Authors:

Hasin Shahed Shad, Md. Mashfiq Rizvee, Nishat Tasnim Roza, S M Ahsanul Hoq

Problem mentioned:

The paper addresses the growing threat of deepfake technology, highlighting its potential for harm through the manipulation of video and audio content. The authors discuss the challenges posed by deepfakes, including their impact on privacy, security, and the spread of misinformation.

Algorithm Used:

The study compared different convolutional neural network architectures for detecting and classifying GAN-generated images. The authors found that architectures like DenseNet169, DenseNet201, and VGGFace performed well, with VGGFace achieving the best overall performance. Additionally, a custom CNN and ResNet50 also showed good performance, while DenseNet121 had lower results.

Results and Discussion:

The study developed effective deepfake detection methods using CNN architectures, notably

VGGFace, to accurately identify manipulated images. This breakthrough technology empowers individuals to discern real from fake, fostering increased vigilance.

Knowledge Acquired:

This study contributes to the field of deepfake detection by demonstrating the potential of various CNN architectures, particularly VGGFace, in detecting deepfake images with high accuracy. The authors emphasize the importance of future work, including applying CNN algorithms to video deepfake detection and exploring more efficient models to combat crime. The dataset utilized in the research is openly accessible on Kaggle, facilitating transparency and enabling further investigation in this critical field.

CHAPTER 3
REQUIREMENT SPECIFICATIONS

3.1. Hardware Requirements:

Minimum:

CPU: Intel Core i5 or equivalent
RAM: 8GB
Storage: 100GB free disk space

For optimal performance, the recommended configuration includes an Intel Core i7 processor or an equivalent, which provides faster processing and better multitasking capabilities. A higher amount of RAM, at least 16GB, ensures smoother operation and quicker data processing. Furthermore, a 500GB SSD or higher is recommended for faster data access and storage. Although not mandatory, a GPU, such as the NVIDIA GeForce GTX 1060 or an equivalent, can significantly accelerate the computations, particularly for deep learning tasks. This optional component can be beneficial for those who want to reduce processing time and improve overall performance.

3.2. Software Requirements:

Operating System:

Windows 10 or Ubuntu 18.04 LTS

The software requirements for a fake news and detection system include a specific operating system to ensure compatibility and optimal performance. The system can run on either Windows 10 or Ubuntu 18.04 LTS, providing users with flexibility in their choice of platform. By specifying these operating systems, the system can take advantage of specific features and functionalities tailored to each platform, ensuring seamless integration and

efficient operation.

3.2.1. Programming Languages:

Python 3.7 or higher

The software requirements for this project involve specifying the programming language and its version, as well as the required libraries and frameworks. The primary programming language used is Python, with a recommended version of 3.7 or higher. This version ensures compatibility with the latest features and libraries, while also providing a stable and reliable platform for development.

3.2.2 Libraries and Frameworks:

NumPy
Pandas
TensorFlow
Scikit-learn

The project requires several libraries and frameworks, including NumPy, Pandas, TensorFlow, and Scikit-learn. These libraries provide essential functionalities for data manipulation, analysis, and machine learning tasks, ensuring seamless integration and efficient operation. By specifying these software requirements, the project can leverage the power and flexibility of Python and its extensive ecosystem of libraries and frameworks.

3.3. Fake News Detection Module:

The Fake News Detection Module is designed to analyze textual content for authenticity using machine learning classifiers. The module implements text preprocessing techniques and TF-IDF vectorization to extract meaningful features from the news articles. The following are the key components of the module:

- **Dataset**: The module requires a labeled dataset of news articles, with each article marked as genuine or fake.
- **Preprocessing:** The text preprocessing techniques used in the module include tokenization, stopword removal, stemming, and lemmatization. These techniques help to clean and standardize the textual data, making it easier to extract features.
- **Vectorization:** The TF-IDF (Term Frequency-Inverse Document Frequency) vectorization technique is used to convert the textual data into numerical vectors, which can be fed into the machine learning algorithms.
- **Classification:** The module uses various machine learning algorithms for classification, including Logistic Regression, Decision Tree, Random Forest, and Gradient Boosting. These algorithms are trained on the preprocessed and vectorized data to classify news articles as genuine or fake.
- **Front-end User Interface:** The module includes a user-friendly front-end interface that allows users to input news articles for analysis and displays the classification results.

3.4. Deepfake Image Detection Module:

- **Facial Detection:** The module uses the MTCNN algorithm to detect and extract facial data from the images.
- **Feature Extraction:** The InceptionResnetV1 model is used to extract features from the facial data.
- **Explainability:** The module employs explainability techniques to enhance interpretability and provide insights into the model's decision-making process. GradCAM is used to generate heatmaps that highlight the regions of the image that contribute most to the model's prediction.
- **User Interface:** The module includes a user-friendly front-end interface that allows users to upload and analyze images. Gradio is used to create the interface.

3.5 Web Scraping for Dataset Creation Requirements:

Objective: Obtain a diverse dataset comprising both genuine and fake news articles for training and testing the Fake News Detection Module.

Scope: Utilize web scraping techniques to collect articles from various reputable news websites as well as sources known for disseminating fake news.

Selection Criteria: Identify websites representing different political affiliations, regions, and topics to ensure dataset diversity.

Ethical Considerations: Ensure compliance with ethical guidelines and legal regulations regarding web scraping, respecting website terms of service and copyright laws.

Data Validation: Implement validation mechanisms to verify the authenticity and reliability of scraped articles, checking for credibility and cross-referencing information where possible.

Metadata Extraction: Extract metadata such as publication date, author information, and article category to enrich the dataset and facilitate analysis.

Preprocessing Requirements: Preprocess the scraped text data to remove noise, standardize formatting, and enhance the quality of the dataset for machine learning purposes.

Volume: Scrape a sufficient number of articles to create a sizable dataset, considering the requirements for model training and validation.

Storage and Management: Store the scraped articles securely in a structured format, maintaining proper documentation and version control to facilitate dataset management and sharing.

Integration with Module: Integrate the scraped dataset seamlessly into the Fake News Detection Module's workflow, enabling efficient training and evaluation of machine learning models.

3.6. Non-functional Requirements:

- **Accuracy:** Advanced ML techniques for high accuracy.
- **Efficiency:** Optimized processing for large data volumes.
- **Usability:** User-friendly interfaces for easy interaction.
- **Robustness:** Designed to handle various fake news and deepfake types.
- **Scalability:** Can scale to handle increased traffic or resources.
- **Security:** Implements encryption and access control measures.

CHAPTER 4
PROBLEM DEFINITION

4.1 Problem Statement

In the current digital landscape, the rapid spread of misinformation and the emergence of sophisticated deepfake technology pose significant challenges to the credibility of online content. Misinformation can distort public opinion, create social unrest, and undermine trust in legitimate sources, while deepfakes blur the lines between reality and fabrication, raising concerns about their potential misuse in various malicious activities.

The primary objectives of this project are:

- **Develop a unified system**, TruthTracker, to detect and combat both text-based misinformation and visually deceptive deepfake content. The system should effectively identify false information and deepfakes, providing users with a reliable means of distinguishing truth from falsehood in the digital sphere.

- **Ensure scalability and adaptability of the system**. TruthTracker should be designed to handle large volumes of data and accommodate diverse user bases. It should also remain up-to-date with new forms of misinformation and deepfake techniques as they continue to evolve.

- **Create an intuitive and user-friendly interface for TruthTracker**. The system should be accessible to individuals and organizations with varying levels of technical expertise, including journalists, researchers, and ordinary citizens. Users should be able to easily navigate and utilize TruthTracker's features to verify the authenticity of news articles and images.

4.2 Relevance of the Problem

The relevance of combating misinformation and deepfake content in today's digital landscape lies in the significant risks they pose to the integrity of online information, public perception, and national security. The unchecked spread of misinformation can distort public opinion, create social unrest, and erode trust in legitimate sources of information. Meanwhile, the proliferation of hyper-realistic deepfake images and videos further exacerbates these challenges, as they blur the boundaries between reality and fabrication, raising concerns about their potential misuse in manipulating public opinion, defaming individuals, or spreading malicious narratives.

Moreover, deepfake technology has the potential to compromise biometric security systems, enabling sophisticated spear phishing attacks and manipulation of biometric security systems. This highlights the urgent need for a robust and comprehensive tool capable of accurately detecting and combating both text-based misinformation and visually deceptive deepfake content.

The development of such a tool is crucial in maintaining trust in digital content and security systems, and in ensuring the integrity of online information. By providing users with a reliable means of discerning truth from falsehood, we can empower individuals and organizations to make informed decisions, mitigate the risks associated with misinformation and deepfakes, and foster a safer, more reliable digital world.

CHAPTER 5

SYSTEM ARCHITECTURE

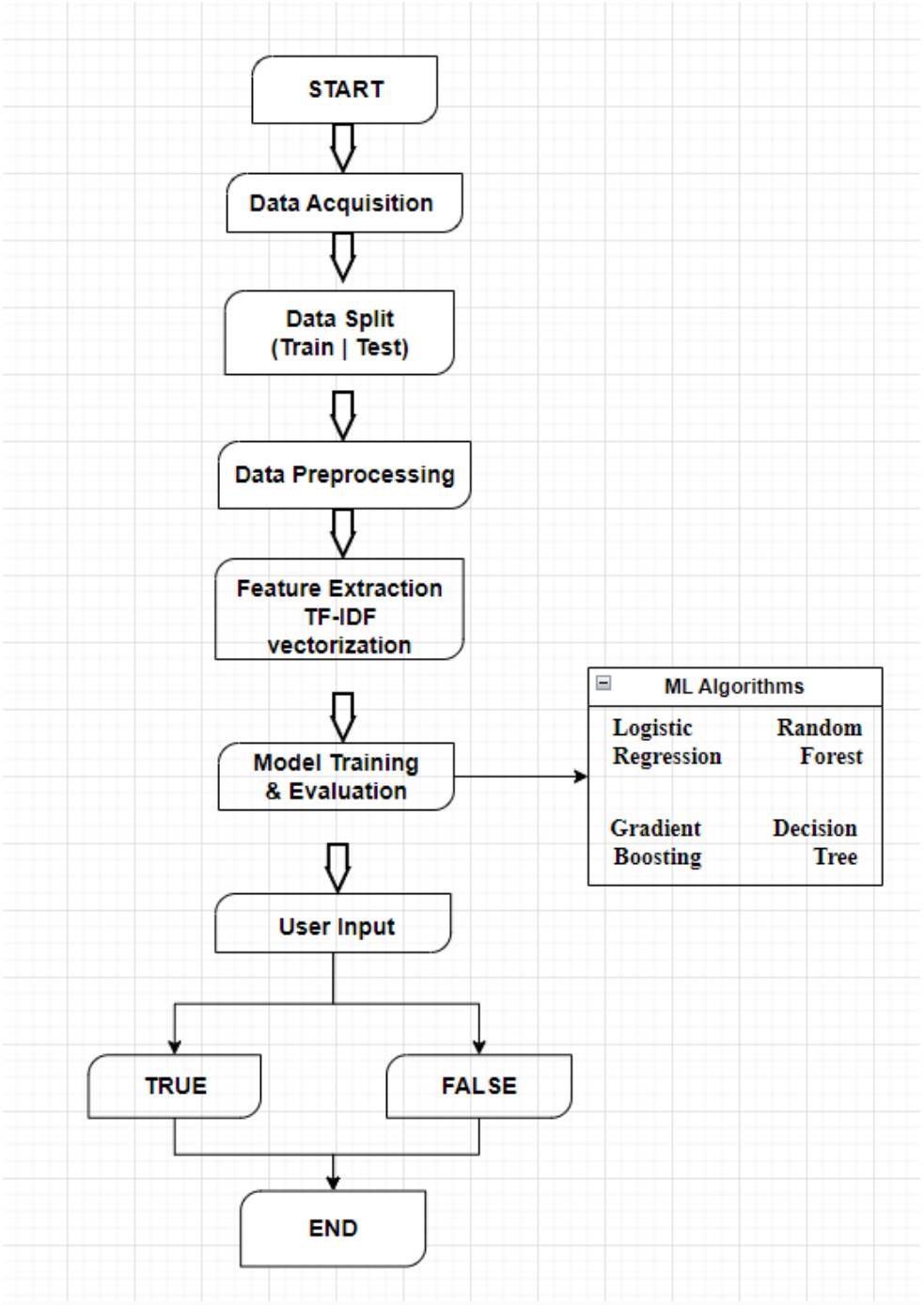

Fig 5.1. System architecture of fake news detection

5.1. System Architecture of Fake News Detection

The fake news detection module of TruthTracker is designed to accurately classify news articles as genuine or fake. The system architecture follows a series of well-defined steps, as outlined below:

Data Acquisition

The system begins by acquiring datasets containing labeled examples of true and fake news articles. These datasets typically come in CSV format and are loaded into the system for further processing.

Data Preprocessing

Text preprocessing techniques are applied to clean and prepare the news articles for classification. This includes steps such as removing punctuation, converting text to lowercase, and handling stopwords to improve the quality of the data.

Feature Extraction

TF-IDF vectorization is employed to convert the preprocessed textual data into numerical feature vectors. This process assigns importance scores to words based on their frequency in the document and across the entire corpus, capturing the distinctive characteristics of each article.

Model Training

Multiple machine learning classifiers, including Logistic Regression, Decision Tree, Random Forest, and Gradient Boosting, are trained on the labeled dataset to learn patterns and distinguish between genuine and fake news. Each classifier is trained using the TF-IDF vectors as input features.

Model Evaluation

The trained models are evaluated using various metrics such as accuracy, precision, recall, and F1-score to assess their performance in classifying news articles. This step ensures that the models are reliable and robust in identifying fake news.

Manual Testing

A function is provided to manually test the fake news detection system by inputting a news article. This allows users to interact with the system and validate its performance on real-world data.

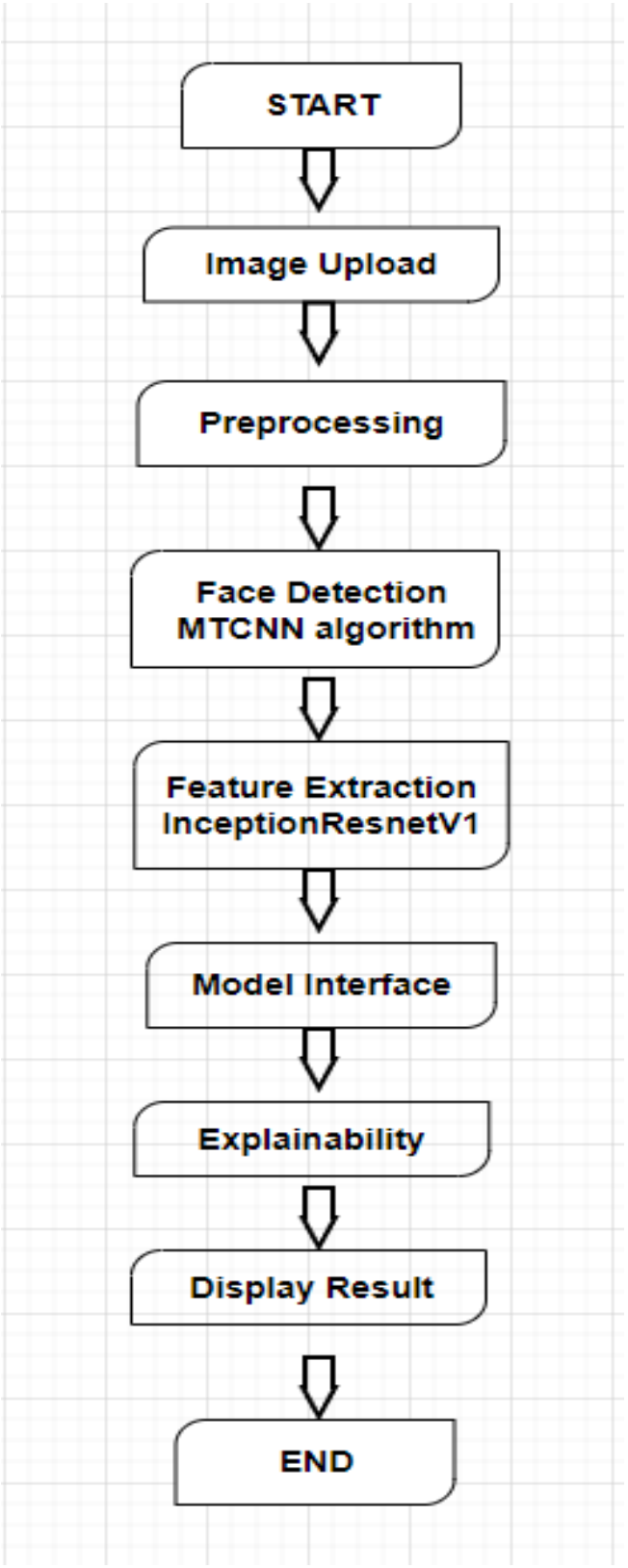

Fig 5.2.: System architecture of deepfake image detection

5.2. System architecture of deepfake image detection

The deepfake image detection module of TruthTracker is designed to identify manipulated images with high accuracy and reliability. The system architecture follows a series of well-defined steps, as outlined below:

Image Acquisition

The system begins by acquiring images that need to be analyzed for deepfake manipulations. Users can upload images through the graphical user interface (GUI) or provide image URLs for analysis.

Face Detection

Multi-task Cascaded Convolutional Networks (MTCNN) are employed for face detection in the uploaded images. This step identifies the location and size of faces present in the images, providing a crucial input for subsequent feature extraction.

Feature Extraction

The system extracts facial features from the detected faces using a pre-trained model called InceptionResnetV1. This model is specifically designed for feature extraction from facial images and provides a rich representation of facial characteristics. The extracted features serve as input for the deep learning model used in the inference stage.

Model Inference

The extracted features are fed into a deep learning model for inference. The model is trained to differentiate between genuine and manipulated facial features, enabling it to detect deepfake manipulations accurately. The model's output is a binary classification indicating whether the image is genuine or manipulated.

Explainability

Gradient-weighted Class Activation Mapping (GradCAM) is employed to generate class activation maps for model explainability. These maps highlight the regions of the image that influence the model's decision, providing insights into why certain images are classified as deepfakes. This feature enhances the system's transparency and trustworthiness, allowing users to understand the rationale behind the model's predictions.

User Interface

The deepfake detection system features a user-friendly interface developed with Gradio, allowing users to interact with the system seamlessly. Users can upload images, visualize the results, and interpret the generated class activation maps through the GUI. The interface is designed to be intuitive and accessible, ensuring that users of all technical backgrounds can effectively utilize the system.

CHAPTER 6

IMPLEMENTATION

6.1 : Fake News Detection

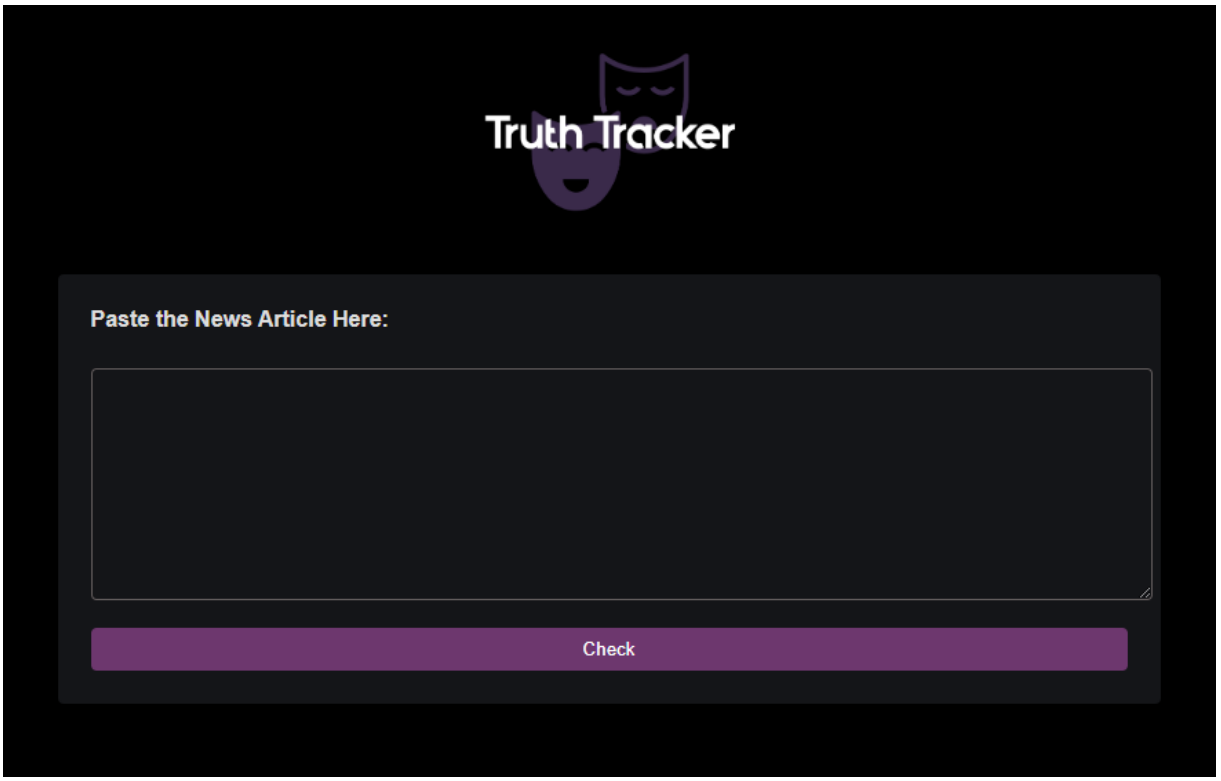

Fig 6.1: Truth Tracker News Tool Detecting True and False News

The implementation of the Fake News Detection system for TruthTracker involved several key steps, as described below:

Data Collection

Web scraping techniques were used to collect articles from various reputable news websites and sources known for disseminating fake news. ParseHub was used to scrape news articles from the Times of India archive.

The dataset consisted of two CSV files, True.csv and False.csv, containing labeled examples of genuine and fake news articles, respectively. The metadata included publication date, author information, and article category. The True.csv file contained two columns: 'title' and

'text'.

Data Preprocessing

The text data was preprocessed using several techniques:
Tokenization: The text was broken down into individual tokens or words.
Stopword Removal: Common words that do not contribute to the analysis, such as 'the', 'and', and 'a', were removed.
Stemming and Lemmatization: Words were reduced to their base or root form, allowing for more effective analysis.
Vectorization: TF-IDF vectorization was used to convert the text data into numerical vectors. This process assigns importance scores to words based on their frequency in the document and across the entire corpus, capturing the distinctive characteristics of each article.

Model Training

Multiple machine learning algorithms were used, including Logistic Regression, Decision Tree, Random Forest, and Gradient Boosting.
The training process involved splitting the data into training, validation, and test sets, and hyperparameter tuning to optimize the performance of each model. After training, the Random Forest algorithm was found to be the best fit for the model. The trained model was saved for use in the application using the Python language and the Flask framework.

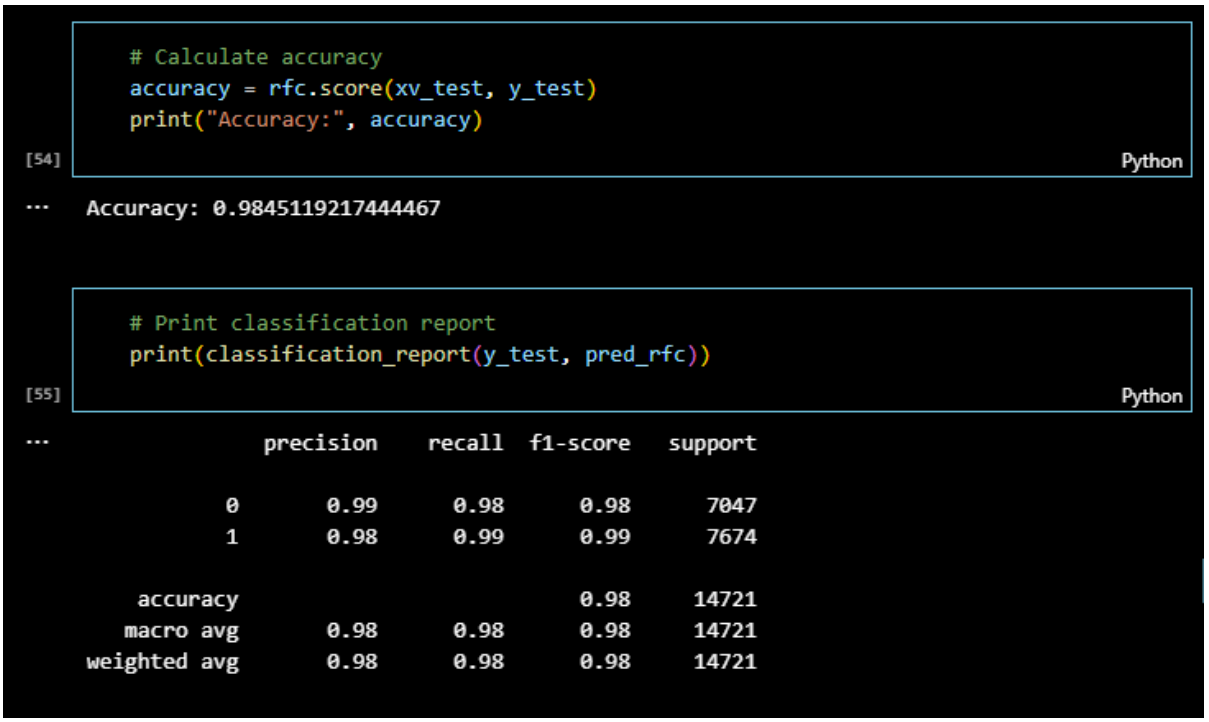

Fig 6.2: Accuracy and Classification Report for Random Forest Classifier

Model Evaluation

The performance of the models was evaluated using several metrics, including accuracy, precision, recall, and F1-score.

The Random Forest model performed the best, achieving high scores in all evaluation metrics.

User Interface

The user interface was designed to allow users to input news articles and view classification results.

The front-end interface was developed using HTML and CSS, with a static template design.

6.1.1 Results and Analysis and Integration of IFCN Fact-Checking Organizations for Enhanced Reliability and Transparency

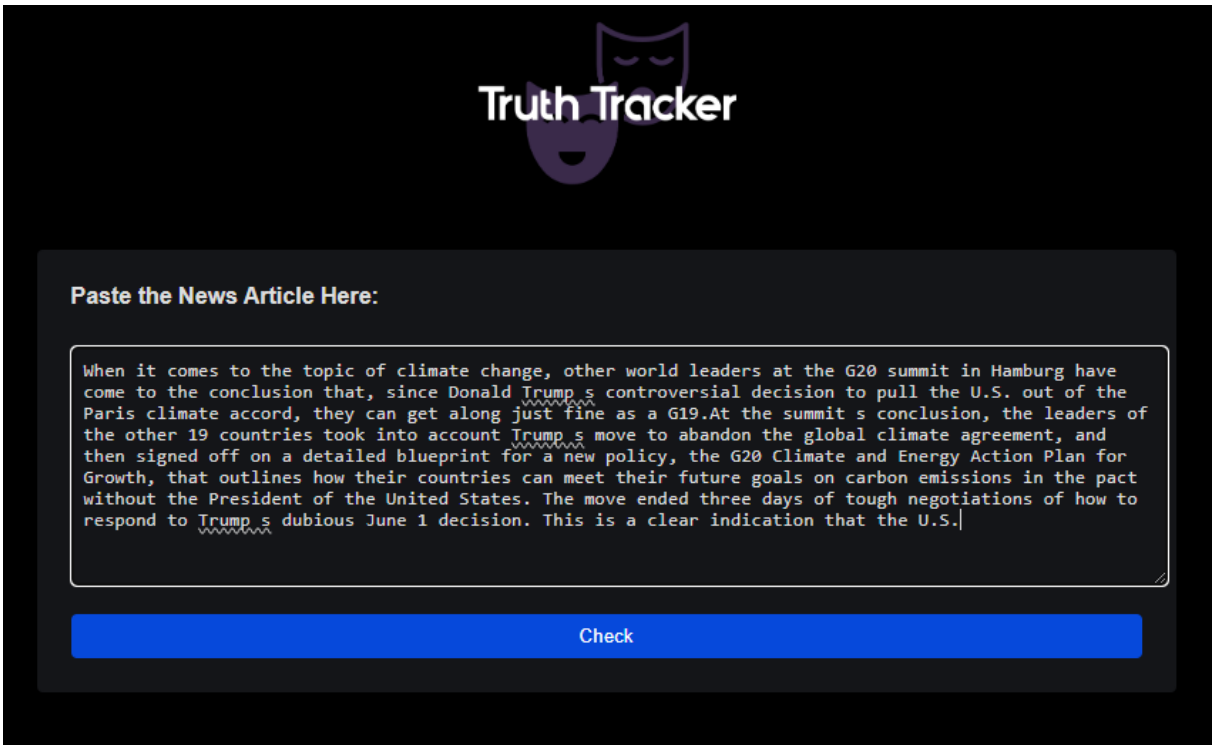

Fig 6.3: Truth Tracker News Tool Detecting False News

Fig 6.4: Output detected as fake (False)

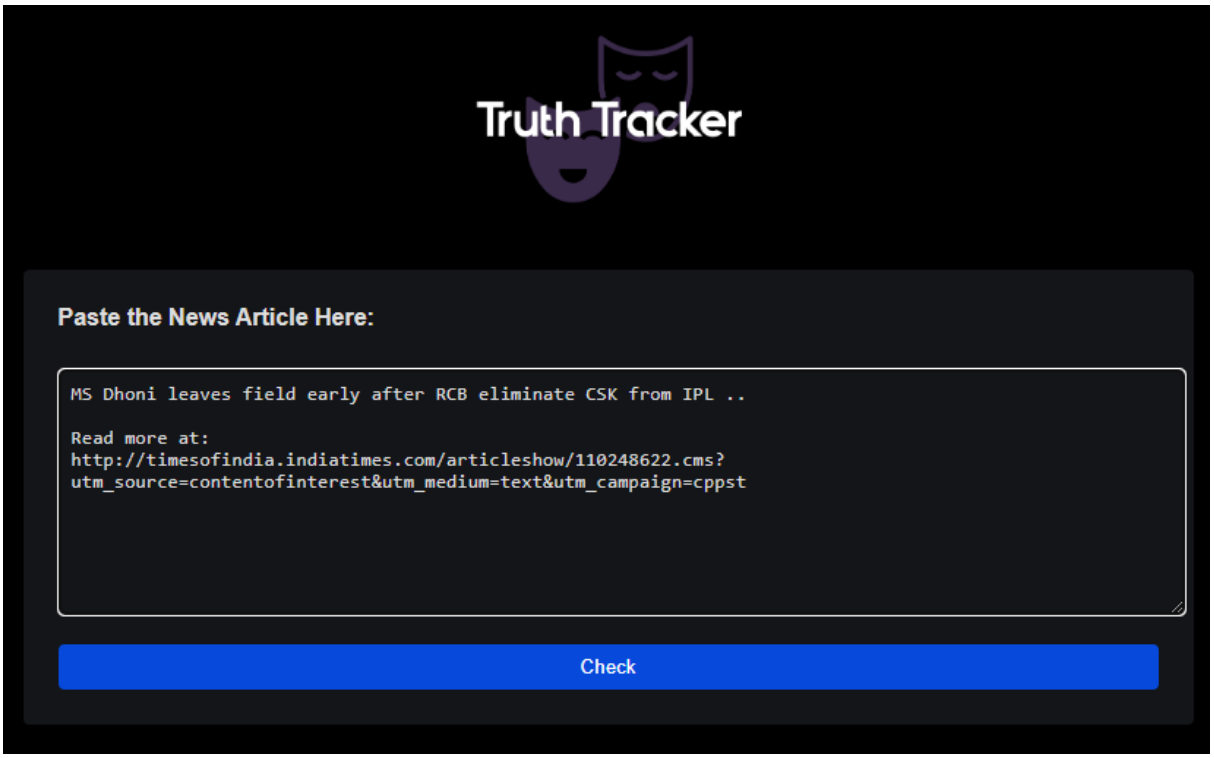

Fig 6.5: Truth Tracker News Tool Detecting True News

Fig 6.6: Output detected genuine(True)

Fig 6.7: Integrating IFCN fact checking organizations

To further enhance the reliability and transparency of the Fake News Detection system for TruthTracker, we have integrated the International Fact-Checking Network (IFCN) fact checking organizations into the front-end user interface.

The following IFCN fact checking organizations have been added to the front-end, with direct links to their websites and WhatsApp channels:

- Boom
- Fact Crescendo
- Factly
- DigitEYE

- India Today Fact Check
- NewsChecker
- NewsMobile
- The Quint

These organizations are internationally verified and provide additional resources for users to verify the authenticity of news articles. By integrating these organizations into the front-end user interface, users can easily access their websites and WhatsApp channels to further validate the results of the Fake News Detection system.

This integration provides an additional layer of transparency and accountability, ensuring that users can trust the results of the system and have access to multiple sources of information to make informed decisions. The use of IFCN fact checking organizations also enhances the credibility of the system, providing users with confidence in its ability to accurately detect fake news.

6.2 Deepfake Image Detection with Results and Analysis

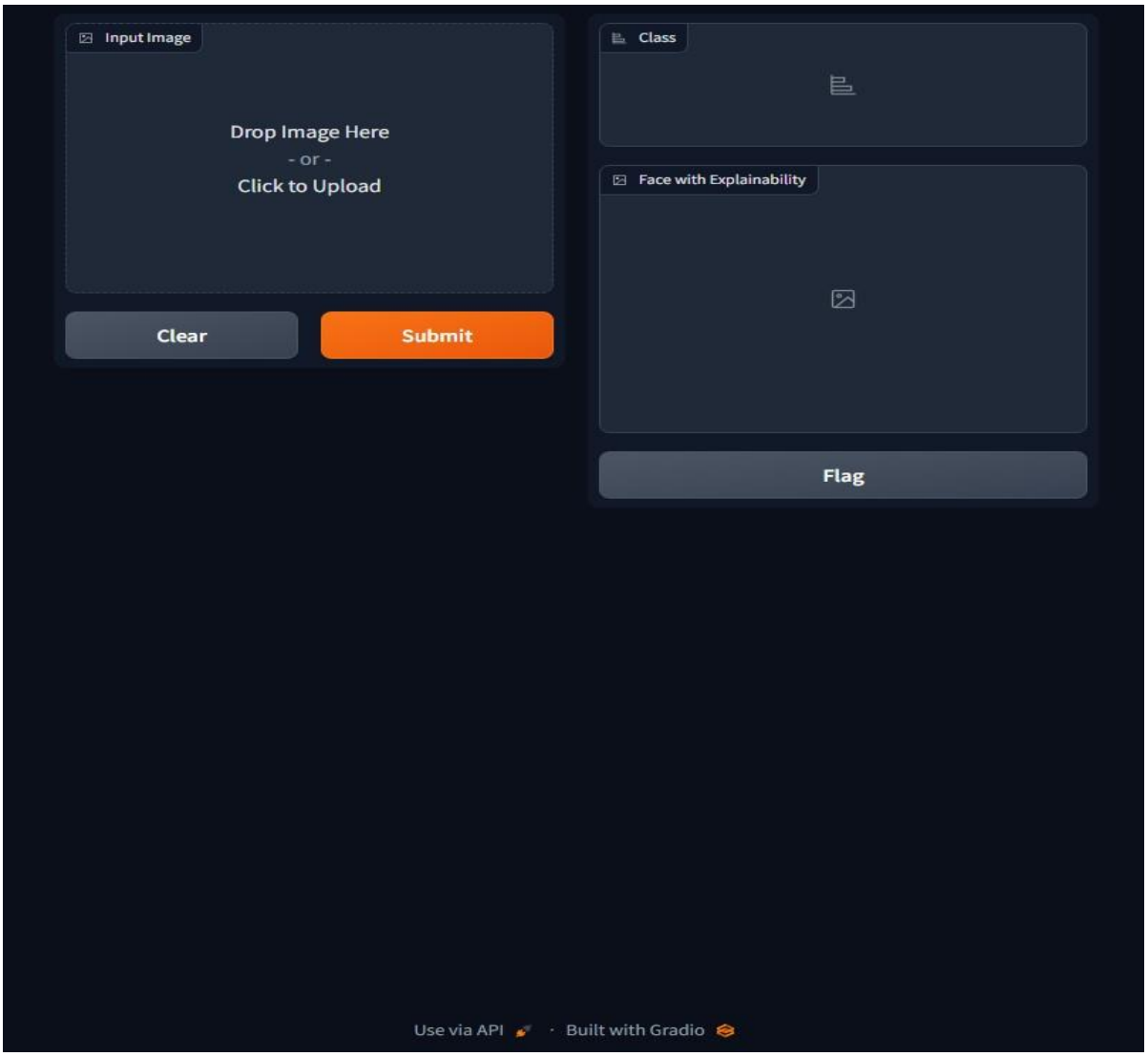

Fig 6.8: The Deepfake Detection Tool

The implementation of the Deepfake Image Detection system for TruthTracker involved several key steps, as described below:

Image Acquisition

Images were acquired through user upload, allowing users to upload images for analysis.
Specific datasets used for training and testing included FaceForensics++ and VGGFace2 dataset.

Face Detection

The Multi-Task Cascaded Convolutional Networks (MTCNN) algorithm was used for detecting faces in images.

Feature Extraction

InceptionResnetV1 was used to extract features from the detected faces.
Model Inference

A deep learning model was used to classify images as deepfake or genuine.
The training process involved data preparation, model architecture design, and hyperparameter tuning.

Explainability

Gradient-weighted Class Activation Mapping (GradCAM) was used to generate heatmaps, highlighting the regions of the image that contribute most to the model's decision.
These heatmaps help in understanding the model's decision-making process, providing transparency and interpretability.

User Interface

The user interface was designed to allow users to upload and analyze images, and view results.
The front-end interface was developed using Gradio, a Python library for creating user interfaces for machine learning models.

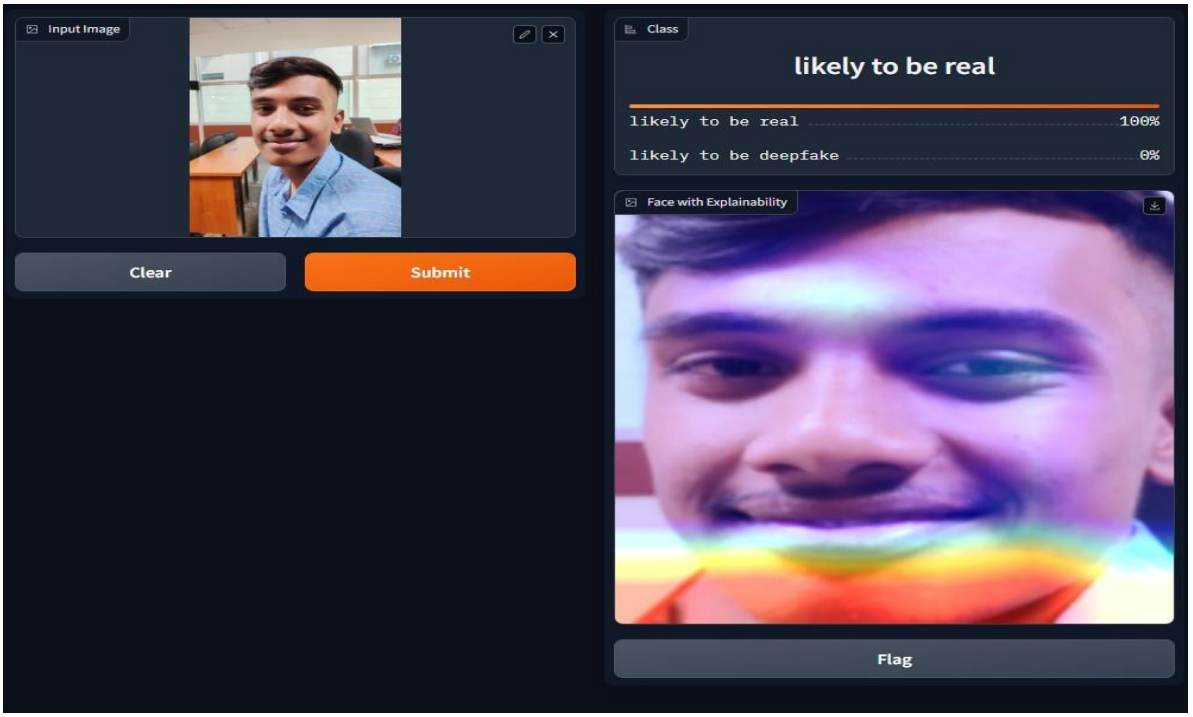

Fig 6.9 : Detecting Real image

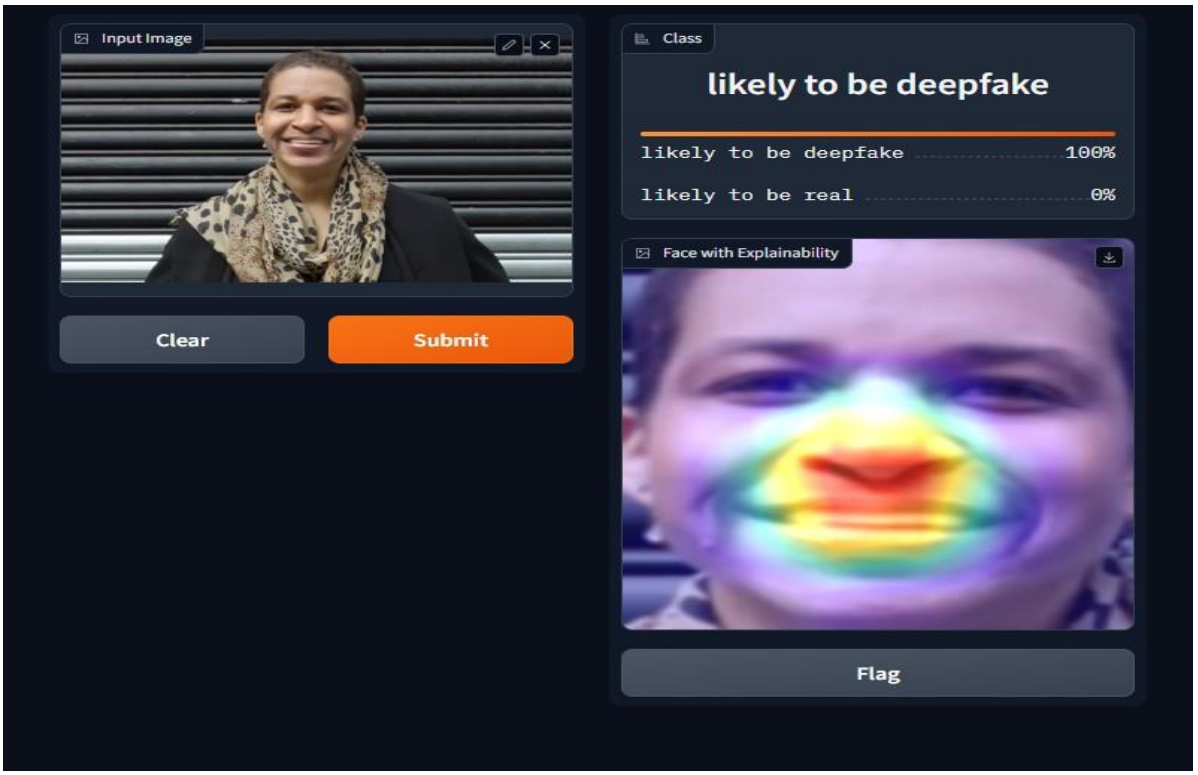

Fig 6.10 : Detecting Deepfake image

CHAPTER 7
CONCLUSION AND FUTURE WORK

In this project, we have developed TruthTracker, a comprehensive tool for detecting and combatting both text-based misinformation and visually deceptive deepfake content. The system architecture for both modules, deepfake image detection and fake news detection, has been presented, detailing the various stages from data acquisition to model inference and user interface.

For the deepfake image detection module, we have implemented a sophisticated architecture using Multi-Task Cascaded Convolutional Networks (MTCNN) for face detection, InceptionResnetV1 for feature extraction, and a deep learning model for inference. Gradient-weighted Class Activation Mapping (GradCAM) is employed to generate class activation maps for model explainability. The user-friendly interface, developed with Gradio, allows users to interact with the system seamlessly.

The fake news detection module of TruthTracker follows a systematic architecture to accurately classify news articles as genuine or fake. We have used web scraping techniques to collect articles from various reputable news websites and sources known for disseminating fake news. After data preprocessing, TF-IDF vectorization is employed to convert the preprocessed textual data into numerical feature vectors. Multiple machine learning classifiers, including Logistic Regression, Decision Tree, Random Forest, and Gradient Boosting, are trained on the labeled dataset to learn patterns and distinguish between genuine and fake news.

In addition to the deepfake image detection and fake news detection modules, we have

integrated the International Fact-Checking Network (IFCN) fact-checking organizations into our front-end for further verification. Users can access these organizations' websites and WhatsApp channels directly from our platform for additional fact-checking.

For future work, we aim to expand the deepfake image detection module to include video deepfakes and improve the model's performance by incorporating more sophisticated feature extraction techniques and deep learning architectures. In the fake news detection module, we plan to explore the use of advanced natural language processing techniques and deep learning models to improve the accuracy of news classification.

In summary, TruthTracker provides a robust and reliable solution for detecting deepfake images and fake news, contributing to a more secure and trustworthy digital environment. By continuously improving and expanding the system's capabilities, we aim to stay ahead of the ever-evolving challenges posed by misinformation and deepfake technology.

REFERENCES

[1] Johnson Adeleke Adeyiga, Philip Gbounmi Toriola, Temitope Elizabeth Abioye, Adebisi Esther Oluwatosin, Oluwasefunmi 'Tale Arogundade. “Fake News Detection Using a Logistic Regression Model and Natural Language Processing Techniques”. 2023, DOI: https://doi.org/10.21203/rs.3.rs-3156168/v1

[2] Nidhi Singh Kushwaha, Pawan Singh. “Fake News Detection using Machine Learning: A Comprehensive Analysis” 2022. DOI: https://doi.org/10.54060/JMSS/002.01.001

[3]Reham Jehad Al-Shammari, Suhad A. Yousif. “Fake News Classification Using Random Forest and Decision Tree (J48)” ResearchGate, 2020. DOI: 10.22401/ANJS.23.4.09

[4] Anjali Jain, Avinash Shakya, Harsh Khatter, Amit Kumar Gupta, “A smart System for Fake News Detection Using Machine Learning,” ResearchGate, 2019. DOI:10.1109/ICICT46931.2019.8977659

[5] Dr. P.Sruthi, Dr.T.Bhaskar, Bommagalla Ankitha, Duggirala Mercy Sunada, Yellanur Bhargavi, Yerra Manasa. "Deepfake Detection", 2024, Published Paper ID: JETIR2404454

[6] Agu, Edward .O.; Dennis, Samuel Toochukwu, “ Deepfake Detection using Convolutional Neural Networks ”, 2023, DOI:10.1109/ICoAC59537.2023.10250107

[7] Hasin Shahed Shad, Md. Mashfiq Rizvee, Nishat Tasnim Roza, S M Ahsanul Hoq, “Comparative Analysis of Deepfake Image Detection Method Using Convolutional Neural Network”, 2021, DOI:10.1155/2021/3111676

TruthTracker-Combatting Misinformation and Deepfakes Online

ORIGINALITY REPORT

17%	14%	13%	%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	www.ijraset.com Internet Source	3%
2	dl.ifip.org Internet Source	2%
3	www.researchsquare.com Internet Source	1%
4	dergipark.org.tr Internet Source	1%
5	de.overleaf.com Internet Source	1%
6	www.ijert.org Internet Source	1%
7	speedypaper.x10.mx Internet Source	1%
8	fastercapital.com Internet Source	<1%
9	github.com Internet Source	<1%

10	Esma Aïmeur, Sabrine Amri, Gilles Brassard. "Fake news, disinformation and misinformation in social media: a review", Social Network Analysis and Mining, 2023 Publication	<1 %
11	Sofia I. Hernandez Torres, Austin Ruiz, Lawrence Holland, Ryan Ortiz, Eric J. Snider. "Evaluation of Deep Learning Model Architectures for Point-of-Care Ultrasound Diagnostics", Bioengineering, 2024 Publication	<1 %
12	"Mobile Radio Communications and 5G Networks", Springer Science and Business Media LLC, 2024 Publication	<1 %
13	Deepak P, Tanmoy Chakraborty, Cheng Long, Santhosh Kumar G. "Data Science for Fake News", Springer Science and Business Media LLC, 2021 Publication	<1 %
14	globaljournals.org Internet Source	<1 %
15	ijsrst.com Internet Source	<1 %
16	www.bio-conferences.org Internet Source	<1 %

17	www.coursehero.com Internet Source	<1 %
18	www.researchgate.net Internet Source	<1 %
19	medium.com Internet Source	<1 %
20	"Proceedings of the 2nd International Conference on Big Data, IoT and Machine Learning", Springer Science and Business Media LLC, 2024 Publication	<1 %
21	Neel Save, Omkar Thopate, Vaibhav Waghe, Manisha Bansode, Najib Ghatte. "Artificial Intelligence based Fake News Classification System", 2021 12th International Conference on Computing Communication and Networking Technologies (ICCCNT), 2021 Publication	<1 %
22	www.irjmets.com Internet Source	<1 %
23	www.ncbi.nlm.nih.gov Internet Source	<1 %
24	www.grin.com Internet Source	<1 %
25	ijsrcseit.com Internet Source	<1 %

26	mail.anjs.edu.iq Internet Source	<1 %
27	pdfcoffee.com Internet Source	<1 %
28	www.mdpi.com Internet Source	<1 %
29	César González Fernández, Isaac Martín De Diego, Alberto Fernández-Isabel, Juan Fernando Jiménez Viseu et al. "Detecting Low-Credibility Medical Websites Through Semi-Supervised Learning Techniques", IEEE Access, 2023 Publication	<1 %
30	arxiv.org Internet Source	<1 %
31	Mahabuba Akhter, Syed Md. Minhaz Hossain, Rizma Sijana Nigar, Srabanti Paul et al. "COVID-19 Fake News Detection using Deep Learning Model", Annals of Data Science, 2024 Publication	<1 %
32	Najwan Thair Ali, Karrar Falih Hassan, Muataz Najim Abdullah, Zainab Salam Al-Hchimy. "The Application of Random Forest to the Classification of Fake News", BIO Web of Conferences, 2024 Publication	<1 %

33 en.wikipedia.org
Internet Source

<1%

Exclude quotes	On	Exclude matches	< 14 words
Exclude bibliography	On		