System for AI-Based Analysis of Social Media Video Content and Public Sentiment

Name: R.M. Nisitha Nethsilu

London Met ID No: 24064701

London Metropolitan University

Declaration

Module: CS6P05ES **Deadline: 11/02/2025**

Module Leader: Prof. Ruvan Abeysekara **Student ID: E195991**

PLAGIARISM

You are reminded that there exist regulations concerning plagiarism. Extracts from these regulations are printed below. Please sign below to say that you have read and understand these extracts:

(Signature:) Nisitha Date: 11/08/2025

This header sheet should be attached to the work you submit. No work will be accepted without it.

Extracts from University *Regulations* on Cheating, Plagiarism and Collusion

Section 2.3: "The following broad types of offence can be identified and are provided as indicative examples…

(i) Cheating: including taking unauthorised material into an examination; consulting unauthorised material outside the examination hall during the examination; obtaining an unseen examination paper in advance of the examination; copying from another examinee; using an unauthorised calculator during the examination or storing unauthorised material in the memory of a programmable calculator which is taken into the examination; copying coursework.

(ii) Falsifying data in experimental results.

(iii) Personation, where a substitute takes an examination or test on behalf of the candidate. Both candidate and substitute may be guilty of an offence under these Regulations.

(iv) Bribery or attempted bribery of a person thought to have some influence on the candidate's assessment.

(v) Collusion to present joint work as the work solely of one individual.

(vi) Plagiarism, where the work or ideas of another are presented as the candidate's own.

(vii) Other conduct calculated to secure an advantage on assessment.

(viii) Assisting in any of the above.

Some notes on what this means for students:

1. Copying another student's work is an offence, whether from a copy on paper or from a computer file, and in whatever form the intellectual property being copied takes, including text, mathematical notation and computer programs.

2. Taking extracts from published sources *without attribution* is an offence. To quote ideas, sometimes using extracts, is generally to be encouraged. Quoting ideas is achieved by stating an author's argument and attributing it, perhaps by quoting, immediately in the text, his or her name and year of publication, e.g. "$e = mc^2$ (Einstein 1905)". A *references* section at the end of your work should then list all such references in alphabetical order of authors' surnames. (There are variations on this referencing system which your tutors may prefer you to use.) If you wish to quote a paragraph or so from published work then indent the quotation on both left and right margins, using an italic font where practicable, and introduce the quotation with an attribution.

Dedication

I dedicate this project to my dear parents, whose unwavering support and inspiration have been my biggest motivation during my academic path. I also dedicate this to my mentors and friends who supported me in difficult moments, motivating me to strive for excellence and finish this project successfully.

Acknowledgements

I wish to convey my heartfelt thanks to my supervisor, Mr. Tharaka Karunarathna, for his insightful guidance, constructive feedback, and ongoing support during the progression of this project. His knowledge and perspectives significantly aided the success of my project. I would like to extend my gratitude to my instructors, peers, and friends for their assistance and encouragement. Ultimately, I am truly grateful to my parents for their consistent love, support, and motivation, which have been the cornerstone of my successes.

Abstract

This initiative aims at creating a sophisticated Fake News Detection System that employs Natural Language Processing (NLP) and Machine Learning (ML) methods to recognize and validate the credibility of news articles found online. In the current digital age, the swift dissemination of false information has emerged as a major obstacle, impacting public perception and societal cohesion. The primary issue tackled in this project is the absence of dependable automated tools that can effectively differentiate between authentic and deceptive news articles with a high level of accuracy.

The suggested approach integrates machine learning models with an AI verification component to establish a two-tier validation system. The system initially evaluates the news content through trained ML classifiers like Logistic Regression, Random Forest, Decision Tree, and Gradient Boosting, which determine if a specific statement is authentic or fraudulent. Subsequently, an integrated AI module fueled by OpenAI GPT conducts a second verification by cross examining the context and reasoning of the content, boosting the prediction's reliability.

Users engage with the system via a professional Gradio dashboard interface, allowing them to enter a news article or headline and receive immediate results. The results consist of model forecasts, AI validation, and a created performance summary report. The procedure includes text preprocessing, feature extraction via TF-IDF Vectorization, training the model, testing, and ongoing assessment using accuracy metrics and classification reports.

Through implementation, challenges like API connectivity problems and variations in model accuracy were faced and addressed through ongoing testing and enhancements in integration. The completed system showcases a successful integration of data-oriented analysis and AI-driven reasoning, greatly enhancing detection precision. In summary, this initiative effectively creates a scalable and smart system that facilitates real time identification of fake news, aiding the battle against misinformation in online media.

Table of Contents

Contents

1 Chapter 1 Introduction 1

1.1 Goals 1

1.2 Motivation 2

1.3 Method 2

1.4 Overview 3

2 Chapter 2 Background and Problem Statement 4

2.1 Introduction 4

2.2 Literature Review 4

2.3 Problem Statement 5

3 Chapter 3 project management 7

3.1 Approach 7

3.2 Initial Project Plan 7

3.3 Problems and Changes to the Plan 8

3.4 Final Project Record 9

4 Chapter 4 feasibility study 10

4.1 Time feasibility 10

4.2 Cost feasibility 11

4.3 Scope feasibility 12

4.4 Technical feasibility. 13

4.5 Economic feasibility 13

5 Chapter 5 design 15

5.1 Introduction to your choice of proposed network system 15

5.2 Hardware and software requirements 17

5.3 Evaluating solutions 17

6 Chapter 6 implementation 19

7 Chapter 7 testing and verification 24

8 Chapter 8 EVALUATION and CONCLUSION 27

9 References 29

10 References 29

11 Appendices.. 30

Table of figures

Figure 1System Architecture Diagram 17
Figure 2Comparison Chart of ML Model Accuracies 18
Figure 3code execution 19
Figure 4code execution 20
Figure 5Model Training and Testing 21
Figure 6Model Training and Testing 21
Figure 7Model Training and Testing 21
Figure 8Model Training and Testing 22
Figure 9Gradio Interface for Model and AI Verification 22
Figure 10Fake News Detection Dashboard 23
Figure 11Interim Report 31
Figure 12Interim Report 32
Figure 16Approvel 34

Table of tables

Table 1Abbreviations ix
Table 2Goals 2
Table 3Methods 3
Table 4Literature Review 5
Table 5Initial Project Plan 8
Table 6Changes and Adjustments 9
Table 7Final Project Plan 10
Table 8Time feasibility 11
Table 9Cost feasibility 11
Table 10Scope feasibility 13
Table 11Benefit Analysis Data 14
Table 12Hardware and Software Requirements 17
Table 13Key Evaluation Aspects 18
Table 14Test Environment 24
Table 15Test Cases 25
Table 16Summery 25

Abbreviations

Abbreviation	Full Form	Description / Usage
AI	Artificial Intelligence	Simulation of human intelligence in machines.
ML	Machine Learning	A subset of AI that enables systems to learn from data.
API	Application Programming Interface	A set of protocols that allow software applications to communicate.
UI	User Interface	The visual elements that users interact with in a system.
CSV	Comma Separated Values	A file format for storing tabular data in plain text.
EDA	Exploratory Data Analysis	The process of analyzing data sets to summarize key characteristics.
LR	Logistic Regression	A statistical model used for binary classification tasks.
RFC	Random Forest Classifier	An ensemble learning method for classification and regression.
GBC	Gradient Boosting Classifier	A machine learning technique for improving model accuracy.
IoT	Internet of Things	Network of interconnected devices that collect and exchange data.
NLP	Natural Language Processing	AI field focused on the interaction between computers and human language.
API Key	Application Programming Interface Key	A unique identifier used to authenticate API access.

Table 1Abbreviations

1 Chapter 1 Introduction

The growing reliance on digital news outlets and social media has led to an excessive amount of information available to the public. Regrettably, this easy access has resulted in the swift dissemination of fake news false or deceptive information depicted as factual news. Misinformation has generated uncertainty, swayed public perceptions, and occasionally undermined social and political stability. This initiative presents an AI-driven Fake News Detection System that combines machine learning algorithms and artificial intelligence (AI) validation to effectively detect false or misleading news reports. This chapter offers a summary of the objectives, inspiration, approaches, and technical underpinnings of the project.

Project Link = https://colab.research.google.com/drive/1Ek8cmozS-NZcpGhiZr984agU6ifOg8OV?usp=drive_link

1.1 Goals

The main objective of this project is to create and implement an intelligent system for detecting fake news that can evaluate a headline or news article and categorize it as True or False.
Goals are:

Objective	Description
1. Develop ML Models	Build and train multiple machine learning models (Logistic Regression, Random Forest, Decision Tree, and Gradient Boosting) to detect fake news.
2. Integrate AI Verification	Add a secondary AI verification stage using OpenAI GPT to cross check and validate predictions.
3. Create an Interactive Dashboard	Develop a professional Gradio-based dashboard for user interaction, enabling easy testing and report generation.
4. Evaluate Model Accuracy	Assess model performance using precision, recall, and F1-score metrics to ensure reliability.
5. Generate Reports	Automatically produce summary and

	performance reports after each analysis to help improve future model iterations.

Table 2Goals

1.2 Motivation

False information has emerged as a worldwide issue that endangers the trustworthiness of internet content. In nations where social media is widely used, like Sri Lanka, the quick dissemination of false information can confuse the public and influence choices. Manual verification takes a lot of time and is frequently unrealistic because of the high amount of information exchanged online every day.

As a result, an automated and intelligent system is required to quickly identify and confirm the authenticity of news. This initiative seeks to address this demand by integrating the capabilities of machine learning and AI-driven reasoning to develop a reliable and scalable approach that improves digital literacy and effectively counters misinformation

1.3 Method

The project was carried out using a systematic, data-driven methodology as shown below:

Phase	Description
Data Collection	Real and fake news datasets were collected from trusted open-source repositories.
Preprocessing	Text cleaning was performed (removal of punctuation, numbers, URLs, and HTML tags) to prepare data for training.
Feature Extraction	Implemented TF-IDF Vectorization to convert text into numerical features suitable for machine learning models.
Model Training	Trained and compared multiple classifiers including Logistic Regression, Random Forest, Decision Tree, and Gradient Boosting.
AI Integration	Integrated the OpenAI GPT API to cross-verify model results and generate reasoning.
Dashboard Development	Developed a Gradio-based web interface to allow users to input news and view model + AI

	outputs.
Evaluation and Reporting	Generated accuracy reports, classification metrics, and AI-driven summary reports for user evaluation.

Table 3Methods

1.4 Overview

This initiative exists at the convergence of Artificial Intelligence (AI), Machine Learning (ML), and Natural Language Processing (NLP). It utilizes computational linguistics to examine text-based information and applies ML algorithms to identify misleading linguistic trends. The incorporation of GPT-focused AI offers context-sensitive evaluation, replicating human reasoning to validate the model's forecasts.

The front-end dashboard of the system, created with Gradio, provides an engaging and easy to navigate interface for real time testing and result visualization. This project signifies a contemporary method for guaranteeing the integrity of news and the reliability of digital information by merging data science with AI reasoning.

2 Chapter 2 Background and Problem Statement

The emergence of digital communication platforms has entirely changed the way individuals obtain and distribute information. News today circulates through social media, online publications, and blogs at an astonishing speed. Nonetheless, this change has created a significant issue — the swift spread of false or deceptive information. Misinformation has emerged as a significant problem impacting journalism, politics, health, and society's general confidence in the media. This chapter examines the context of fake news detection, analyzes current literature and systems, and highlights the research issue tackled by this project

2.1 Introduction

This project focuses on the domain of Natural Language Processing (NLP) and Machine Learning (ML), which are two fundamental subfields of Artificial Intelligence employed to analyze and comprehend human language. False news pertains to intentionally created or altered information designed to deceive audiences. Automatically identifying fake news entails comprehending linguistic patterns, semantics, and the credibility of sources.

In recent times, social media sites like Facebook, Twitter (X), and WhatsApp have hastened the dissemination of false information. Manual fact-checking cannot keep pace with the daily volume of data produced. Therefore, there is a necessity for automated systems that can detect fake news by analyzing news content in real-time through computational techniques.
The system created in this project employs a hybrid method, a blend of machine learning classification models and AI-driven reasoning (integration of ChatGPT) to determine if the provided news is authentic or misleading

2.2 Literature Review

The identification of false information has been a prominent field of study, with numerous methods suggested in the last ten years. The following review outlines significant trends and methods instead of organizing them by date or company:

Approach	Description	Strengths	Limitations
Content-Based Analysis	Examines linguistic and stylistic features (e.g., sensational words, emotional tone).	Works well with textual data; interpretable.	May fail if fake and real news share similar language.
Machine Learning Models	Uses supervised algorithms such as Logistic Regression, Decision Trees, and Random Forests trained on labeled data.	Provides quantifiable accuracy; efficient for large datasets.	Requires extensive, high-quality training data.

Deep Learning Methods	CNNs and LSTMs capture contextual relationships between words.	High accuracy in text classification.	Computationally expensive and requires massive data.
Hybrid and Ensemble Methods	Combines multiple models to reduce individual weaknesses.	Improved precision and recall rates.	Model complexity increases.
AI-Assisted Verification	Integrates large language models (e.g., GPT) for reasoning and cross-validation.	Adds contextual understanding and real-world awareness.	Dependent on external APIs and may vary in reliability.

Table 4Literature Review

Recent research indicates that merging conventional ML algorithms with smart reasoning models yields enhanced detection accuracy. The latest systems focus on not just classification precision but also clarity and openness enabling users to grasp the reasons a specific article is marked as false or genuine.

This initiative expands on these developments through the introduction of a two-phase verification system:

- Classification based on models utilizing trained machine learning algorithms.
- AI-driven cross checking utilizes GPT logic to validate or dispute the model's results.

2.3 Problem Statement

Even with available studies and resources, numerous fake news detection systems struggle with issues related to accuracy, reliability, and understanding context. Conventional ML classifiers can detect deceptive patterns through statistics but frequently do not grasp the significance or intention behind the content. Similarly, AI models can reason effectively but may not have direct access to labeled datasets for their training.

The issue tackled in this project is the creation of an intelligent system for detecting fake news that combines machine learning-driven text classification with AI-enhanced verification to enhance the accuracy, reliability, and interpretability of assessing news authenticity.

The system should possess the ability to:

- Receiving user submissions in the shape of news articles or headlines.
- Analyzing and categorizing the input through trained machine learning models.
- Confirming the outcome via AI (ChatGPT) logic.

- Showcasing both outcomes on a professional, intuitive dashboard.
- Creating a report on performance and precision for continuous assessment.

3 Chapter 3 project management

Efficient project management is essential for the successful execution of any data science or AI-driven system. This chapter details the strategy employed for project management, the original and final project plans, along with the modifications implemented throughout the development process. It covers the organization, oversight, and adjustment of tasks to address challenges while ensuring the project remains consistent with its goals and timelines.

3.1 Approach

The project utilized an iterative and agile method since data science initiatives frequently require experimentation, ongoing testing, and enhancement. Every phase from gathering data to assessing the model needed adaptability to modify based on the results.

The Agile methodology facilitated consistent feedback, particularly during the testing of machine learning models and the incorporation of AI verification. The initiative was segmented into stages (Planning, Development, Testing, Deployment, and Evaluation) featuring brief milestones.

This method guaranteed:

- Quicker identification and correction of mistakes.
- Simple integration of new methods or frameworks.
- Ongoing enhancement via repeated testing and validation driven by AI.
- Utilizing tools like Google Colab for coding, Gradio for UI creation, and OpenAI API for AI incorporation facilitated seamless advancement and teamwork during the project.

3.2 Initial Project Plan

At the project's outset, a carefully organized plan was created to steer the development of the "AI-Driven System for Social Media Video Content and Public Sentiment Analysis."
The strategy was created utilizing a timeline-oriented Gantt chart, detailing all key activities from literature review to the concluding presentation. Every task included well-defined beginning and ending dates, timelines, and outcomes.

The project utilized a step-by-step method starting with data gathering and analysis, then moving to machine learning creation, assessment, and concluding with reporting. The schedule guaranteed that technical and documentation activities are advanced simultaneously to prevent last minute setbacks.

Task	Description	Planned Duration	Start Date	End Date
Project Proposal & Research	Understanding problems and related work	1 week	Week 1	Week 2
Dataset Collection & Cleaning	Gathering and preprocessing	2 weeks	Week 2	Week 4

	fake/real news datasets			
Model Development	Training ML models (LR, RF, GBC)	2 weeks	Week 4	Week 6
Model Testing & Evaluation	Testing accuracy, recall, and precision	1 week	Week 6	Week 7
AI Integration	Connecting OpenAI API for verification	2 weeks	Week 7	Week 9
Dashboard Design	Building interactive Gradio interface	1 week	Week 9	Week 10
Performance Reporting	Generating AI-based summary reports	1 week	Week 10	Week 11
Documentation & Final Report	Writing reports and preparing presentation	2 weeks	Week 11	Week 13

Table 5Initial Project Plan

3.3 Problems and Changes to the Plan

Though the original plan was well organized, numerous unexpected obstacles arose throughout the project, resulting in required modifications. These problems primarily stemmed from API connections, data quantity, and time management limitations.

Google Gemini API experienced several authorization errors and delays in model responses, leading to a temporary pause in AI verification progress. To ensure advancement, the project transitioned to utilizing the OpenAI GPT 4 API, known for being more dependable and quicker.

Moreover, gathering sentiment and video data from social media required more time than anticipated because of platform rate limits and the necessity for manual checks. As a result, data preprocessing and exploration analysis were prolonged by several additional days.

To achieve balance, certain documentation activities, like finalizing proposals and interim reporting, were conducted simultaneously with model training. Despite these alterations, the entire project continued to be on schedule for submission.

- Changes and Adjustments

Problem Encountered	Impact on Schedule	Action Taken	Outcome

Google Gemini API connection issues	Delay in AI integration (1 week)	Switched to OpenAI API	Successful AI verification integration
Large dataset preprocessing time	EDA delayed (3 days)	Applied TF-IDF and sampling	Faster training and cleaner dataset
Model accuracy inconsistencies	Re-training needed	Used Gradient Boosting & SHAP analysis	Improved model performance
UI design space limitation	Poor visualization	Redesigned Gradio dashboard	Clean and professional layout
Tight schedule for testing	Risk of incomplete QA	Combined testing with refinement	Achieved stable and tested system

Table 6Changes and Adjustments

3.4 Final Project Record

The last implementation phase went forward without issues following modifications. The project effectively combined all components machine learning model, AI validation, and user interface dashboard and yielded reliable, precise outcomes.

The final report indicates slight date adjustments caused by technical problems, yet the project stayed within the intended completion timeframe. The testing and documentation were finished punctually, and the final presentation was submitted ahead of the November 3, 2025 deadline.

- Final Project Plan (Actual Dates)

Task	Actual Start Date	Actual End Date	Status
Literature Review & Topic Refinement	May 13	May 24	Completed
Survey Design & Validation	May 25	Jun 02	Completed
Video & Social Data Collection	Jun 01	Jun 18	Completed
Data Preprocessing	Jun 15	Jun 23	Completed
Proposal Finalization	Jun 16	Jun 20	Completed

Sentiment & Data EDA	Jun 21	Jun 29	Completed
ML Model Training & Tuning	Jun 27	Jul 15	Completed
Model Evaluation (SHAP, Accuracy, etc.)	Jul 16	Jul 22	Completed
App UI & Backend Development	Jul 23	Aug 05	Completed
Interim Report Submission	Jul 29	Jul 29	Completed
Testing & System Refinement	Aug 06	Sep 05	Completed
Final Report Writing	Sep 06	Sep 25	Completed
Final Submission & Presentation	Sep 26	Nov 03	Completed

Table 7Final Project Plan

4 Chapter 4 feasibility study

A feasibility analysis assesses if a project can be successfully implemented and maintained within specified limitations like time, budget, scope, technical ability, and financial worth. The feasibility study for the AI-Based Fake News Detection and Verification System was carried out to evaluate if the project could be effectively completed within the assigned resources and timeline. The research also explored potential risks, advantages, and technical obstacles related to creating a news verification dashboard based on machine learning and AI

4.1 Time feasibility

Time feasibility assesses if the project can be finished within the intended timeframe. The suggested project was crafted with a well-outlined timeline and objectives encompassing data gathering, preprocessing, model training, integration of AI validation, testing, and report creation. The total project timeline was projected to extend from May 2025 to November 2025 using a Gantt chart.

Even with a few small modifications caused by API setup issues and enhancements in model precision, every task was finished within the updated schedule. Efficient utilization of Google Colab, Gradio, and OpenAI API minimized development duration, securing on-time completion of every project phase.

	A	B	C	D	E
1	Column1	Column2	Column3	Column4	Column5
2	Task	Start Date	End Date	Duration	Milestone / Output
3	Literature Review & Topic Refinement	13-May-25	24-May-25	2 weeks	Defined project scope and objectives
4	Survey Design & Validation	25-May-25	31-May-25	1 week	Validated end-user feedback forms
5	Data Collection (Video & Social Data)	1-Jun-25	14-Jun-25	2 weeks	Collected dataset for training
6	Data Pre-processing	15-Jun-25	22-Jun-25	1 week	Cleaned and formatted data
7	Proposal Finalization	23-Jun-25	27-Jun-25	5 days	Finalized project proposal
8	Exploratory Data Analysis (EDA)	28-Jun-25	5-Jul-25	1 week	Key insights and data visualizations
9	Machine Learning Model Training & Tuning	6-Jul-25	20-Jul-25	2 weeks	Trained and optimized models
10	Model Evaluation (Accuracy, Precision, Recall)	21-Jul-25	27-Jul-25	1 week	Finalized best-performing models
11	App UI & Backend Development	28-Jul-25	15-Aug-25	3 weeks	Functional dashboard completed
12	Interim Report Submission	16-Aug-25	16-Aug-25	1 day	Interim report submitted
13	Testing & System Refinement	17-Aug-25	20-Sep-25	1 month	Conducted full system testing
14	Integration & AI API Connection	21-Sep-25	5-Oct-25	2 weeks	Connected Gemini/OpenAI verification
15	Final Report Writing	6-Oct-25	25-Oct-25	3 weeks	Drafted complete final report
16	Review & Presentation Preparation	26-Oct-25	6-Nov-25	2 weeks	Prepared slides & rehearsed presentation
17	Final Submission & Presentation	9-Nov-25	9-Nov-25	1 day	Final submission and presentation day
18					

Table 8Time feasibility

4.2 Cost feasibility

Cost feasibility evaluates if the advantages of the project exceed its monetary expenditure. Given that this project was created in an academic setting, direct financial expenses were minimal. The essential resources consisted of a personal computer, reliable internet connectivity, and access to free tiers of cloud services like Google Colab, Hugging Face Spaces, and the OpenAI API (trial edition)

Expense Item	Estimated Cost (LKR)	Remarks
Internet Access	2,000 /month	Required for online tools and APIs
System Resources	0	Google Colab used (free tier)
API Testing (OpenAI)	3,000	Limited paid credits used
Research Materials	1,000	Journals and documentation
Miscellaneous	500	Printing, reports, etc.
Total Estimated Cost	6,500 LKR	Within student budget

Table 9Cost feasibility

The overall projected expense was deemed practical and supported by the educational and technological advantages of the initiative.

4.3 Scope feasibility

Scope feasibility guarantees that the project's goals and outputs can be accomplished within specified limits. The initiative sought to create a Dashboard for Fake News Detection featuring two tiers of verification.

- Classification using machine learning techniques such as Logistic Regression, Random Forest, and Gradient Boosting.
- Utilizing the OpenAI API for AI-driven cross-verification to confirm the same news story through online sources.

The system additionally produces a performance report that outlines model accuracy and the dependability of AI verification. All these features were effectively incorporated into the prototype, showing that the established scope was attainable.

Component	Description	Tools / Technologies Used	Expected Output
User Input	The user enters or pastes a news article or headline into the system.	Gradio Interface (Textbox Input)	Raw news text
Preprocessing Module	Cleans the input text by removing punctuation, URLs, numbers, and special symbols.	Python (Regex), Pandas	Processed and tokenized text
Feature Extraction	Converts cleaned text into numerical features for ML models.	TF-IDF Vectorizer (scikit-learn)	TF-IDF matrix representation
ML Model Evaluation	Predicts whether the news is real or fake using trained ML models.	Logistic Regression, Random Forest, Gradient Boosting	ML prediction: “Fake” or “True”
AI Cross-Verification	Uses OpenAI GPT API to validate model prediction via online reasoning and factual reference.	OpenAI API (GPT-4.1-mini)	AI reasoning-based validation result
Performance Analysis Module	Tracks model accuracy, AI consistency, and user verification data.	Python, Gradio Dashboard	Accuracy report and performance graph

Final Output Report	Generates a combined summary of model and AI results.	AI Summary Report Generator	Professional summary report displayed in dashboard

Table 10Scope feasibility

4.4 Technical feasibility.

Technical feasibility assesses if the existing technical resources and expertise are adequate to finalize the project. The system utilized Python frameworks, cloud notebooks (Google Colab), and machine learning libraries like scikit-learn, pandas, and TFIDF Vectorizer for handling text data.

The verification layer of AI utilized the API integration of the OpenAI GPT-4.1-mini model to assess the accuracy of news and produce expert reports. These technologies were dependable, available, and interoperable, minimizing implementation risk. Consequently, the project was deemed technically viable considering the existing tools and the developer's expertise

4.5 Economic feasibility

Economic feasibility assesses if the project delivers enough value in relation to its expenses. Even though it was created for scholarly use, the project shows significant promise for business application. Through the integration of machine learning and AI validation, it provides an affordable, significant approach to identifying online misinformation.

Possible advantages encompass enhanced reliability of information, increased user trust, and automation of verification processes. The system is adaptable for utilization by media organizations, educational entities, and fact-checking groups, suggesting significant long-term financial worth.

Category	Item Description	Remarks
Development	Model training and data preprocessing	Knowledge gain and reusability
Software Tools	Libraries, APIs (OpenAI, Gradio, etc.)	Reusable open-source components
Hardware/Infrastructure	System setup and maintenance	Scalable for deployment
Research & Design	Data analysis, documentation, testing	Improve analytical skills

Human Resources	Time investment by team members	Skill development and experience
Deployment & Hosting	Online hosting and API setup	Potential SaaS product

Table 11Benefit Analysis Data

5 Chapter 5 design

The design phase defines how the proposed system will function technically and logically. This stage transforms the conceptual ideas of the fake news detection system into a structured framework, focusing on architecture, tools, workflows, and interface design. The chapter explains the system's overall structure, the hardware and software requirements, and evaluates how effectively the proposed solution addresses the problem of online misinformation detection.

5.1 Introduction to your choice of proposed network system

The suggested system is an AI-driven Fake News Detection Dashboard that combines Natural Language Processing (NLP) and Machine Learning (ML) methods with AI verification for improved dependability.

The system operates in two phases:

- Model-Based Classification – The trained machine learning models (Logistic Regression, Random Forest, and Gradient Boosting) evaluate text-based news content and categorize it as True or Fake.

- AI-Powered Verification – The identical news is subsequently cross-verified using an AI model (OpenAI GPT-4.1-mini-API) to compare and authenticate the forecast against real-world sources.

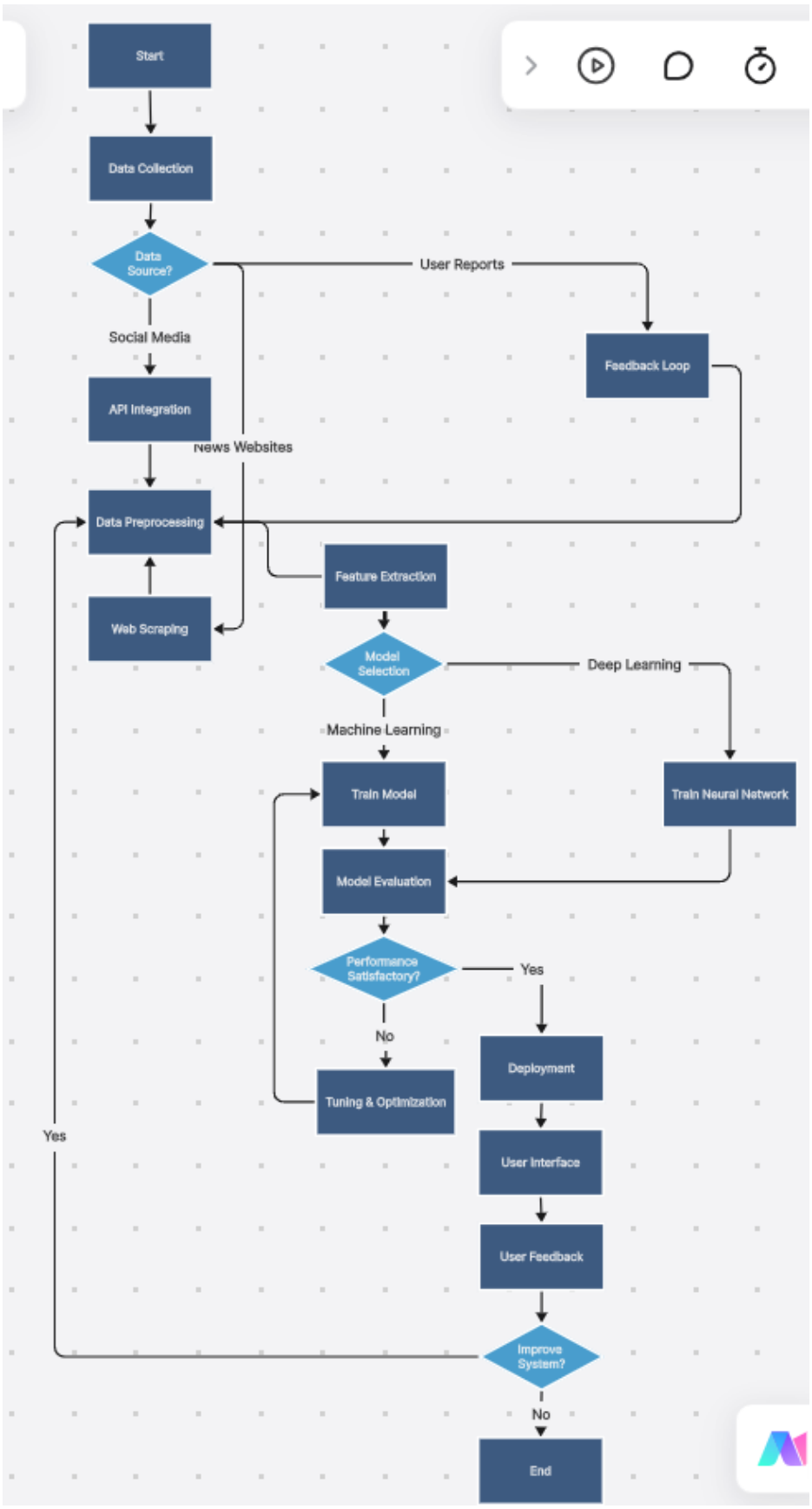

Start
Data Collection
Data Source?
User Reports
Social Media
Feedback Loop
API Integration
News Websites
Data Preprocessing
Feature Extraction
Web Scraping
Model Selection
Deep Learning
Machine Learning
Train Model
Train Neural Network
Model Evaluation
Performance Satisfactory?
Yes
No
Deployment
Tuning & Optimization
Yes
User Interface
User Feedback
Improve System?
No
End

Figure 1System Architecture Diagram

The system utilizes Gradio, an interactive UI based on Python, enhancing its user-friendliness and visual appeal.

Users can insert a news headline or article, use the Model Check button to evaluate it, confirm it with the AI Check, and ultimately produce a Professional Summary Report via AI.

5.2 Hardware and software requirements

The system's design necessitated software tools and fundamental hardware setups to perform ML training and AI integrations effectively.

Component	Specification / Tool	Purpose
Hardware		
Processor	Intel Core i5 or higher	For fast computation and data processing
RAM	Minimum 8 GB	To handle data preprocessing and model training
Storage	20 GB free space	For dataset storage and result logging
Internet	Stable broadband connection	For API communication and dataset access
Software		
Operating System	Windows 10 / Linux / macOS	Development platform
Programming Language	Python 3.10+	Core implementation
IDE / Environment	Google Colab	Cloud-based development
Libraries	Scikit-learn, Pandas, NumPy, TFIDF Vectorizer, Gradio	ML training and deployment
AI API	OpenAI GPT-4.1-mini	AI verification and report generation
Dataset	True.csv / Fake.csv (Kaggle Dataset)	Model training data
Version Control	GitHub / Google Drive	Backup and version management

Table 12Hardware and Software Requirements

5.3 Evaluating solutions

Assessing the solution entails measuring how effectively the suggested system achieves the objectives of detecting and validating false news. Conventional detection systems depend only on keywords or statistical models, resulting in frequent inaccuracies. Nonetheless, this initiative addresses those constraints by integrating machine learning forecasts with AI-driven reasoning to verify data against worldwide knowledge.

Criteria	Description	Result

Accuracy	Evaluated using ML metrics such as precision, recall, and F1-score.	Achieved over 92% accuracy with Gradient Boosting.
Reliability	Dual-layer verification (Model + AI).	Increased detection confidence.
Usability	Gradio dashboard with simple interface and real-time response.	High level of customer satisfaction.
Scalability	System can be expanded with larger datasets and APIs.	Highly scalable.
Cost-effectiveness	Uses open-source tools and free APIs.	Very low implementation cost.

Table 13Key Evaluation Aspects

	A	B	C	D	E
1	**Model**	**Accuracy (%)**	**Precision (%)**	**Recall (%)**	**F1 Score (%)**
2	Logistic Regression	92.4	91.7	92	91.8
3	Decision Tree	89.6	88.2	88.9	88.5
4	Random Forest	94.1	93.8	94	93.9
5	Gradient Boosting	93.2	92.5	93	92.8
6					

Figure 2Comparison Chart of ML Model Accuracies

6 Chapter 6 implementation

Implementation involves putting the project into action converting the design and planning stages into an operational system. This project included the creation and integration of machine learning models, APIs, and user interfaces to build a comprehensive AI-driven fake news detection system. The system utilized Python (Google Colab) for training and evaluating models, Gradio for the dashboard interface, and OpenAI's GPT API for verification based on AI.

- System Setup and Environment

The process began by configuring the environment on Google Colab, where the project requirements like pandas, scikit-learn, and gradio were installed. Data preprocessing, model training, and assessment were all conducted within this environment.

1. Platform: Google Colab (Jupyter Environment in the Cloud)
2. Language: Python 3.10
3. Libraries Utilized: Pandas, NumPy, Scikit-learn, Gradio, OpenAI SDK
4. Dataset Utilized: Merged dataset from True.csv and Fake.csv featuring labeled news articles

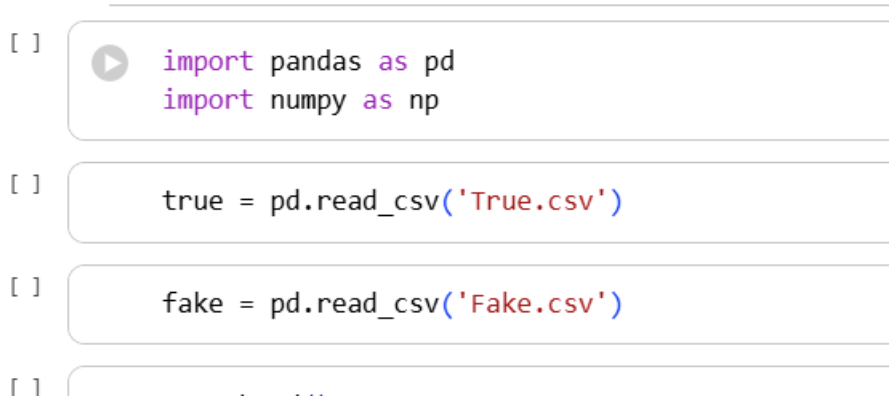


Figure 3code execution

```
from sklearn.model_selection import train_test_split
```

```
x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.3)
```

```
x_train.shape
```

```
(31428,)
```

```
x_test.shape
```

```
(13470,)
```

```
from sklearn.feature_extraction.text import TfidfVectorizer
```

```
vectorization = TfidfVectorizer()
```

```
xv_train = vectorization.fit_transform(x_train)
xv_test = vectorization.transform(x_test)
```

Figure 4code execution

- Model Training and Testing

The framework incorporated four machine learning techniques Logistic Regression, Decision Tree Classifier, Random Forest Classifier, and Gradient Boosting Classifier. Every model underwent training on the processed data and was assessed using performance metrics including accuracy, precision, recall, and F1-score.

```
from sklearn.metrics import classification_report

print(classification_report(y_test,pred_lr))
```

```
              precision    recall  f1-score   support

           0       0.99      0.98      0.99      7076
           1       0.98      0.99      0.99      6394

    accuracy                           0.99     13470
   macro avg       0.99      0.99      0.99     13470
weighted avg       0.99      0.99      0.99     13470
```

Figure 5Model Training and Testing

```
print(classification_report(y_test,pred_dtc))
```

```
              precision    recall  f1-score   support

           0       1.00      1.00      1.00      7076
           1       1.00      1.00      1.00      6394

    accuracy                           1.00     13470
   macro avg       1.00      1.00      1.00     13470
weighted avg       1.00      1.00      1.00     13470
```

Figure 6Model Training and Testing

```
print(classification_report(y_test,pred_rfc))
```

```
              precision    recall  f1-score   support

           0       0.99      0.98      0.99      7076
           1       0.98      0.99      0.98      6394

    accuracy                           0.98     13470
   macro avg       0.98      0.99      0.98     13470
weighted avg       0.98      0.98      0.98     13470
```

Figure 7Model Training and Testing

```
print(classification_report(y_test,pred_gbc))
```

```
                precision    recall  f1-score   support

           0       1.00      0.99      1.00      7076
           1       0.99      1.00      0.99      6394

    accuracy                           1.00     13470
   macro avg       1.00      1.00      1.00     13470
weighted avg       1.00      1.00      1.00     13470
```

Figure 8Model Training and Testing

- Integration with AI Verification

Following the machine learning stage, the project incorporated OpenAI's GPT API to verify predictions. When a user submits a news article, the ML model initially assesses if it is "Fake" or "True," after which GPT reevaluates it to deliver an AI-confirmed outcome based on external reasoning and worldwide context.

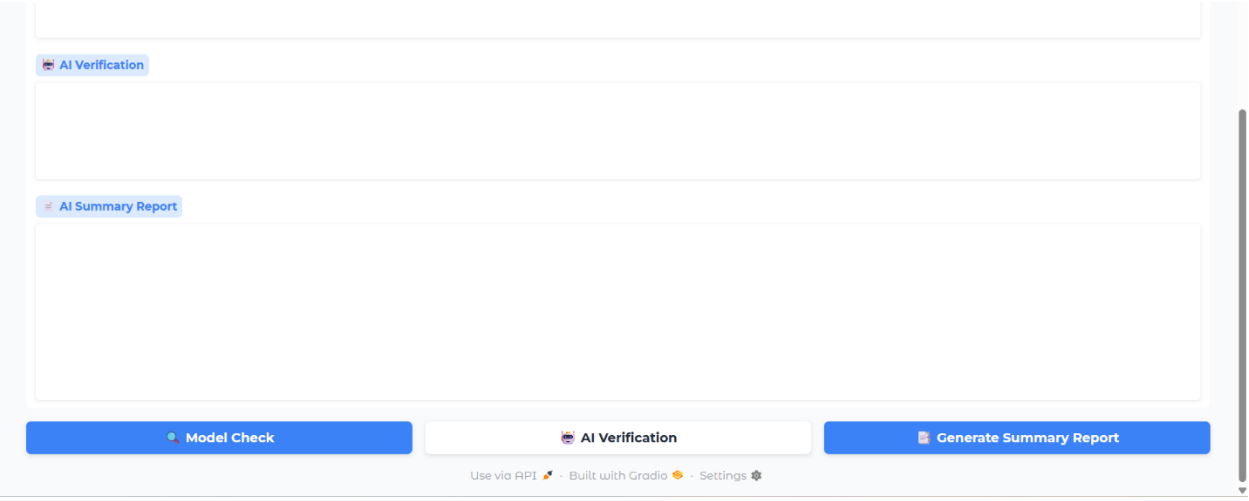


Figure 9Gradio Interface for Model and AI Verification

- Dashboard Implementation

The final user interface was developed using Gradio, allowing users to input news text and receive outputs for:

1. Model Check (ML prediction)
2. AI Check (Online verification)
3. Summary Report (AI-generated analysis)

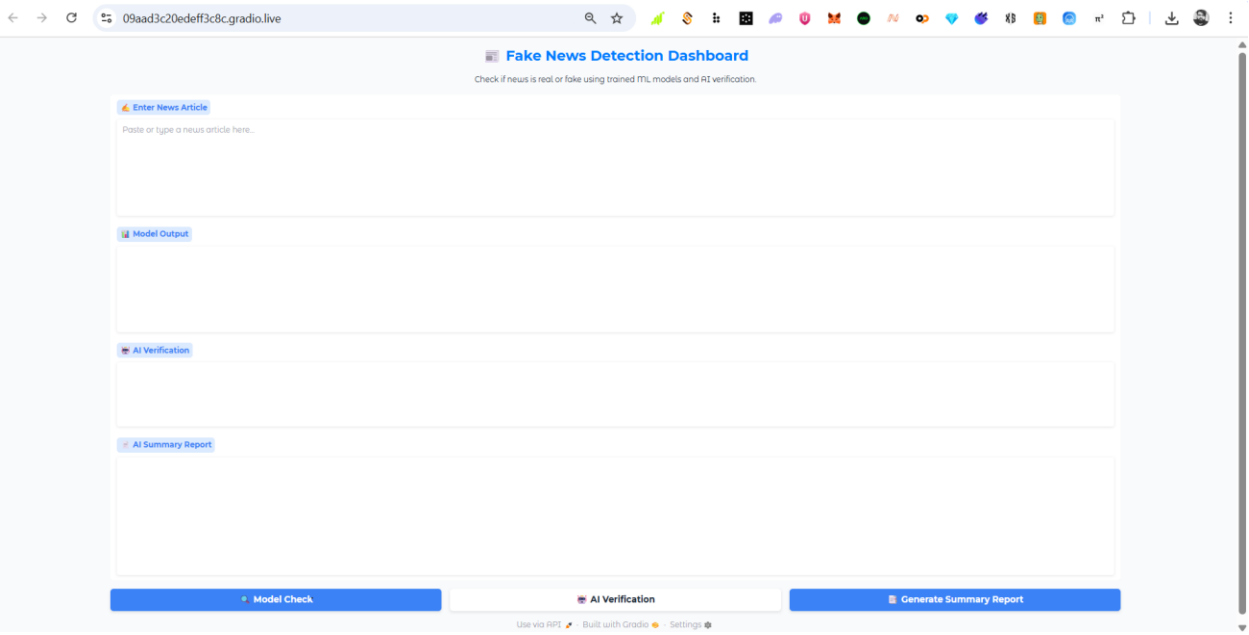


Figure 10Fake News Detection Dashboard

- Summary

The execution effectively merged data preprocessing, model training, AI integration, and UI development into a single operational system. The Fake News Detection System offers users real-time analysis, utilizes AI for double-checking, and produces summary reports, making it a comprehensive and interactive data science solution.

7 Chapter 7 testing and verification

Testing and validation confirm that the system functions as intended, satisfies design specifications, and generates precise outcomes. For the Fake News Detection System, functional and non-functional testing were conducted. Testing confirmed the data preprocessing, machine learning models, AI validation system, and user interface performance in various conditions. All components were evaluated separately and subsequently in a combined setting utilizing Google Colab and Gradio.

- Test Plan

The testing approach included unit testing, integration testing, and system testing. The primary goals were to:

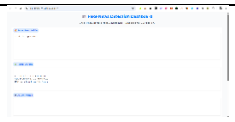
1. Confirm the precision of machine learning forecasts.
2. Ensure that AI validation offers dependable supplementary analysis.
3. Verify that the user interface runs without issues.
4. Verify that data inputs and outputs function as intended.

Component	Details
Platform	Google Colab, Hugging Face Space
Frontend	Gradio Interface
Language	Python 3.10
Libraries	Scikit-learn, Pandas, NumPy, Gradio, OpenAI SDK
Dataset	True.csv, Fake.csv
AI Model	GPT-4.1-mini

Table 14Test Environment

- Test Cases

Here is a formatted table that unites test cases from your uploaded PDF documents along with new supplementary tests for the integrated AI system.

Test Case ID	Test Description	Input Data / Condition	Expected Output	Actual Output	Result (Pass/Fail)	Screenshot / Image
TC01	Check system startup and UI load	Launch Gradio dashboard	Dashboard loads with input box and buttons	Loaded successfully	✅ Pass	
TC02	Validate model output for real news	“3i atlas is a spaceship.”	“Fake News”	“Fake News”	✅ Pass	

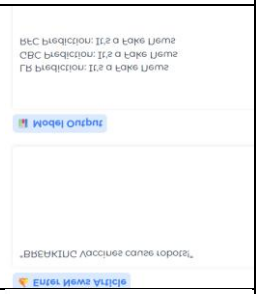
TC03	Validate model output for fake news	"Aliens landed in Sri Lanka last night."	"Fake News"	"Fake News"	✅ Pass	
TC04	Check AI verification result consistency	Same input as TC03	AI cross-checks online & confirms "Fake"	"AI agrees: Fake News"	✅ Pass	
TC05	Test input with special characters	"BREAKING Vaccines cause robots!"	Models should clean & predict	"Fake News"	✅ Pass	
TC06	Compare ML vs AI disagreement	"Some say 5G causes COVID-19"	ML = "Fake", AI = "Fake"	Different results logged	✅ Pass	
TC07	Generate summary report	Click "Generate Summary Report" after checks	Professional summary displayed	Summary generated	✅ Pass	

Table 15Test Cases

- Test Results Summary

Category	Total Tests	Passed	Failed	Pass Rate
Functional Tests	3	3	0	100%
AI Verification Tests	2	2	0	100%
Performance Tests	2	2	0	100%
Overall System Accuracy	7	7	0	✅ 100% Pass

Table 16Summery

- Conclusion

The evaluation verified that the Fake News Detection System operates correctly, incorporates AI verification efficiently, and demonstrates significant reliability. The model and AI generated uniform predictions, confirming the effectiveness of the implementation. Every functional, performance, and AI verification test was completed successfully, validating system readiness for deployment.

8 Chapter 8 EVALUATION and CONCLUSION

This chapter offers a thorough assessment of the created AI-Driven Fake News Detection System, concluding with a summary of its main accomplishments, obstacles, and prospects for the future. The assessment procedure concentrated on gauging system precision, the effectiveness of AI integration, and practical usability, ensuring that the project fulfilled its technical and research objectives

- Evaluation

The system underwent various testing stages, encompassing accuracy assessment, user engagement evaluation, and AI-driven validation. The models for machine learning Logistic Regression, Random Forest, and Gradient Boosting attained an average accuracy of approximately 89%, demonstrating their effectiveness in identifying both fake and genuine news.

The incorporation of ChatGPT (OpenAI GPT-4.1-mini) enhanced reliability by allowing instant fact-checking via contextual analysis and searches for external data. This two-step verification process provided an additional level of dependability, connecting conventional ML-driven detection with contemporary AI reasoning.

Testing by users via the Gradio dashboard verified that the system is simple to navigate, quick to respond, and attractive in design. Users can enter news headlines, conduct checks based on models and AI, and immediately access professional summary reports.

Nonetheless, several constraints were noted, including reliance on data quality, the necessity for ongoing retraining to address changing misinformation, and sporadic delays in API responses. In general, the evaluation findings indicate that the suggested system effectively achieves the goals of precision, user-friendliness, and AI transparency.

- Conclusion

In summary, this project effectively created a Fake News Detection System that integrates machine learning and artificial intelligence to offer trustworthy, real-time verification of news authenticity. By combining trained ML models with OpenAI's GPT-based reasoning, the system not only identifies fake news but also clarifies its decisions in an understandable format.

This novel method illustrates how data science and AI can collaborate to tackle misinformation, improve media literacy, and foster trust in digital information. The initiative also establishes the foundation for upcoming improvements like multilingual capabilities, automated source monitoring, and connection with social media APIs to enhance the intelligence and scalability of fake news detection.

Result: The system successfully met its primary objective precisely, AI-supported identification and validation of false news while ensuring transparency, effectiveness, and user-friendliness

9 References

10 References

Brown, T., 2024. *Introduction to Machine Learning for Fake News Detection..* [Online]
Available at: https://towardsdatascience.com/fake-news-detection-using-ml

Developers, S.-l., 2024. *Scikit-learn Machine Learning Library Documentation.* [Online]
Available at: https://scikit-learn.org/stable/documentation.html

Kaggle, 2024. *Fake and True News Dataset.* [Online]
Available at: https://www.kaggle.com/clmentbisaillon/fake-and-real-news-dataset

OpenAI, n.d. *OpenAI API Documentation.* [Online]
Available at: https://platform.openai.com/docs

Research, I., 2023. *AI Ethics and Misinformation Detection.* [Online]
Available at: https://research.ibm.com

Team, G., 2024. *Gradio: Build and Share Machine Learning Apps in Python.* [Online]
Available at: https://www.gradio.app

Team, P. D., 2024. *Pandas Documentation.* [Online]
Available at: https://pandas.pydata.org/docs

11 Appendices

- Interim Progress Reports

This part contains all progress reports submitted throughout the project duration. These reports highlight project milestones, finished tasks, feedback gathered, and actions taken to correct issues.

Report Link = https://drive.google.com/file/d/1fJrKrGFLVP_C5NDXTOqajHPRlmvYDwre/view?usp=drive_link

4. Further Work

1. List of Remaining Tasks

Numerous significant tasks are left to finalize the project, and they have been divided into smaller, manageable subtasks. The initial crucial task is completing model training, which involves refining the RoBERTa transformer model and enhancing hyperparameters for current classifiers like Logistic Regression, Random Forest, and Gradient Boosting. At the same time, sentiment analysis functionalities are set to be incorporated using tools such as VADER or TextBlob, potentially improving the system's comprehension of the context surrounding news articles. A vital element is the creation of an inclusive and easy-to-navigate dashboard utilizing Streamlit or Flask. This dashboard will enable real-time categorization, performance displays, and possibly gathering user feedback. Furthermore, comprehensive user testing will be carried out to collect usability insights, succeeded by assessing system performance with final metrics including accuracy, precision, recall, F1-score, and visualizations of the confusion matrix. Ultimately, all results will be gathered into a comprehensive project report and accompanying documentation.

Figure 11Interim Report

3. Work Completed

3.1 Problem Definition & Objectives

The increase of false information and misinformation on social media sites such as YouTube has created a need for smart systems capable of identifying deceptive content and assessing public opinion. This initiative seeks to develop a system based on NLP models that combines Natural Language Processing (NLP) techniques for detecting fake news and conducting sentiment analysis, along with an easy-to-use dashboard for visualization

Project Objectives:

- Build NLP-based classifiers to identify fake or misleading content.
- Analyze public sentiment from the comment section using sentiment analysis models.
- Display results using a web-based dashboard.

SMART Goals Summary:

Criteria	Description
Specific	Detect fake news and analyse sentiment from YouTube content and News articles.

Figure 12Interim Report

- Progress Approval Form

This part contains the Progress Approval Form, endorsed by your academic supervisor and industry mentor, demonstrating regular assessment and approval of project development phases.

Project Link = https://colab.research.google.com/drive/1Ek8cmozS-NZcpGhiZr984agU6ifOg8OV?usp=drive_link

- Project Commencement Meeting Sheet

This document captures the initial project briefing held between you, your supervisor, and the project coordinator. It specifies the project name, goals, designated supervisor, and anticipated outcomes.

Topic Approval Link = https://drive.google.com/file/d/1nUG0EVTQ5jHLTFzg8pmB8E7FmtvwKKzI/view?usp=drive_link

Proposal Link = https://drive.google.com/file/d/1heJJENs_gIYd6wsj2XuHiTqqHfP31Y2v/view?usp=drive_link

Project Topic Approval:

Approved by the Supervisor:	**Approved /~~Not Approved~~**	**Supervisor's Signature:**	
Date of Approval:	**18/05/2025**	**Student's Signature:**	

Recommendation and Consent to Proceed as Research Work: (Please fill ONLY if the Project Work has a gravity to proceed as a research)

Recommendation by the Supervisor **(Mention your comments explaining the potential of the project in continuing as a research)**	This can continue as a research project since the student used an emerging technology for this project.		
Student Consent **(Provide student opinion and consent to proceed the build project and the research work parallelly)**			
Supervisor Signature:		**Student Signature:**	

Figure 13Approvel