



Project Report on

Authentext: Detecting AI Generated Text

*Submitted in partial fulfillment of the requirements for the
award of the degree of*

Bachelor of Technology

in

Computer Science and Engineering

By

**Hanan Maryam Jamal (U2103099)
Hannah Rachel Abraham (U2103100)
Maanas Krishnan (U2103128)
Meenakshi Saji (U2103134)**

Under the guidance of

Mr. Ajith S.

**Department of Computer Science and Engineering
Rajagiri School of Engineering & Technology (Autonomous)
(Parent University: APJ Abdul Kalam Technological University)**

Rajagiri Valley, Kakkanad, Kochi, 682039

April 2025

CERTIFICATE

*This is to certify that the project report entitled "**Authentext: Detecting AI Generated Text**" is a bonafide record of the work done by **Hanan Maryam Jamal (U2103099)**, **Hannah Rachel Abraham (U2103100)**, **Maanas Krishnan (U2103128)**, **Meenakshi Saji (U2103134)**, submitted to the Rajagiri School of Engineering & Technology (RSET) (Autonomous) in partial fulfillment of the requirements for the award of the degree of Bachelor of Technology (B. Tech.) in Computer Science and Engineering during the academic year 2021-2025.*

Mr. Ajith S.

Project Guide

Assistant Professor

Dept. of CSE

RSET

Ms. Anu Maria Joykutty

Project Coordinator

Assistant Professor

Dept. of CSE

RSET

Dr. Preetha K G

Professor & HOD

Dept. of CSE

RSET

ACKNOWLEDGMENT

We wish to express our sincere gratitude towards **Rev. Dr. Jaison Paul Mulerikkal CMI**, Principal of RSET, and **Dr Preetha K G**, Head of the Department of Computer Science and Engineering for providing us with the opportunity to undertake our project, "Authentext: Detecting AI Generated Text".

We are highly indebted to our project coordinators, **Ms. Anu Maria Joykutty**, Assistant Professor, Department of Computer Science and Engineering, **Dr. Sminu Izudheen**, Professor, Department of Computer Science and Engineering, for their valuable support.

It is indeed our pleasure and a moment of satisfaction for us to express our sincere gratitude to our project guide **Mr. Ajith S.**, Assistant Professor, Department of Computer Science and Engineering, for his patience and all the priceless advice and wisdom he has shared with us.

Last but not the least, We would like to express our sincere gratitude towards all other teachers and friends for their continuous support and constructive ideas.

Hanan Maryam Jamal

Hannah Rachel Abraham

Maanas Krishnan

Meenakshi Saji

Abstract

The increasing prevalence of AI-generated text has raised concerns regarding its distinguishability from human-written content, particularly in domains where authenticity and personal expression are of prime importance. This project addresses the challenge of identifying AI-generated text and transforming it into a more human-like form. The primary objective of this project is to develop a machine learning model that not only classifies text as AI-generated or human-written but also enhances the AI-generated text to make it indistinguishable from human writing.

The expected outcomes include a robust classification model capable of distinguishing between AI and human-generated text and a transformation pipeline that can refine AI-generated text, making it more human-like in tone, style, and expression.

The social relevance of this work lies in its potential to enhance human-AI collaboration in creative fields, ensuring that AI-generated text can be used effectively without compromising the authenticity and emotional impact that human-written content would provide.

Contents

Acknowledgment	i
Abstract	ii
List of Abbreviations	vi
List of Figures	vii
List of Tables	viii
1 Introduction	1
1.1 Background	1
1.2 Problem Definition	2
1.3 Scope and Motivation	2
1.4 Objectives	2
1.5 Challenges	3
1.6 Assumptions	3
1.7 Societal / Industrial Relevance	4
1.8 Organization of the Report	4
1.9 Conclusion	5
2 Literature Survey	6
2.1 BERT based AI Generated Text classification (2023)	6
2.1.1 Methodology	6
2.1.2 Benefits	7
2.1.3 Drawbacks	7
2.1.4 Conclusion	7
2.2 Controlling AI-Assisted Plagiarism in ESL Writing using AI Detectors (2023)	7
2.2.1 Methodology	8

2.2.2	Benefits	8
2.2.3	Drawbacks	8
2.2.4	Conclusion	9
2.3	DetectGPT: Zero-Shot Machine-Generated Text Detection Using Probability Curvature (2023)	9
2.3.1	Methodology	9
2.3.2	Benefits	10
2.3.3	Drawbacks	10
2.3.4	Conclusion	10
2.4	A Hybrid Method for AI-Generated Text Detection (2024)	11
2.4.1	Methodology	11
2.4.2	Benefits	11
2.4.3	Drawbacks	12
2.4.4	Conclusion	12
2.5	Compression-based Lightweight Detection of Machine Generated Text (2024)	12
2.5.1	Methodology	12
2.5.2	Benefits	13
2.5.3	Drawbacks	13
2.5.4	Conclusion	13
2.6	Chapter Summary	14
2.6.1	Technique Summary	14
2.6.2	Identified Gaps	15
3	System Design	17
3.1	Module Division	17
3.1.1	Data Preprocessing	17
3.1.2	Text Classification Model	18
3.1.3	Explainable AI Module	18
3.1.4	Paraphrasing Module	19
3.2	Datasets Identified	21
3.3	Tools and Technologies	21
3.4	Work Division	21

3.5	Key Deliverables	22
3.6	Project Schedule	23
3.7	Conclusion	23
4	Results and Discussions	24
4.1	Results	24
4.1.1	Text Classification module	24
4.1.2	Explainable AI module	26
4.1.3	Paraphrasing module	26
4.2	Output	27
4.3	Discussion	30
4.4	Conclusion	31
5	Conclusion	32
5.1	Conclusion	32
5.2	Future Scope	33
	References	34
	Appendix A: Presentation	35
	Appendix B: Vision, Mission, Programme Outcomes and Course Outcomes	49
	Appendix C: CO-PO-PSO Mapping	52

List of Abbreviations

LLM	- Large Language Model
MGT	- Machine Generated Text
NCD	- Normalized Compression Distance
MLP	- Multi-layer Perceptron
PLM	- Pre-trained Language Model
NLP	- Natural Language Processing
GPT	- Generative Pretraining Transformer
RoBERTa	- Robustly Optimized BERT Approach
GLTR	- Giant Language model Test Room
BERT	- Bidirectional Encoder Representation from Transformers
TF-IDF	- Term Frequency Inverse Document Frequency
SGD	- Stochastic Gradient Descent
BPE	- Byte Pair Encoding
ROC-AUC	- Receiver Operating Characteristic Area Under the Curve

List of Figures

2.1	Framework of C-Net	14
3.1	Architecture Diagram	17
3.2	Feedforward Lightweight Neural Network	18
3.3	LIME Architecture Diagram	19
3.4	GPT-2 Architecture Diagram	20
3.5	Sequence Diagram	20
3.6	Project Schedule	23
4.1	Accuracy and Loss curves of Feedforward Lightweight Neural Network . . .	26
4.2	User Interface snapshot 1	27
4.3	User Interface snapshot 2	28
4.4	User Interface snapshot 3	28
4.5	User Interface snapshot 4	29
4.6	User Interface snapshot 5	30

List of Tables

2.1	Summary of Different Detection Models, their benfits and drawbacks . . .	15
4.1	Comparison of Different Neural Network Models for Text Classification . .	25
4.2	Comparison of different models for Paraphrasing	27

Chapter 1

Introduction

The project "Authentext: Detecting AI-Generated Text," which is presented in this chapter, will classify text as either human-written or AI-generated and enhance the authenticity of AI-generated text. As artificial intelligence advances, it can be difficult to distinguish between human-generated and machine-generated information in any field, especially when it comes to the value placed on emotional authenticity and personal expression. The project attempts to address this deficiency.

1.1 Background

Text production using artificial intelligence has revolutionized a number of industries, from virtual assistants and customer support to automated reporting and content creation. The most recent natural language processing models can currently produce extremely sophisticated and contextually appropriate content. However, concerns about the authenticity of AI-generated content are growing in popularity, especially in fields where human creativity and originality are crucial. Without obvious authorship, the seamless integration of AI-written content into human-focused domains is expected to easily raise questions about ethics, integrity, and trust. For example, AI-written content in journalistic or educational applications may be diluted or misunderstood because it lacks the emotional intelligence and cultural nuances that are essential to human communication. Therefore, determining whether the text is human-generated or AI-written is the only way to protect communication's authenticity, integrity, and moral standards.

The creation of a machine learning model that can distinguish between writing produced by AI and human is how this project addresses these problems. It includes a transformation module that will enhance the sophistication, expressiveness, and human-likeness of AI-generated text, going beyond basic classification. This tactic aims to promote moral

human-AI cooperation in artistic fields so that AI-generated text can coexist with human writing in a way that strengthens rather than diminishes the emotional resonance and authenticity of human writing. In order to improve its usability in academic, artistic, and professional contexts, the expected result is a tool that can provide a robust solution for identifying AI-generated writing while reengineering it to retain the quality and relatability typical of human writing.

1.2 Problem Definition

Developing a machine learning model that can distinguish between writing that is human-written and AI-generated, as well as translating AI-generated content to make it more human-like, are the objectives of this project. How to distinguish AI-generated text from human-written content for authenticity and emotional depth in AI-generated writing is the issue that the project attempts to address.

1.3 Scope and Motivation

The project's goal is to create a reliable text classification model that can distinguish between text produced by AI and text written by humans. Making AI-generated text more nuanced, emotive, and indistinguishable from human-written content is another aspect of it. In order for consumers to fully comprehend how the models arrive at their classifications, explainable AI would be employed.

It is crucial to develop technologies that will guarantee AI-generated material has the emotional resonance and human touch associated with human writing as its usage grows. Building confidence in AI was the driving force behind this study since writing produced by AI should be less robotic and have a higher emotional intelligence. It focuses on supporting human creativity while adhering to ethical standards, which holds AI use accountable.

1.4 Objectives

1. Create a deep learning model that can accurately identify text as either human-written or AI-generated.

2. To improve comprehension of the model's functioning, apply explainable AI approaches.
3. Establish a procedure for editing AI-generated writing to make it sound more human.
4. Create an algorithm to examine the linguistic characteristics of both human and AI texts.
5. Describe the effectiveness of the model using performance metrics for both text categorization and qualitative transformation accuracy measurements.
6. In order to ensure text sample repeatability across various applications, the developed model should be able to withstand instability.

1.5 Challenges

With AI-generated text, this project must combat the challenge of preserving authenticity during transformation without sacrificing the intended meaning. When altered AI content looks so much like human writing that it can be abused in dishonest circumstances, ethical quandaries may arise.

1.6 Assumptions

1. The unique features of AI-generated text can be identified using machine learning techniques.
2. A sizable and comprehensive label dataset that accurately captures the nuances of both human and AI writing is available.
3. There are well-established techniques for editing AI-generated material to give it a more human and emotional feel.
4. There are pretty well-defined evaluation measures available for classification accuracy as well as for transformation quality.

1.7 Societal / Industrial Relevance

Both applications demonstrate the project’s social and industrial value. AI output that is realistic and meaningful will increase user usage and confidence in creative fields like customer service, content creation, and news authoring. Companies that deal with the authenticity of information can benefit from preventing AI output from deceiving its users by encouraging openness regarding the source and nature of its output.

1.8 Organization of the Report

- **Chapter 1**

An introduction that includes the project’s history, the definition of the problem, its scope and motivation, its goals, its difficulties, its presumptions, and its social and industrial significance.

- **Chapter 2**

Literature Survey, which covers a variety of text categorization techniques with a focus on techniques that can be used to distinguish between text produced by artificial intelligence and text written by humans.

- **Chapter 3**

Module Division, detailing the key project modules, including explainable AI, text classification model, data preprocessing, paraphrasing, and text transformation.

- **Chapter 4**

Results and Discussion, reporting the assessment of our proposed method in three fundamental modules: text classification, explainability, and paraphrasing. Model performance, interpretability, and enhancements are discussed in this chapter.

- **Chapter 5**

Conclusion, summarizing the report’s major findings and highlighting future directions to improve the system.

1.9 Conclusion

The "Authentext: Detecting AI-Generated Text" project has been introduced in this chapter, along with its goals, parameters, and motivations. The research bridges a critical gap in the current AI era: the ability to distinguish between writing authored by humans and AI-generated language, as well as to enhance the authenticity of AI-generated text to make it more emotionally compelling and expressive. In order to facilitate the moral and responsible implementation of AI in text-based applications, a categorization model and a transformation pipeline are being developed to improve the text produced by AI. It ensures that human writing retains its natural quality, giving AI-generated content legitimacy and authenticity while promoting positive human-AI cooperation. In fields where emotional resonance and authenticity are crucial for engagement, such as journalism, content authoring, and customer service, this technique has far broader applicability. The approach, technical design, and implementation of this solution—as well as how it might bridge the gap between AI and human communication—will be covered in the upcoming chapters.

Chapter 2

Literature Survey

2.1 BERT based AI Generated Text classification (2023)

[1]The BERT (Bidirectional Encoder Representations from Transformers) deep learning technique serves as the foundation for the AI-driven text identification model presented in this research. The increasing prevalence of AI-generated information has made its detection crucial in a variety of domains, including media, network security, and public opinion monitoring. The model uses BERT's language comprehension capabilities to identify extremely accurate and consistent material that is both human-written and produced by AI.

2.1.1 Methodology

- Data Preprocessing: Tokenize words, remove stop words, stem, remove excess spaces and numerals, and convert the text to lower case.
- Dataset Preparation: To assess model performance and generalization, divide the data set into 60% training and 40% test sets.
- Model Training: For the particular objective of binary classification for AI-generated text identification, the previously trained BERT model was refined. Batch size and learning rate are two examples of training parameters that are adjusted for convergence stability.
- Evaluation: Determine the performance of a model via accuracy and loss across training epochs, thereby enabling its generalization capacity on the test and training set.

2.1.2 Benefits

- High detection accuracy of 99.72% in training
- High generalization capacity with little loss of accuracy from test and training sets.
- It is a bidirectional approach that improves detection accuracy by enabling BERT to retrieve contextual relationships of text.

2.1.3 Drawbacks

- Has a high computing need due to the complexity of BERT.
- Only does binary classification (AI-generated or human-written), which may not detect nuanced text features.
- Has a tendency to perform poorly when tested on new datasets with significantly different features.

2.1.4 Conclusion

A reliable method for identifying AI-generated content is offered by the BERT-based models used to recognize AI-generated text, which demonstrate remarkably high accuracy and generalization in studies. This study demonstrates BERT's text recognition capabilities and makes recommendations for further model optimization and feature engineering to increase its applicability in various contexts and domains.

2.2 Controlling AI-Assisted Plagiarism in ESL Writing using AI Detectors (2023)

[2]The use of AI detectors to combat the growing danger of AI-enabled cheating in ESL writing is covered in this article. ESL teachers are faced with the additional challenge of identifying AI-generated work in their students' assignments, despite the ease with which systems such as ChatGPT can generate high-quality prose. In order to validate plagiarism, two GPT-2-based RoBERTa classifiers—the GPT-2 Output Detector and CrossplagDetector—are evaluated in this study for their suitability for identifying AI-generated content in ESL contexts.

2.2.1 Methodology

- **Dataset:** A total of 240 essays—120 written by humans and 120 by ChatGPT—were used in the experiment. Before ChatGPT was made available to the public, human essays from ESL student assignments were used to avoid contamination.
- **AI Detectors:** GPT-2 Output Detector Demo and Crossplag AI Content Detector, two RoBERTa-based AI detectors, were used. Each of the two detection technologies was employed separately and contrasted with the others.
- **Data Analysis:** Both detectors produced originality scores for each essay. The accuracy and effectiveness of each detector in distinguishing between text generated by AI and human were evaluated using descriptive and inferential statistical methods, such as confusion matrices and Mann-Whitney U tests.

2.2.2 Benefits

- Extremely high detection accuracy for AI-generated text, with both detectors achieving similar classification accuracy of about 89
- Despite being trained using an earlier AI model, they are able to distinguish between essays written by humans and those produced by AI.
- A potential first step toward a reliable AI-based plagiarism detector is the RoBERTa model’s refined training.

2.2.3 Drawbacks

- Regardless of whether detection reliability is affected, detectors trained on GPT-2 data will not work on more recent, complex models like GPT-3.5.
- These detectors are impractical for low-resource environments due to high computational needs.
- Some texts were misclassified by both detectors, either as false negatives or false positives (human texts that were mistakenly identified as AI-written).

2.2.4 Conclusion

This research identifies the strengths and weaknesses of AI detectors in fighting AI-aided plagiarism in ESL writing. The findings show that GPT-2-based RoBERTa classifiers, including the GPT-2 Output Detector and Crossplag AI Content Detector, can accurately identify human-written and AI-written essays. Nevertheless, the use of outdated AI models for training is a major disadvantage since newer models can bypass detection. Moreover, the computational requirements of such detectors render them less accessible in low-resource settings. Although AI-powered plagiarism detection software is a promising direction toward ensuring academic integrity, ongoing improvements in detection techniques are needed to keep up with the development of AI writing technologies.

2.3 DetectGPT: Zero-Shot Machine-Generated Text Detection Using Probability Curvature (2023)

[3]Large language models' (LLMs') increasing complexity has raised demand for AI-generated text identification and application in domains like academia and media where content authenticity is crucial. Without the need for extensive labeled datasets or model-specific fine-tuning, "DetectGPT" presents a revolutionary zero-shot method for identifying machine-generated text by looking for patterns in the unique curvature of AI-generated text.

2.3.1 Methodology

- Probability Curvature Hypothesis: Avoids the presumption that the log probability function of the producing model has negative curvature, which is where machine-generated text is found.
- Log Probability Analysis: Since AI text has a larger log probability than human text, it compares the perturbed and original log probabilities.
- Mask-Filling Model Perturbation Sampling: This method uses T5 or similar models to generate slightly altered text samples (perturbations).
- Calculation of Perturbation Discrepancy Score: This is the difference between the probability of the original text and the mean probability of the perturbed versions

to determine its origin.

- **Algorithm Design:** Regions of negative curvature in log probability functions are used as a signal of AI development in this effective method of comparing probability differences.

2.3.2 Benefits

- **Zero-Shot Detection:** It's a flexible tool for new models and data kinds because it doesn't require fine-tuning on specific data or models.
- **High Accuracy:** It achieves high AUROC scores in experiments across datasets, reflecting significant gains over existing zero-shot techniques.
- **Effective Detection Process:** Compared to supervised methods, it requires less computing power because it only requires log probabilities and a mask-filling model.

2.3.3 Drawbacks

- **Model Access Dependency:** Not all APIs can provide white-box access to the log probabilities of the producing model.
- **Reduced Paraphrasing Robustness:** When AI-generated content is extensively paraphrased or altered by hand, performance suffers.
- **Computational Complexity:** When working with large datasets, perturbation sampling and comparison require extra computational steps that are computationally demanding.

2.3.4 Conclusion

In contrast to the traditional supervised detection techniques, DetectGPT is a promising zero-shot approach for machine-generated text detection based on probability curvature. In identifying domains where model access is feasible, the method of emphasizing local perturbations and probability differences has shown itself to be very accurate. This work demonstrates how zero-shot techniques can adapt to evolving LLM capabilities, providing a tool to help preserve the authenticity of content.

2.4 A Hybrid Method for AI-Generated Text Detection (2024)

[4]In order to distinguish AI-text from human-language, this work suggests a hybrid approach that combines deep machine learning models with the traditional text analysis method, TF-IDF. The method uses an ensemble model that leverages the strengths of both conventional and innovative methods to address the problem complexity.

2.4.1 Methodology

- TF-IDF (Term Frequency-Inverse Document Frequency): Retrieves terms by taking into account term frequency in the documents and the term rarity within the corpus.
- Bayesian Classifiers: Applies Bayes' theorem to estimate the probability that a piece of text would belong to certain classes, e.g., AI-generated versus written by human.
- Stochastic Gradient Descent (SGD): Learns model parameters by iteratively minimizing a loss function.
- CatBoost and LightGBM: They are gradient boosting algorithms based on decision trees that may be employed to tackle complex big datasets.
- Byte Pair Encoding (BPE): It encodes text into subword units to enable improved generalization and treatment of rare words.
- DeBERTa (Disentangled BERT): It is a state-of-the-art language model that generates context-aware text representations.
- Ensemble Modeling: It averages the predictions of various models with the help of a voting system to enhance accuracy and minimize bias.

2.4.2 Benefits

- High Accuracy: The model's high ROC-AUC score suggests that it is effective at differentiating between AI and human writing.
- Robustness: It uses a variety of models to increase resistance to various textual styles and structures.

- Scalability: It was created to effectively process complicated and big datasets with minimal processing overhead.

2.4.3 Drawbacks

- Complexity: The hybrid model demands a large amount of processing power and adjustment.
- Data Dependency: Performance is significantly influenced by the caliber and variety of training data.
- Overfitting risk: The combination may limit generalization on unseen data by overfitting specific dataset features.

2.4.4 Conclusion

In order to maintain the authenticity of content on digital platforms, this research presents a hybrid approach that shows very high accuracy in identifying AI-generated content. By laying the groundwork for robust solutions in the identification of AI-generated text, the suggested model can increase confidence in online communication.

2.5 Compression-based Lightweight Detection of Machine Generated Text (2024)

[5]In the absence of model internals, this study introduces C-Net, a lightweight compression-based model that can effectively and efficiently detect machine-generated text (MGT) in black-box scenarios. Effective MGT identification across several domains is required due to the recent emergence of sophisticated AI-generated material, which includes automated online evaluations and fake news. This paper demonstrates how C-Net may overcome the drawbacks of earlier methods by utilizing text compressions and parameter efficiency as a key feature extraction method, making it incredibly resource-efficient and adaptable.

2.5.1 Methodology

- Data Generation: The question-answer dataset is used as the reference set to which C-Net will compare the sample machine-like text that it generates.

- **Text Compression:** It provides lossless text compression for comparison and uses Normalized Compression Distance (NCD) to determine how comparable sample text and machine reference are.
- **Classifier Training:** To accurately categorize MGT with minimal computational overhead, it combines a classifier with frozen embeddings from a pre-trained language model.

2.5.2 Benefits

- Its high detection accuracy from fewer parameters makes it suitable for real-time and low-resource applications.
- It functions well in black-box settings without requiring access to specific model parameters or data.
- Cross-language capabilities are suggested due to its success on English and Chinese datasets.

2.5.3 Drawbacks

- Fine linguistic subtleties in complex AI-generated content will probably be challenging for compression-based models to handle effectively.
- Affected scalability in several scenarios, it yields subpar results for languages or texts that differ greatly from the training data structure.
- It is lightweight, but when used with lengthy or complex texts, performance is probably compromised on performance when working with lengthy or extremely complicated texts.

2.5.4 Conclusion

By striking a balance between high accuracy and reasonable, low resource requirements, C-Net offers a good, workable solution for MGT identification. It was created with practical use in settings where access to internal model content is restricted. Even though it might make it more difficult to detect more complex AI-generated material and many other

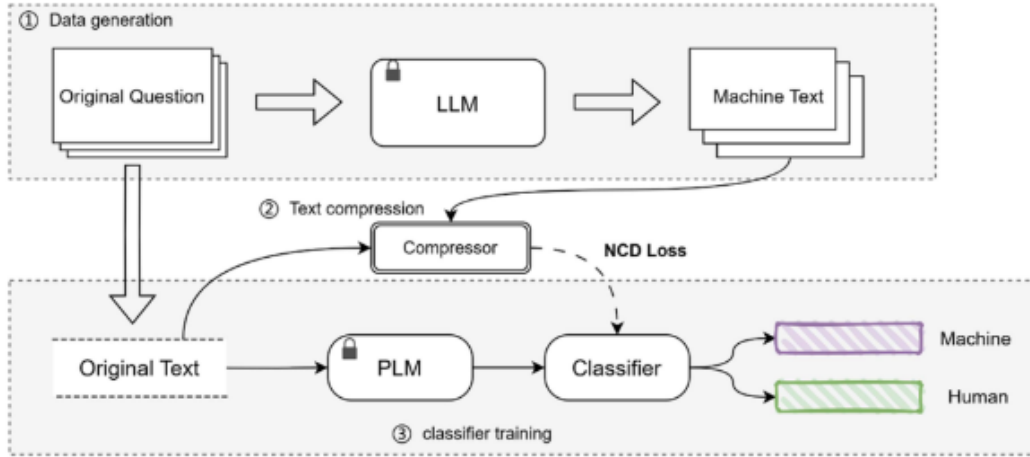


Figure 2.1: Framework of C-Net

languages, C-Net is one such incredibly resource-efficient and high-performance MGT detection technique.

2.6 Chapter Summary

2.6.1 Technique Summary

An overview of the benefits and drawbacks of each AI text identification technique is given in the following table.

Method	Advantages	Disadvantages
[1]BERT-based Classifier	High accuracy, excellent generalization capability, excellent in detecting AI-written text with deep contextual awareness.	Moderately higher test loss than training loss; model's performance may decline with more sophisticated or varied real-world texts.
[2]RoBERTa-based Classifiers	Strong at detecting AI-generated text, including AI-assisted plagiarism in ESL settings. Greater accuracy than other classifiers.	Inconsistent detection accuracy across datasets; problems with identifying newer iterations of AI models, less accurate in certain settings.

Method	Advantages	Disadvantages
[3]DetectGPT	Zero-shot detection without requiring further training data.	Sensitive to slight rewrites or paraphrasing, needing fine perturbations.
[4]Hybrid Detection Model	Traditional and latest techniques combined for high detection rates.	Ensembles and several instances of models make it complex.
[5]C-Net	Effective for machine-generated text detection in black-box use cases, less number of parameters, 99.5% accuracy.	Less flexibility; performance can get affected when used with extremely large datasets or new models.

Table 2.1: Summary of Different Detection Models, their benefits and drawbacks

2.6.2 Identified Gaps

Although AI text detection techniques have advanced significantly, there are multiple limitations in today’s state of the art:

1. **Narrow Cross-Model Generalization:** The majority of models in use today are made to identify outputs from specific AI models, such as GPT-2 or GPT-3. Yet, given the speed at which generative AI is developing, they might not be robust enough to be used to more recent and complex models, losing their usefulness.
2. **High-Resource Computational Dependency:** Cutting-edge detection models can be computationally demanding, especially deep learning architectures like BERT and RoBERTa. Their dependence prevents them from being used in real-time or in environments with limited resources.
3. **Binary Classification Constraint:** Whether AI-generated or human-written, binary classification has been a major focus of the models. These techniques might not be sufficient for the required nuanced material. Investigating hierarchical or multi-class classification with varying degrees of AI text production engagement might be required.
4. **Performance Degradation with Paraphrasing or Editing:** When AI-generated con-

tent is paraphrased or somewhat modified, many detection methods suffer from a reduction in resilience. This flaw makes real-world use more difficult because content may have been slightly altered or rephrased to evade detection.

5. Language and Domain Restrictions: Certain models perform best in specific areas or languages (such as Chinese or English). For detection techniques to have broader relevance across various contexts and content kinds, they must be cross-linguistic and cross-domain adaptive.

Chapter 3

System Design

The project is comprised of multiple large-scale modules, each of which deals with one distinct facet of the system’s functionality and design.

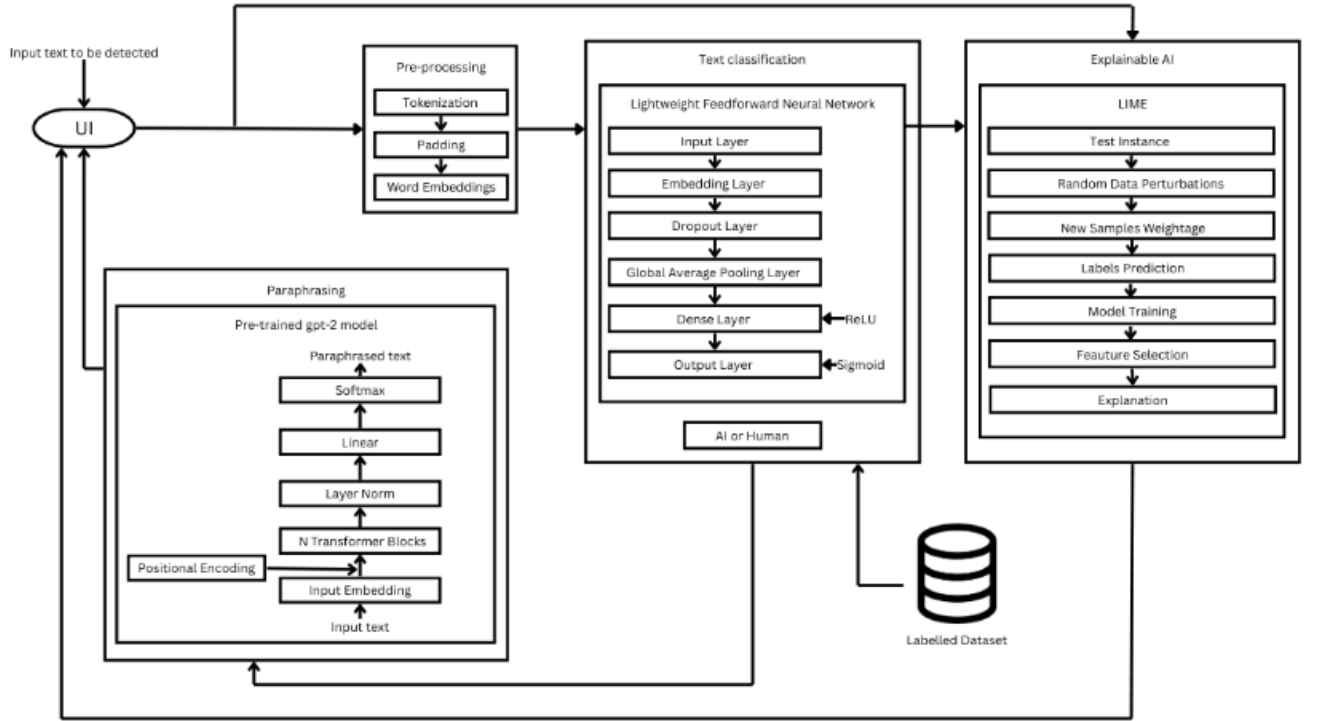


Figure 3.1: Architecture Diagram

3.1 Module Division

3.1.1 Data Preprocessing

To establish an unbiased sample, the data is first balanced by taking an equal amount of data points from each class. Any inherent order is then eliminated by shuffling the data. After that, the data is separated into sets for training and validation in order to assess the model. The word-to-index mapping created from training data is then used to

tokenize the input text and transform it into integer sequences. To ensure that the input dimensions are the same, the sequences are padded to a predetermined size. These integer indices are transformed into dense word vectors during training by feeding this prepared input into an embedding layer.

3.1.2 Text Classification Model

With the aid of a neural network, the text classification module separates text produced by artificial intelligence from text written by humans. Word indices are mapped to dense vectors by an embedding layer, and overfitting is prevented by a dropout layer. After that, embeddings are reduced to a fixed-size vector via a global average pooling layer. It is routed into the ReLU-activated hidden dense layer, followed by a sigmoid-activated layer that outputs binary classification probabilities. With accuracy as the main criterion, the model is trained using binary cross-entropy loss and the Adam optimizer (learning rate 0.00045). A logging callback monitors training progress, and early termination monitors validity loss to prevent overfitting. This compact design offers reliable and effective text classification, making it easier to distinguish between information produced by AI and that written by humans.

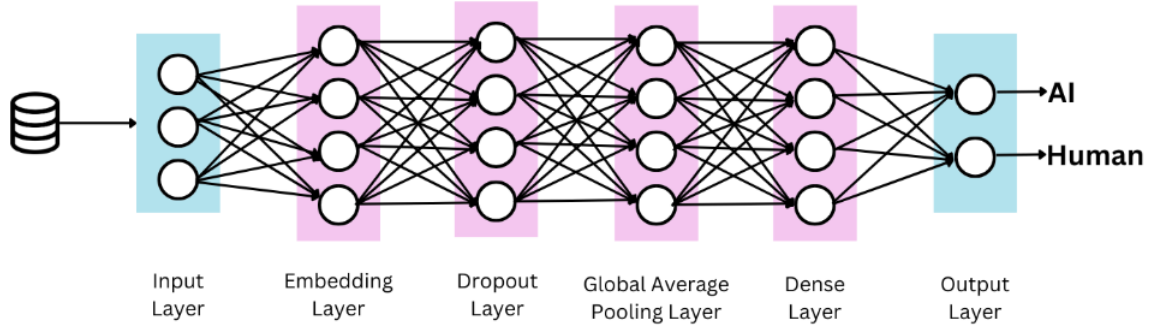


Figure 3.2: Feedforward Lightweight Neural Network

3.1.3 Explainable AI Module

This module provides explainable AI techniques such as the LIME method, which creates a perturbed sample, gives the samples weights, and trains a straightforward interpretable model that explains how the features affect the classification model's predictions. The procedure is clear and easy to utilize in this way.

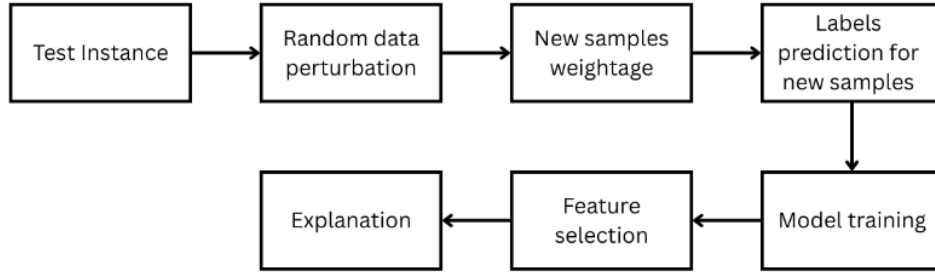


Figure 3.3: LIME Architecture Diagram

3.1.4 Paraphrasing Module

To paraphrase in our process of converting AI-written text into more human composition, we have employed the use of the GPT-2 Large model. The GPT-2 is a transformer language model that is most effective for natural language generation because it can generate text that is coherent and contextually consistent. By utilizing beam search decoding and sampling techniques like top-k (70), top-p (0.95), and repetition penalty (1.2), we sought to generate paraphrased responses that reproduce the original meaning with enhanced fluency and minimized repetitiveness. The model was explicitly instructed to paraphrase the input without altering its length and semantic coherence, such that the text converted remains natural and genuine. This paraphrasing process serves to play an important part in perfecting content generated by AI to make it closer to human-authored content and yet without losing its core meaning.

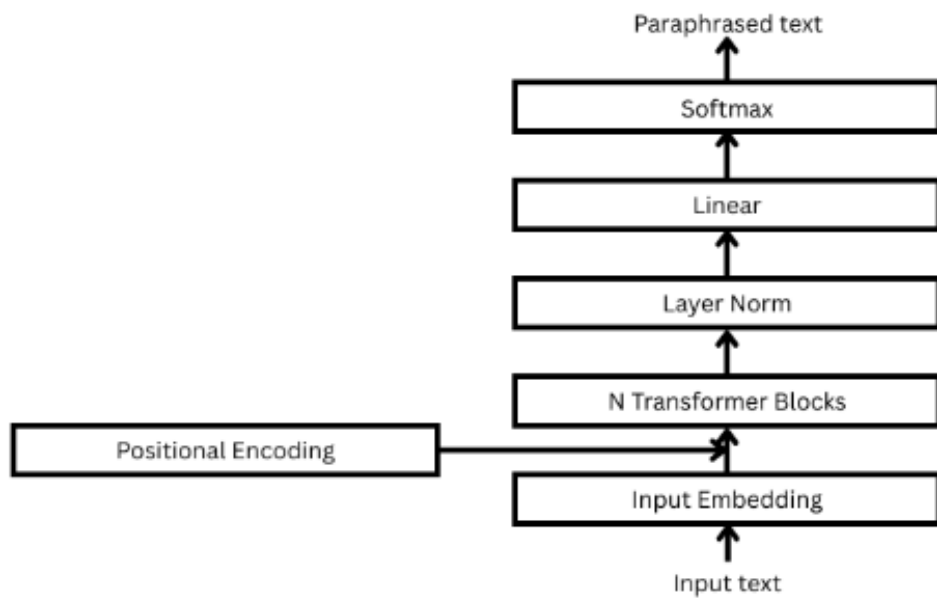


Figure 3.4: GPT-2 Architecture Diagram

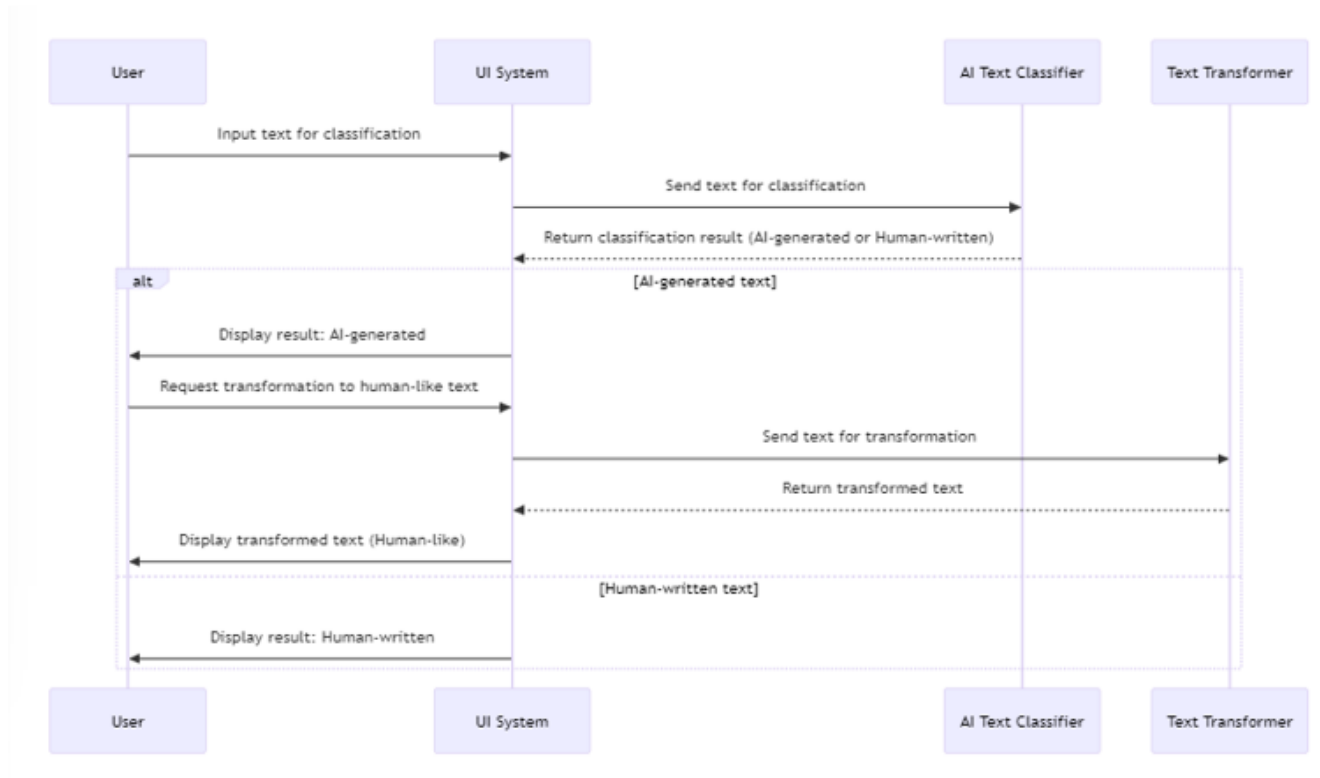


Figure 3.5: Sequence Diagram

3.2 Datasets Identified

Each of the roughly 500,000 items in the data collection represents a distinct text sample. The two domains are:

- **Text:** A string field that holds the record's text content.
- **Generated:** A binary field that contains the text's source, with 1 denoting AI-generated text and 0 denoting human-generated content.

3.3 Tools and Technologies

- **Software:**
 1. Python - backend
 2. HTML,CSS,Flask - frontend
 3. VScode
 4. Labelled binary dataset
- **Hardware:** Any device running Windows 10 or later that has the necessary software installed can access the web application.

3.4 Work Division

- **Meenakshi Saji:** Concentrated on preprocessing the data. Implemented a tokenizer for tokenizing the text into sequences and padded the sequences to input of equal sizes. Optimized vocabulary management for the handling of out-of-vocabulary tokens and synchronized the vocabulary with the embedding layer of the model. Examined and calibrated sequence lengths for the best possible model performance.
- **Hannah Rachel Abraham:** Accountable for model design and development. This comprised model structure development, such as embedding layers, dropout layers, and dense layers trained for binary classification. Integration of text classifier and LIME.

- **Hanan Maryam Jamal:** Responsible for dataset cleaning, removal of inconsistencies and missing values, and raw text tokenization and filtering; and balancing data by taking samples of the same number of AI-generated and human responses. Applied explainable AI techniques like LIME to make the model predictions transparent and interpretable. Also designed the user interface.
- **Maanas Krishnan:** Concentrated on post-training assignments. A prediction script was created for real-time inference, and the model’s generalizability was examined using unseen data. In order to improve the human-likeness, tone, and style of AI-generated writing, a paraphrase model with sophisticated algorithms was used.

3.5 Key Deliverables

Classification Model

- A machine learning model that has been trained and tested to accurately classify text into categories created by AI and by humans.
- Performance metrics that show the model’s stability and dependability, including accuracy, precision, recall, and F1 score.

Text Transformation Pipeline

- A software module that improves AI-written writing so that its tone, style, and emotional impact are all identical to those of human-written language.

Explainability Module

- An explainable AI module that describes key characteristics or patterns that influenced the model’s transformation and classification decision-making.

Working Prototype

- Users will be able to write text, determine if it was created by AI or by humans, and amend AI-generated content if needed in this user-friendly prototype that integrates transformation and classification capabilities.

3.6 Project Schedule

Task	October	November	January	February	March
Feature Extraction	<div></div>				
Text Classification		<div></div>	<div></div>		
Explainable AI			<div></div>	<div></div>	
Paraphrasing				<div></div>	<div></div>
GUI Integration					<div></div>

Figure 3.6: Project Schedule

3.7 Conclusion

The architecture and essential elements of the "Authentext: Detecting AI-Generated Text" project are methodically explained in this chapter. In order to accomplish the project's goals, this chapter divides the system into modules that focus on essential functions such text categorization, explainable AI, data pretreatment and feature extraction, and text transformation. Information about the dataset, tools, and technologies used is also included in this chapter. In addition to deliverables like a text transformation pipeline, a high-performing classification model, and an intuitive prototype, modular design promotes efficacy, scalability, and transparency—all of which contribute to the project's goal of classifying and refining AI-generated text until it is indistinguishable from human-written content.

Chapter 4

Results and Discussions

This chapter presents the result and discussion of our proposed approach, featuring three key modules: text classification, explainable AI (XAI), and paraphrasing. The classification module is devised in such a way as to distinguish AI-generated text from human-written text. For this purpose, we employed and compared two lightweight neural network architectures: compression-based lightweight neural networks and feedforward lightweight neural networks. The correctness of these models is investigated using their classification accuracy and efficiency. To enhance the interpretability of our classification results, we integrated LIME (Local Interpretable Model-agnostic Explanations) as our explainability technique. The paraphrasing module focuses on transforming AI-generated text to make it more indistinguishable from human writing. We implemented and compared multiple transformer-based models for this task, including T5, GPT-2, GPT-4, and Gemini. These models were evaluated in terms of their ability to preserve meaning, improve fluency, and enhance readability.

4.1 Results

4.1.1 Text Classification module

In our early experiments, we used C-Net, a lightweight neural network based on compression, and it produced a high accuracy of 97% in classification. But upon further analysis, we realized that there were issues related to Normalized Compression Distance (NCD) loss computation, which resulted in overfitting. The model also had a propensity to learn patterns too rigidly, making its ability to detect unseen samples of text of any content or situation suboptimal. To overcome these deficiencies, we shifted to a feedforward light neural network, whose accuracy was 95.69%. Although somewhat lower than that of the first C-Net model, the feedforward neural network showed stronger generalization and

resilience. In addition, we pondered employing BERT, which is a transformer-based model of high performance on NLP tasks. Nonetheless, BERT is a black-box model, constraining us to access and interpret its internal parameters. As a central focus of our project, especially through LIME (Local Interpretable Model-agnostic Explanations), we shunned BERT because of incompatibility with our explainability needs. The above comparison pinpoints the balance between accuracy, interpretability, and generalizability, corroborating our use of a feedforward lightweight neural network for detecting AI-generated text.

Model	Advantages	Disadvantages
[1]BERT	Strong contextual embedding with deep semantic understanding	Resource-intensive, functions as a "black box," making it harder to explain and integrate with XAI, limits flexibility due to its fixed architecture
[5]Compressed lightweight neural network	Fewer layers with reduced neurons and parameters, high accuracy	Struggles with highly diverse text patterns, NCD loss does not capture differences in text efficiently
Feedforward lightweight neural network	Fewer layers, minimal neurons, and optimized parameters, highly interpretable, captures key patterns in text, high accuracy	Lacks deep contextual understanding

Table 4.1: Comparison of Different Neural Network Models for Text Classification

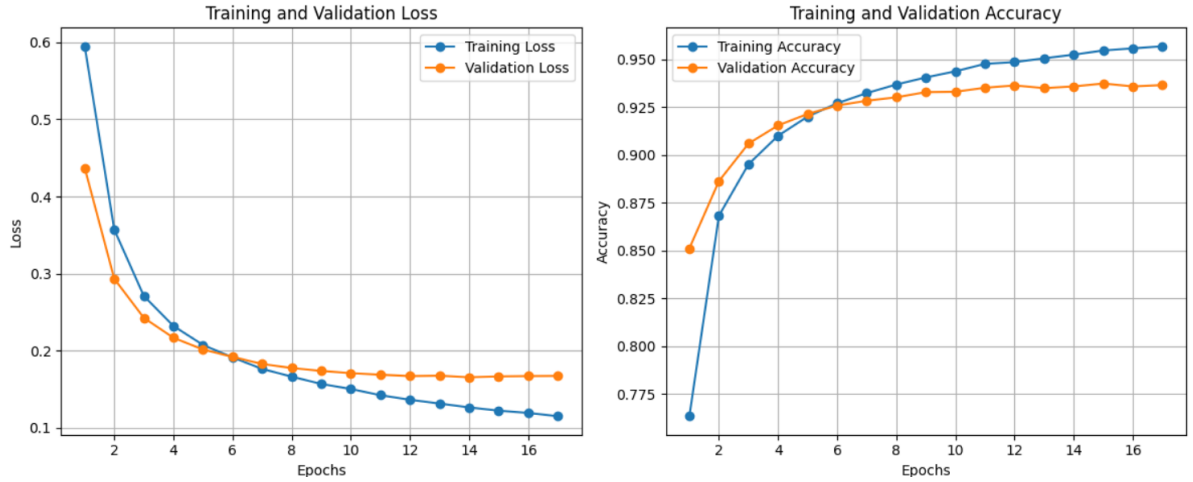


Figure 4.1: Accuracy and Loss curves of Feedforward Lightweight Neural Network

4.1.2 Explainable AI module

In order to facilitate the interpretability of our classifier model, we incorporated Local Interpretable Model-agnostic Explanations (LIME) as our method of explainability. LIME sheds light on the decision-making process of the model by presenting significant features of the input text that drive classification. Our model explainability was measured through the indication of the top 20 impacting features in every sample of the input text. Some of these features were specific words or phrases that played a crucial role in deciding the outcome of the model. The results prove that LIME effectively gives us transparency by helping us know why a given text is labeled as AI-generated or human-written. The visual explanations LIME produces validate our neural network’s prediction. This builds up more confidence in the model’s predictions, increasing the interpretability for end users.

4.1.3 Paraphrasing module

Our paraphrasing module is programmed to convert AI-written text so that it becomes even less distinguishable from human text without losing the original intent. To begin with, we used the T5 model for paraphrasing. But while testing, we noticed that T5 showed a very high inclination to summarize input text instead of rewording it. This behavior rendered it unfit for our target use since the paraphrased outputs tended to be too general or lacked structural diversity. To address this shortcoming, we used a pretrained GPT-2 model, and to our relief, it performed much better. In contrast to T5,

GPT-2 produced paraphrased text that maintained the entire context and meaning of the original input but with suitable lexical and syntactic variation. The outputs closely matched the provided prompts, indicating that the model is indeed good at following instructions to generate precise rephrasals.

Model	Advantages	Disadvantages
T5	Generates fluent and grammatically correct text, capable of understanding context	Tends to summarize instead of paraphrasing, losing important details in the process
GPT-2	Produces accurate paraphrases while preserving meaning, follows prompts effectively	Needs tuning or careful prompt engineering to work at best, occasionally produces wordy responses

Table 4.2: Comparison of different models for Paraphrasing

4.2 Output

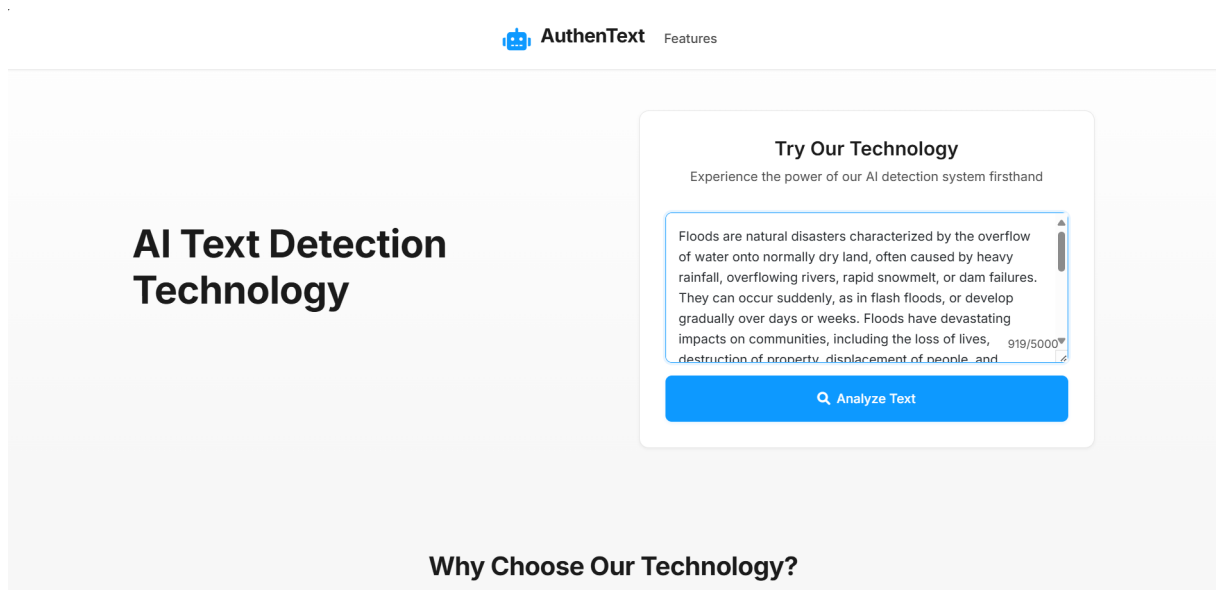


Figure 4.2: User Interface snapshot 1

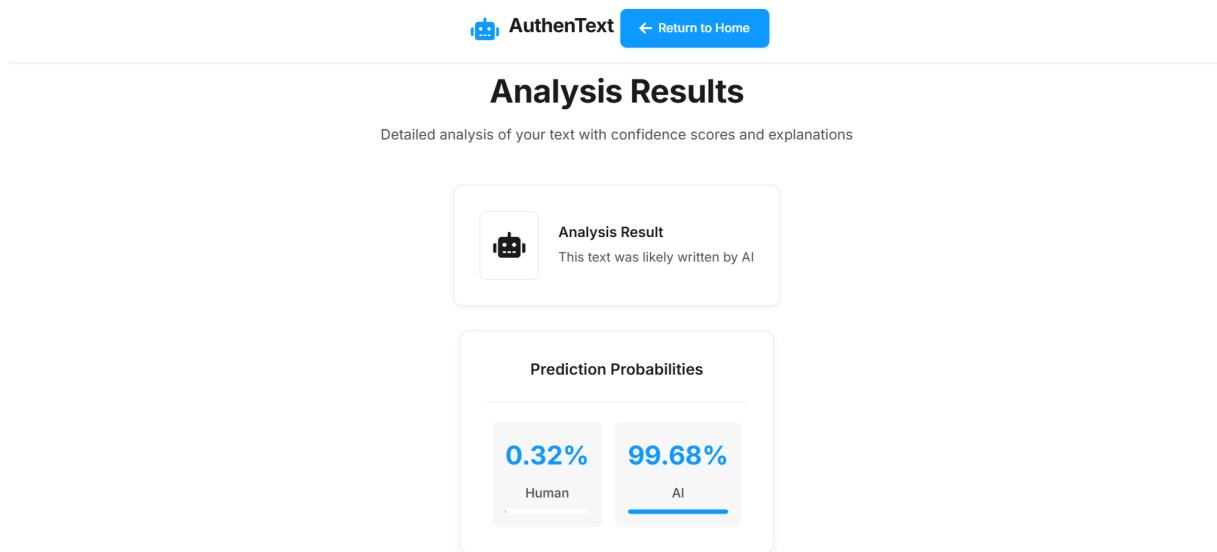


Figure 4.3: User Interface snapshot 2

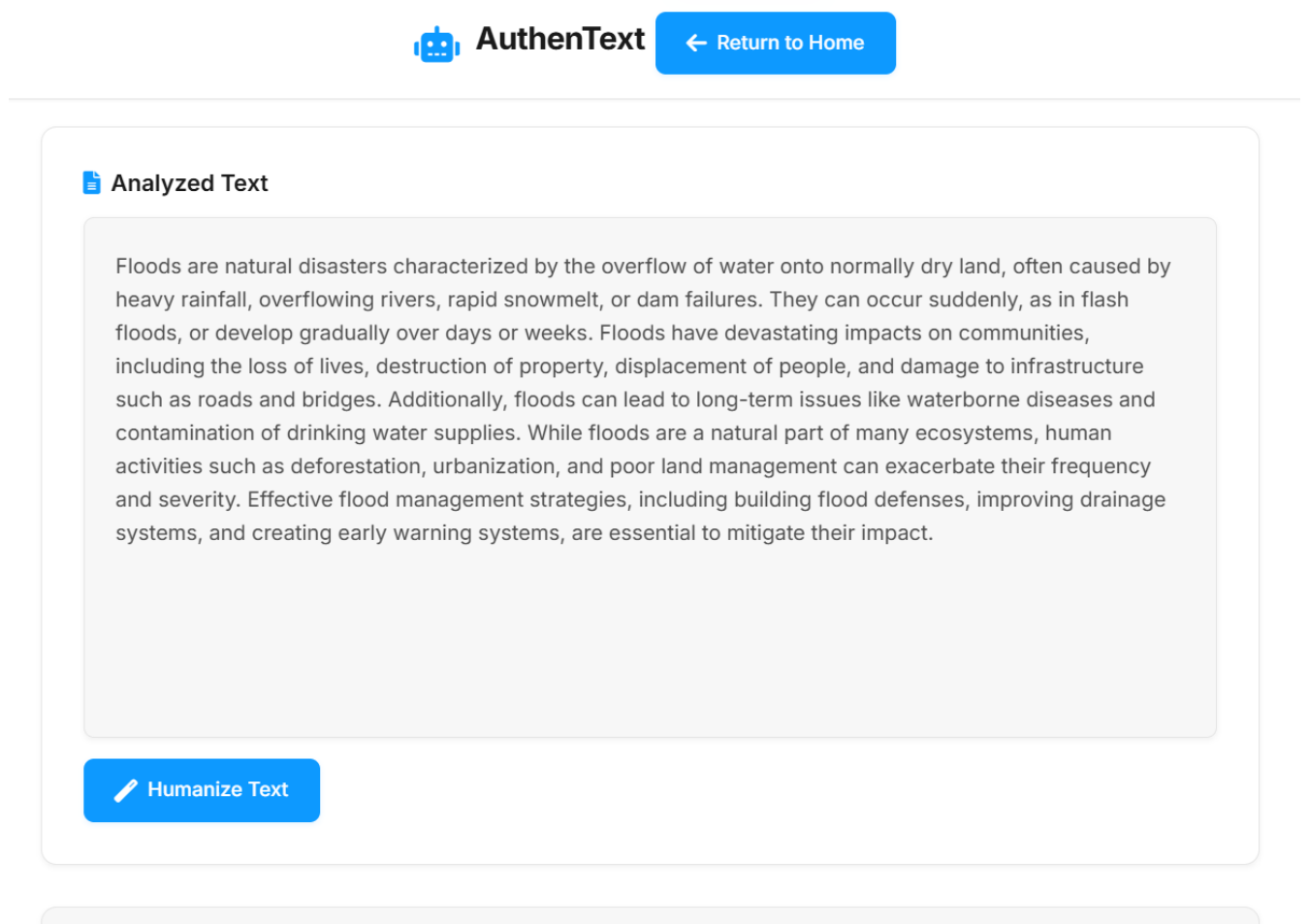


Figure 4.4: User Interface snapshot 3

property
0.06
infrastructure
0.05
flood
0.05
devastating
0.05
as
0.05
early
0.05
are
0.04
exacerbate
0.04
urbanization
0.03
displacement
0.03
gradually
0.03
damage
0.03

Text with highlighted words

Floods are natural disasters characterized by the overflow of water onto normally dry land, often caused by heavy rainfall, overflowing rivers, rapid snowmelt, or dam failures. They can occur suddenly, as in flash floods, or develop gradually over days or weeks. Floods have devastating impacts on communities, including the loss of lives, destruction of property, displacement of people, and damage to infrastructure such as roads and bridges. Additionally, floods can lead to long-term issues like waterborne diseases and contamination of drinking water supplies. While floods are a natural part of many ecosystems, human activities such as deforestation, urbanization, and poor land management can exacerbate their frequency and severity. Effective flood management strategies, including building flood defenses, improving drainage systems, and creating early warning systems, are essential to mitigate their impact.

Figure 4.5: User Interface snapshot 4

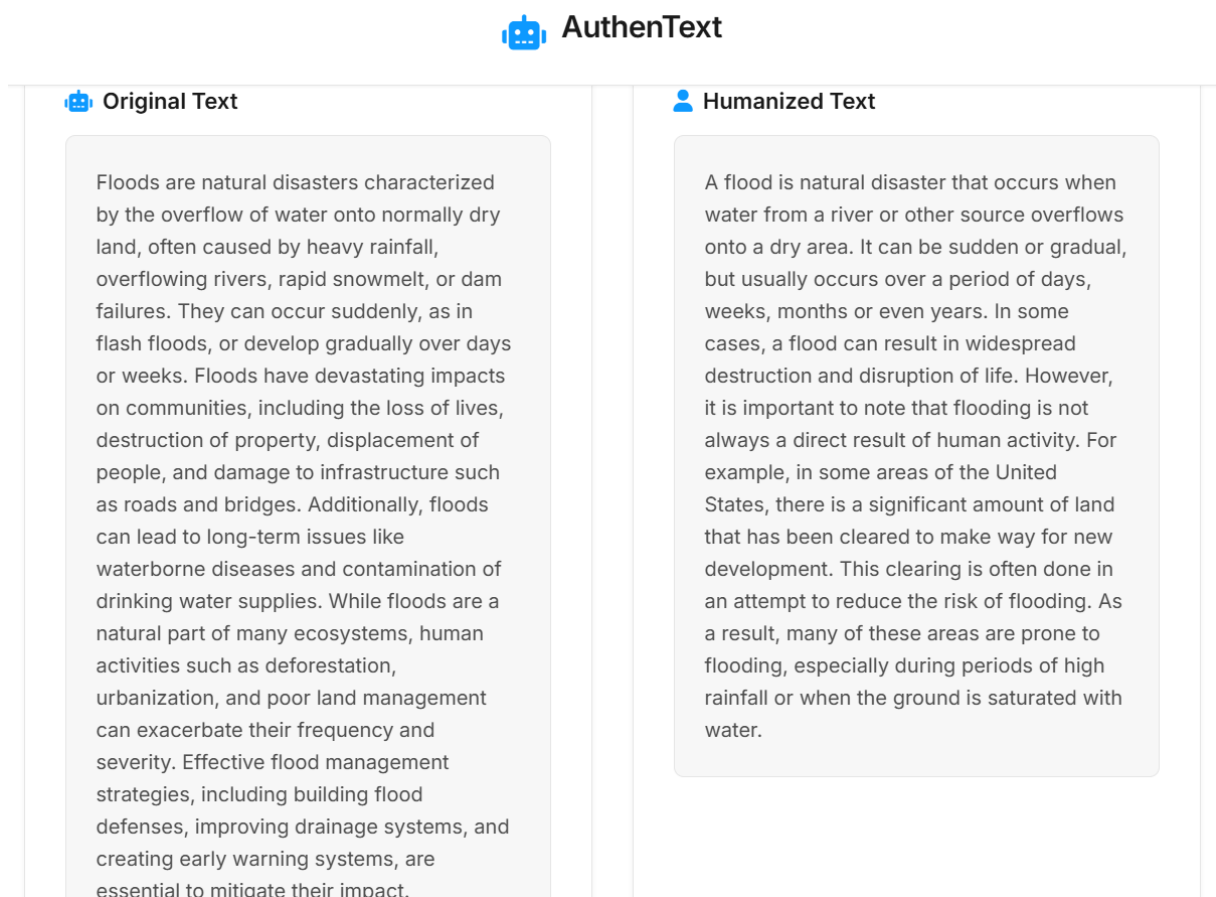


Figure 4.6: User Interface snapshot 5

4.3 Discussion

The outputs of our text classification, explainability, and paraphrasing modules attest to the efficiency of the proposed strategy. Yet, there are some areas where some enhancements can help the system’s performance and usability.

Our present feedforward lightweight neural network has a high accuracy of 95.69%, but it mainly depends on surface level patterns and not profound semantic comprehension. Because it does not inherently examine the contextual word relationships within a sentence, it might have difficulty with more sophisticated linguistic patterns. Adding a transformer-based model would enhance classification accuracy by retaining richer semantic information. Our paraphrasing module based on GPT-2 successfully converts AI-generated text into a more natural form. Nonetheless, to expand its use, we might include several humanizing paraphrasers specifically designed for certain domains like academic writing, business writing, and creative content generation. Through models trained specifically

for various paraphrasing styles, the system could deliver more flexible and context-aware paraphrasing so that AI-produced text is better suited to fit a variety of writing standards.

4.4 Conclusion

This chapter showed the outcome and discussion of our suggested method, including text classification, explainability, and paraphrasing. Our feedforward light neural network was 95.69% accurate, yet does not have deep semantic comprehension, which would be enhanced through transformer models. The LIME explanation module effectively picked out 20 essential features, promoting model transparency. For paraphrasing, although T5 had a tendency to summarize, the pretrained GPT-2 model generated correct paraphrases in sync with input prompts. Enriching this module with domain-based humanizing paraphrasers might make it more versatile. Overall, our system is effective at identifying AI-written text and smoothing out its readability, with potential future improvements centered on semantic comprehension and wider usage.

Chapter 5

Conclusion

5.1 Conclusion

This report fully records all facets of the "Authentext: Detection of AI-Generated Text" project, including its aims, methods, and results. It starts by pointing out the increasing use of AI-generated text and the threat it poses to originality and emotional richness, stressing the necessity for tools that categorize AI-generated material while smoothing it out to look more natural and human. A comprehensive literature review guided our methodology, highlighting most important gaps that guided our design and implementation of the system.

Our system is organized into four major modules: data preprocessing, text classification, explainable AI, and text transformation. The feedforward lightweight neural network-based classification model attained 95.69% accuracy, presenting a trade-off between efficiency and interpretability. Nonetheless, due to its lack of deep semantic understanding, future research may consider transformer-based architectures for better performance. In order to increase transparency, we integrated LIME, which effectively pointed out 20 key features, rendering model predictions more comprehensible. For text paraphrasing, we started by using T5, but it showed tendencies for summarization instead of actual paraphrasing. Our last implementation, utilizing a pre-trained GPT-2 model, provided accurate rewritten results consistent with input prompts. Additional improvements, like integrating domain-specific humanizing paraphrasers, might open up its applications across fields such as academia, journalism, and creative writing.

This project fills the middle ground between AI-generated and human-written text, rendering AI-generated content more authentic and relatable. With applications in journalism, education, creative writing, and customer service, it has the capability to enhance human-AI collaboration while prioritizing transparency and ethical AI usage. Future re-

search will center on improving semantic comprehension, flexibility, and user-friendliness in various settings, helping shape the future of reliable AI-created content.

5.2 Future Scope

The Authentext project effectively labels AI-generated text and improves its human-likeness. Nonetheless, there are a number of areas where improvement and expansion could be made:

- **Enhancing Text Classification with Transformer Models:** Our present feedforward lightweight neural network has very high accuracy (95.69%) but fails to examine semantic relationships between pieces of text. Future versions can include transformer-based architectures (DistilBERT, RoBERTa) to enhance contextual comprehension at the cost of interpretability by using explainable AI methods.
- **Improving Explainability and Trust in AI Decisions:** Although LIME successfully identifies 20 major attributes for interpretability, investigating SHAP (SHapley Additive Explanations) or Integrated Gradients would offer greater insights into the decision-making process of the model. This would further improve transparency and increase user confidence in AI-generated content recognition.
- **Advancing the Paraphrasing Module for Diverse Applications:** The paraphrasing module based on GPT-2 effectively rephrases AI-generated content but can be optimized further by incorporating humanizing paraphrasers trained in various domains. Fine-tuning the models for scholarly writing, journalism, and business content would make the system more generalizable.
- **Developing a Real-time API for Industry Use:** Porting Authentext to a web-based API or browser plugin would allow it to be integrated smoothly with content creation platforms, academic software, and enterprise systems, making it more viable for real-world application.
- **Expanding Dataset and Multilingual Capabilities:** Our model at present targets English text. Augmenting the dataset to multilingual AI-created content would improve worldwide applicability. Deploying multilingual NLP models (e.g., mBERT, XLM-R) would enable detection and tuning across languages.

References

- [1] H. Wang, J. Li, and Z. Li, “Ai-generated text detection and classification based on bert deep learning algorithm,” in *Proceedings of the 2023 International Conference on Artificial Intelligence and Text Processing (AIP 2023)*, Beijing, China, 2023, pp. 1–6.
- [2] K. Ibrahim, “Using ai-based detectors to control ai-assisted plagiarism in esl writing: ‘the terminator versus the machines’,” *Language Testing in Asia*, vol. 13, pp. 1–28, 2023.
- [3] E. Mitchell, Y. Lee, A. Khazatsky, C. D. Manning, and C. Finn, “Detectgpt: zero-shot machine-generated text detection using probability curvature,” in *Proceedings of the 40th International Conference on Machine Learning (ICML)*, Honolulu, HI, USA, 2023.
- [4] Y. Zhang, Q. Leng, M. Zhu, R. Ding, Y. Wu, J. Song, and Y. Gong, “Enhancing text authenticity: a novel hybrid approach for ai-generated text detection,” in *Proceedings of the 2024 IEEE International Conference on Artificial Intelligence (ICAI)*. IEEE, 2024, pp. 115–120.
- [5] Y. Zhou, J. Wen, J. Jia, L. Gao, and Z. Zhang, “C-net: a compression-based lightweight network for machine-generated text detection,” *IEEE Signal Processing Letters*, vol. 31, pp. 1269–1273, 2024.

Appendix A: Presentation



Authentext-Detecting AI Generated Text

Mr. Ajith S
Asst. Professor
Dept of CSE

Hanan Maryam Jamal
Hannah Rachel Abraham
Maanas Krishnan
Meenakshi Saji

1

Problem definition



The growing presence of AI-generated text has raised concerns about its ability to be distinguished from human-written content, particularly in domains where authenticity and personal expression are crucial.

2

Purpose and need



The increasing use of AI-generated text raises concerns about authenticity and emotional resonance in writing, making it essential to distinguish between AI and human-created content. This project addresses the need for a solution that not only identifies AI-generated text but also enhances it, ensuring that it retains the depth and creativity characteristic of human expression.

3

Objective



The primary objective of this project is to develop a robust machine learning system that:

1. **Classifies text** as either AI-generated or human-written, with high accuracy and explainability.
2. **Extracts key features** influencing the classification decision, providing transparency and insights into why the model made its decision.
3. **Transforms AI-generated text** into more human-like forms, ensuring that AI-produced content retains the emotional depth, authenticity, and natural expression found in human writing.

4

Proposed method

1. Text classification:

Develop a machine learning model to classify text as either AI-generated or human-written.

2. Explainable AI:

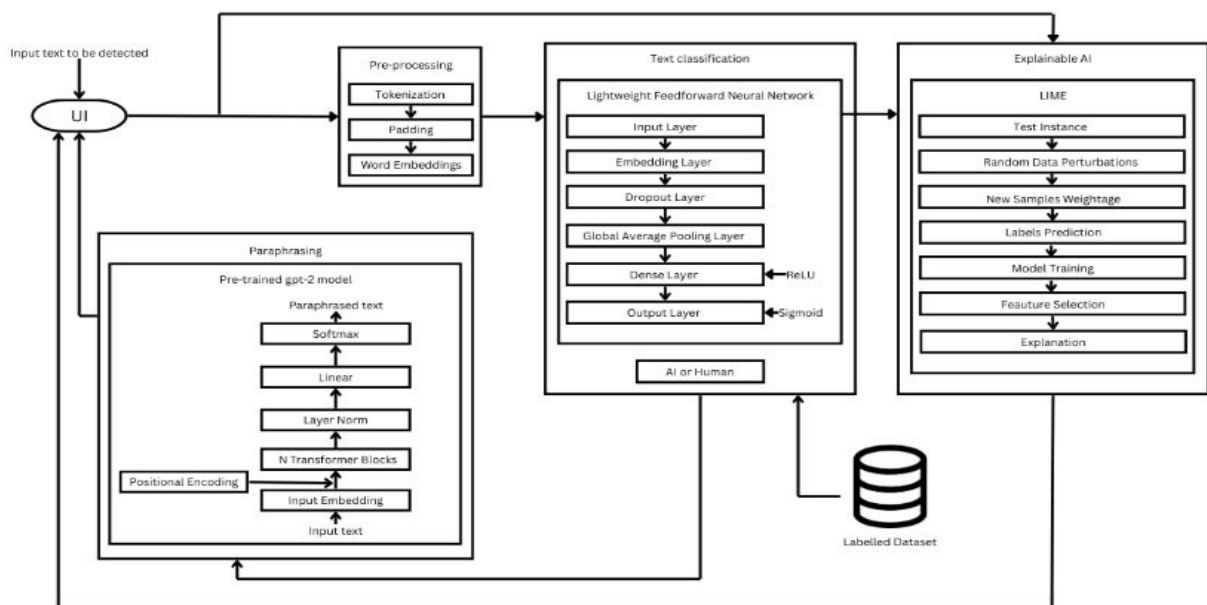
Implement explainable AI techniques to provide insights into the features influencing the model's predictions.

3. Paraphrasing:

Leverage a model to paraphrase and enhance AI-generated text, focusing on adjusting tone, style, and coherence to make it more human-like.

5

Architecture Diagram





Modules in detail

7



Text Classification



- Detect machine-generated text (MGT) by utilizing a **lightweight feedforward neural network**
- The layers in the neural network are:
 - **Embedding Layer**: Transforms word indices into dense vector representations.
 - **Dropout Layer**: Adds regularization by randomly dropping 50% of neurons during training.

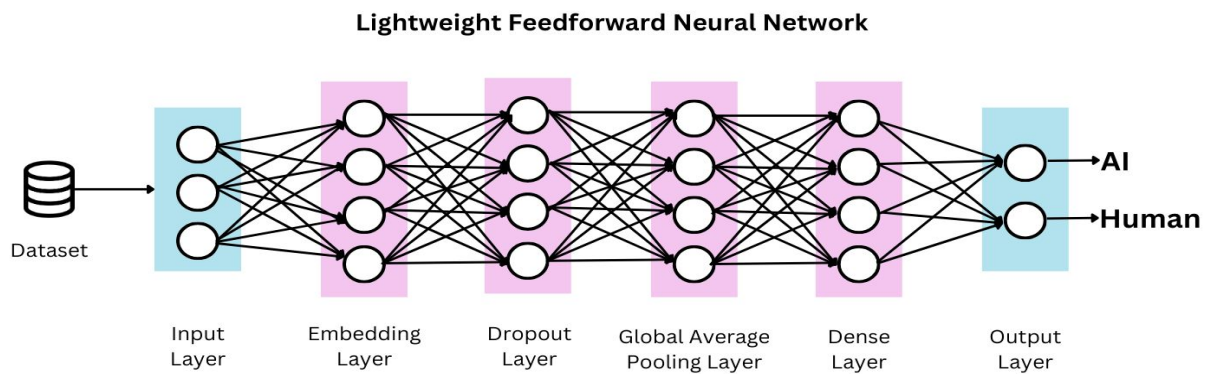
8

Text Classification

- **Global Average Pooling Layer:** Reduces embeddings into a fixed-length vector.
- **Dense Hidden Layer:** Processes features using ReLU activation.
- **Output Layer:** Implements sigmoid activation for binary classification.
- Train the model using the Adam optimizer (learning rate: 0.00045) and binary cross-entropy loss.
- Implement early stopping to monitor validation loss and prevent overfitting.


9

Architecture diagram - Text Classification



10


Comparison - Text classification



Model	Advantages	Disadvantages
Pre-trained BERT	Strong contextual embedding with deep semantic understanding	Resource-intensive, function as "black boxes," making them harder to explain and integrate with XAI, limits flexibility due to its fixed architecture
Compressed lightweight neural network	Fewer layers with reduced neurons and parameters, High accuracy	Struggles with highly diverse text patterns, NCD loss used does not capture differences in text efficiently

11

Comparison - Text classification



Model	Advantages	Disadvantages
Feedforward lightweight neural network	Fewer layers, minimal neurons, and optimized parameters, Highly interpretable, Captures key patterns in text, High accuracy	Lacks deep contextual understanding

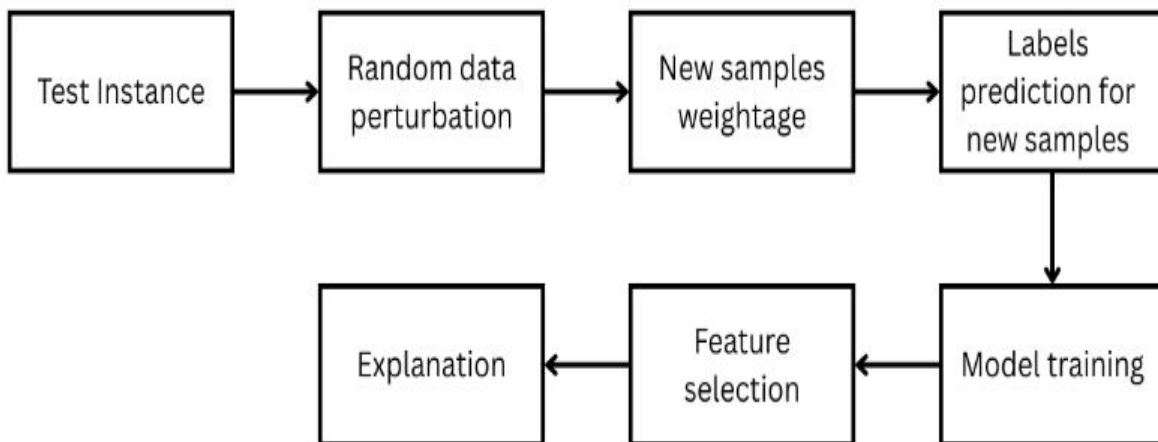
12

Explainable AI-LIME

1. Random Data Perturbation: Generate new samples by altering the features of the original instance.
2. Weight the New Samples: Assign weights based on the similarity of the perturbed samples to the original instance.
3. Train a Weighted, Interpretable Model: Build a simple model to approximate the black-box model's behavior locally.
4. Interpretable Representation: Produce an explanation showing which features influenced the prediction.

13

Architecture Diagram- LIME explainable AI



14

Comparison- Explainable AI models

Model	Advantages	Disadvantages
LIME	-Easy to understand, providing clear and human-friendly explanations -Fast and efficient, making it practical for real-time use	-Can be unstable due to randomness in sampling -No theoretical guarantee of consistency
SHAP	-Provides globally consistent and theoretically grounded explanations -Considers feature interactions	-More complex and requires more computation -Harder to interpret

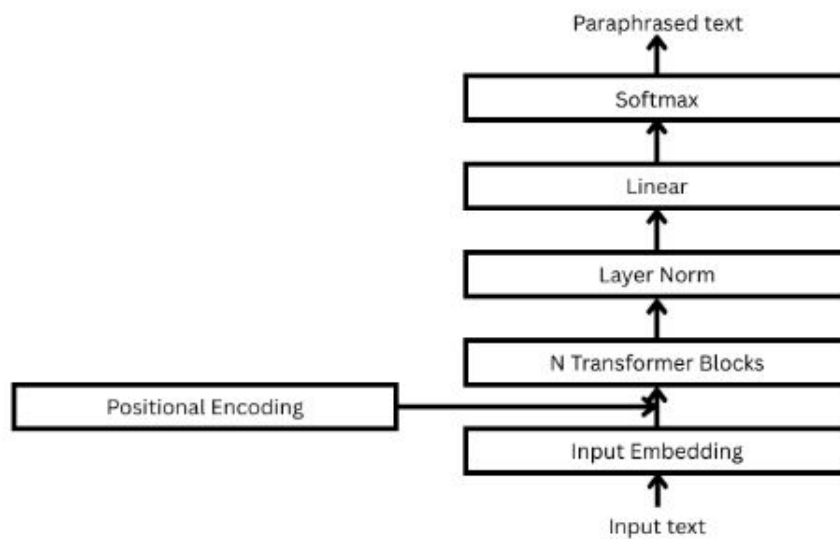
Paraphrasing

Proposed Methodology: GPT-2 for Paraphrasing

The paraphrasing model uses the GPT-2 architecture. It includes:

1. Tokenization of input text using GPT-2 tokenizer.
2. Generating paraphrased text with a controlled decoding strategy.
3. Post-processing to refine output and remove unnecessary prompt text.

Architecture Diagram-Paraphrasing



17

Comparison- Paraphrasing models

Model	Advantages	Disadvantages
GPT-2	Open source model, Easy to fine tune	Not powerful as GPT-4
GPT-4	Highly advanced, accurate responses	Limited number of requests, requires API access
Gemini	Multimodal capabilities, Google-backed	Difficulty in integration
T5 Models	Lightweight, efficient for NLP tasks	Better at summarization than paraphrasing

18

Expected Output

Accurate Classification: A model that reliably distinguishes AI-generated from human-written text.

Text Transformation: A system that refines AI-generated text, making it more human-like.

Enhanced Collaboration: AI-generated content that retains authenticity and emotional impact, enabling better use in creative fields.


19

AI Text Detection Technology

Try Our Technology

Experience the power of our AI detection system firsthand

Floods are natural disasters characterized by the overflow of water onto normally dry land, often caused by heavy rainfall, overflowing rivers, rapid snowmelt, or dam failures. They can occur suddenly, as in flash floods, or develop gradually over days or weeks. Floods have devastating impacts on communities, including the loss of lives, 919/5000 destruction of property, displacement of people, and

 Analyze Text

Why Choose Our Technology?

20

Analysis Results

Detailed analysis of your text with confidence scores and explanations



Analysis Result

This text was likely written by AI

Prediction Probabilities

0.32%

Human

99.68%

AI

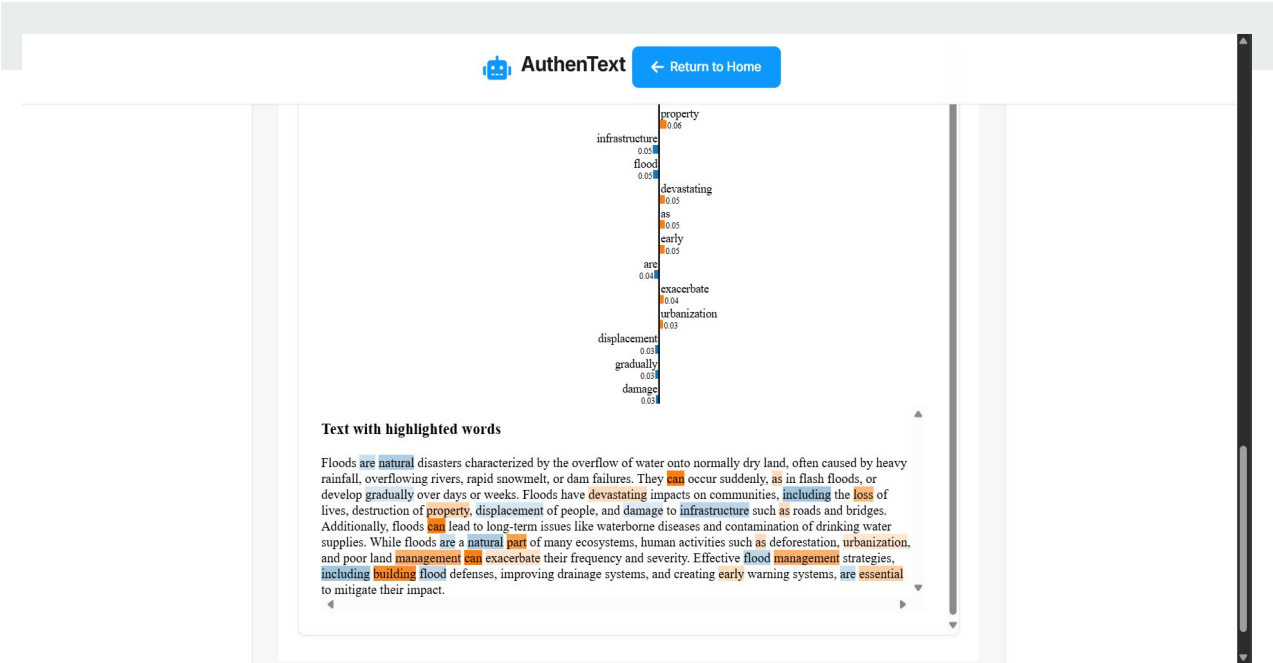
21




Analyzed Text

Floods are natural disasters characterized by the overflow of water onto normally dry land, often caused by heavy rainfall, overflowing rivers, rapid snowmelt, or dam failures. They can occur suddenly, as in flash floods, or develop gradually over days or weeks. Floods have devastating impacts on communities, including the loss of lives, destruction of property, displacement of people, and damage to infrastructure such as roads and bridges. Additionally, floods can lead to long-term issues like waterborne diseases and contamination of drinking water supplies. While floods are a natural part of many ecosystems, human activities such as deforestation, urbanization, and poor land management can exacerbate their frequency and severity. Effective flood management strategies, including building flood defenses, improving drainage systems, and creating early warning systems, are essential to mitigate their impact.

 [Humanize Text](#)

22



<div><div> AuthenText</div></div>		
<div> Original Text</div>	<div><div> Humanized Text</div></div>	
<p>Floods are natural disasters characterized by the overflow of water onto normally dry land, often caused by heavy rainfall, overflowing rivers, rapid snowmelt, or dam failures. They can occur suddenly, as in flash floods, or develop gradually over days or weeks. Floods have devastating impacts on communities, including the loss of lives, destruction of property, displacement of people, and damage to infrastructure such as roads and bridges. Additionally, floods can lead to long-term issues like waterborne diseases and contamination of drinking water supplies. While floods are a natural part of many ecosystems, human activities such as deforestation, urbanization, and poor land management can exacerbate their frequency and severity. Effective flood management strategies, including building flood defenses, improving drainage systems, and creating early warning systems, are essential to mitigate their impact.</p>	<p>A flood is natural disaster that occurs when water from a river or other source overflows onto a dry area. It can be sudden or gradual, but usually occurs over a period of days, weeks, months or even years. In some cases, a flood can result in widespread destruction and disruption of life. However, it is important to note that flooding is not always a direct result of human activity. For example, in some areas of the United States, there is a significant amount of land that has been cleared to make way for new development. This clearing is often done in an attempt to reduce the risk of flooding. As a result, many of these areas are prone to flooding, especially during periods of high rainfall or when the ground is saturated with water.</p>	

Conclusion



This project aims to create a machine learning tool that not only accurately distinguishes between AI-generated and human-written text but also enhances AI-generated content to make it more human-like.

Appendix B: Vision, Mission, Programme Outcomes and Course Outcomes

Appendix B

Vision: To become a Centre of Excellence in Computer Science & Engineering, moulding professionals catering to the research and professional needs of national and international organizations.

Mission: To inspire and nurture students, with up-to-date knowledge in Computer Science & Engineering, Ethics, Team Spirit, Leadership Abilities, Innovation and Creativity to come out with solutions meeting the societal needs.

Program Outcomes:

PO1: Engineering knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

PO2: Problem analysis: Identify, formulate, research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

PO3: Design/development of solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

PO4: Conduct investigations of complex problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

PO5: Modern tool usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modelling to complex engineering activities with an understanding of the limitations.

PO6: The engineer and society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

PO7: Environment and sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

PO8: Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice

PO9: Individual and team work: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings

PO10: Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

PO11: Project management and finance: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

PO12: Life-long learning: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

Program Specific Outcomes:

PSO1: Computer Science Specific Skills: The ability to identify, analyze and design solutions for complex engineering problems in multidisciplinary areas by understanding the core principles and concepts of computer science and thereby engage in national grand challenges.

PSO2: Programming and Software Development Skills: The ability to acquire programming efficiency by designing algorithms and applying standard practices in software project development to deliver quality software products meeting the demands of the industry.

PSO3: Professional Skills: The ability to apply the fundamentals of computer science in competitive research and to develop innovative products to meet the societal needs thereby evolving as an eminent researcher and entrepreneur.

Course Outcomes

CO1: Model and solve real world problems by applying knowledge across domains.

CO2: Develop products, processes, or technologies for sustainable and socially relevant applications.

CO3: Function effectively as an individual and as a leader in diverse teams and to comprehend and execute designated tasks.

CO4: Plan and execute tasks utilizing available resources within timelines, following ethical and professional norms.

CO5: Identify technology/research gaps and propose innovative/creative solutions.

CO6: Organize and communicate technical and scientific findings effectively in written and oral forms.

Appendix C: CO-PO-PSO Mapping

Appendix C

CO-PO AND CO-PSO MAPPING

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2	PSO 3
C O1	2	2	2	1	2	2	2	1	1	1	1	2	3		
C O2	2	2	2		1	3	3	1	1		1	1		2	
C O3									3	2	2	1			3
C O4					2			3	2	2	3	2			3
C O5	2	3	3	1	2							1	3		
C O6					2			2	2	3	1	1			3

3/2/1: high/medium/low