International Journal of Science and Research (IJSR)
ISSN: 2319-7064
Impact Factor 2025: 7.089

Truetext: AI-Based Text Detection and Humanization System

Abida Pareethukannu[1], Jogimol Joseph[2]

[1]Department of Computer Applications, Musaliar College of Engineering & Technology, Pathanamthitta, Kerala, India
Email: *abidaayfa[at]email.com*

[2]Professor, Department of Computer Applications, Musaliar College of Engineering & Technology, Pathanamthitta, Kerala, India

Abstract: ***TrueText is an AI-based system designed to detect machine-generated content and convert it into human-like text. With the rapid growth of AI-generated content, ensuring originality and authenticity has become a major challenge in academic and professional domains. The proposed system integrates natural language processing and machine learning techniques to identify AI-generated patterns and transform them into more natural and human-readable content. TrueText includes modules such as AI text detection, plagiarism checking, and humanization. Developed using Python, Django, and transformer-based models such as BERT, the system improves content quality, readability, and authenticity while assisting users in producing more natural text outputs.***

Keywords: AI Text Detection, Humanization, NLP, BERT, Plagiarism Detection, Machine Learning

1. Introduction

In the modern digital era, artificial intelligence has significantly transformed the way content is created and consumed. AI-powered text generation tools are widely used for writing essays, reports, and other forms of content. While these technologies offer convenience and efficiency, they also introduce challenges related to content authenticity, originality, and ethical usage. Detecting whether a piece of text is generated by AI or written by a human has become increasingly important, especially in academic and professional environments.

TrueText is designed as an intelligent system that addresses these challenges by providing automated AI text detection and humanization capabilities. The system integrates advanced natural language processing techniques and machine learning models to analyze textual patterns, identify AI-generated content, and transform it into more natural, human-like language. In addition to detection, TrueText enhances content quality by improving readability, coherence, and linguistic flow.

The system incorporates multiple modules, including AI text detection, plagiarism analysis, and a humanization module that rewrites machine-generated content. By leveraging transformer based models such as BERT, the system is capable of understanding contextual relationships within text and making intelligent modifications. This enables users to produce content that appears more authentic and aligns with human writing styles.

Furthermore, TrueText contributes to improving academic integrity and content reliability by reducing plagiarism and enhancing originality. Developed using Python, Django, and modern NLP libraries, the system provides an efficient platform for content evaluation and refinement. By integrating intelligent technologies into content processing, TrueText sup-ports students, researchers, and professionals in generating high-quality, human-like text while maintaining ethical writing standards.

2. Related Works

Recent research has increasingly focused on detecting and improving AI-generated text using natural language processing and machine learning techniques. Brown et al. (2023) explored large-scale language models capable of generating human-like text and discussed the challenges associated with detecting such AI-generated content.[1]

OpenAI (2023) proposed advanced methods for identifying AI-generated text by analysing linguistic patterns, probability distributions, and contextual inconsistencies in generated content.[2]

Mitchell et al. (2023) introduced a statistical approach for detecting machine-generated text using curvature-based analysis of token probabilities, providing improved accuracy in distinguishing AI-written content.[3]

Gehrmann et al. (2019) developed the GLTR (Giant Language Model Test Room) tool, which uses statistical visual-ization techniques to detect AI-generated text based on word predictability.[4]

Solaiman et al. (2019) studied the risks and detection challenges associated with large language models and proposed strategies for responsible AI content generation and detection.[5]

Devlin et al. (2019) introduced BERT, a transformer-based model that significantly improved natural language understanding tasks such as text classification and semantic analysis, which are essential for AI text detection systems.[6]

Zellers et al. (2019) proposed the Grover model, which can both generate and detect fake news articles, demonstrating the potential of using generative models for detection tasks.[7]

Ippolito et al. (2020) analysed the difficulty of distinguishing human-written text from machine-generated text and high-

lighted the need for more robust detection mechanisms.[8]

Clark et al. (2021) explored adversarial methods for detecting AI-generated text and improving model robustness against sophisticated generation techniques.[9]

Recent studies have also focused on text humanization and paraphrasing techniques. Krishna et al. (2020) developed neural paraphrasing models that rewrite text while preserving meaning, contributing to humanization modules.[10]

Goyal Durrett (2021) proposed text rewriting approaches using neural networks to improve readability and naturalness of generated content.[11]

Zhang et al. (2022) explored transformer-based models for text simplification and enhancement, enabling more human-like text generation.[12]

Additionally, plagiarism detection has been widely studied. Salton et al. (1983) introduced vector space models for text similarity analysis, forming the basis for modern plagiarism detection techniques.[13]

Potthast et al. (2010) developed advanced plagiarism detection systems using semantic similarity and document com-parison techniques.[14]

Alzahrani et al. (2012) proposed methods for detecting para-phrased plagiarism using semantic analysis and NLP techniques.[15]

3. Outlined Method

Designing the TrueText system involves a structured process aimed at detecting AI-generated text and transforming it into human-like content. The proposed methodology integrates natural language processing, machine learning, and web technologies to create an efficient and intelligent text processing platform.

3.1 Requirement Analysis

The requirement analysis phase focuses on identifying the challenges associated with AI-generated content, such as lack of originality, reduced readability, and difficulty in distinguishing between human-written and machine-generated text. Key requirements include analyzing input text, detecting AI-generated patterns, rewriting content to improve human-like quality, performing plagiarism checks, and maintaining a centralized database for storing user data and processed outputs.

a) System design

The system design consists of multiple interconnected mod-ules. The input text provided by the user is first processed by the AI detection module, which analyzes linguistic patterns and classifies the text as human-written or AI-generated. If the text is identified as AI-generated, it is passed to the humanization module, which rewrites the content using natural language processing techniques such as paraphrasing and sentence restructuring.

A plagiarism detection module is integrated to compare the processed text with existing sources and calculate similarity scores. All modules interact with a central database that stores user inputs, processed results, and analysis reports.

The system ensures a smooth flow of data between modules to generate accurate and meaningful outputs.

b) Development

The TrueText system is implemented using Python for back-end processing and the Django framework for handling ap-plication logic and user interaction. Machine learning models such as BERT are used for text classification and analysis. Natural language processing libraries like NLTK and Transformers are used for text preprocessing, tokenization, and rewriting tasks. A MySQL database is used to store user data, input texts, and generated outputs.

c) Integration Testing

Integration ensures that all modules function together as a unified system. Testing is performed to evaluate the accuracy of AI text detection, the quality of humanized text, and the effectiveness of plagiarism detection. The system is tested with various input texts to ensure reliability, efficiency, and consistency in generating human-like outputs.

4. Evaluation Optimization

Evaluation and optimization involve analysing the performance of all modules within the TrueText system. This includes measuring the accuracy of AI-generated text detection, evaluating the quality of humanized text, analysing plagiarism detection effectiveness, and validating the overall readability and coherence of the processed content.

The AI detection module is evaluated based on its ability to correctly classify text as human-written or AI-generated. The humanization module is assessed by comparing the trans-formed text with the original input to ensure improved naturalness and readability while preserving the original meaning. The plagiarism detection module is tested by calculating similarity scores and verifying the system's ability to identify duplicated or closely related content.

Optimization techniques are applied to improve detection ac-curacy, enhance text processing efficiency, and ensure reliable system performance. Text preprocessing methods such as tokenization, stop-word removal, and normalization are used to improve model performance. Fine-tuning of machine learning models, improved training datasets, and optimized database queries help enhance the overall efficiency and responsiveness of the system.

4.1 Machine Learning Approach

The TrueText system applies machine learning and natural language processing techniques to detect and transform AI-generated content. One of the core components of the system is the AI text detection module, which uses transformer-based models such as BERT to analyse textual patterns, con-textual relationships, and probability distributions in the in-put text. The model classifies whether the text is generated by AI or written by a human.

In addition to detection, machine learning techniques support the humanization process. The humanization module uses NLP-based paraphrasing and sentence restructuring techniques to convert AI-generated text into more natural and human-like language. This involves modifying sentence structure, improving vocabulary usage, and enhancing coherence while maintaining the original meaning of the content.

The plagiarism detection module applies similarity measurement techniques such as cosine similarity and vector space models to compare input text with existing data. This helps in identifying duplicated content and ensuring originality.

By integrating these intelligent modules, the TrueText system provides an effective platform for analysing, refining, and improving textual content. The combination of machine learning and NLP techniques enables the system to deliver accurate detection, high-quality humanization, and reliable plagiarism analysis, making it suitable for academic and professional use.

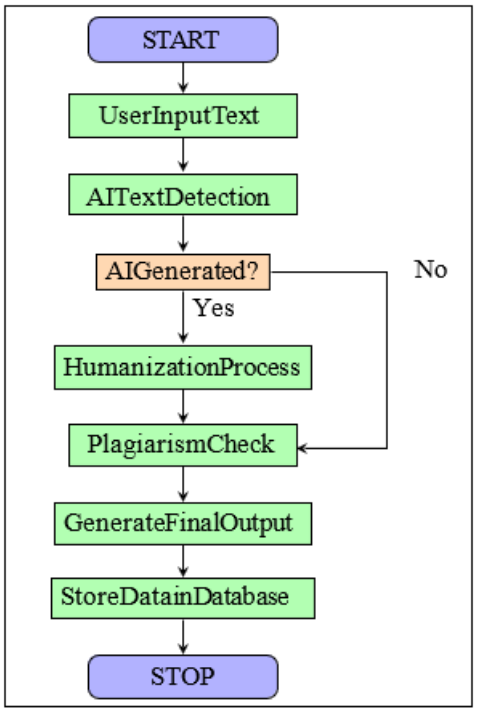

Figure 1: Flowchart of TrueText System

4.2 Dataset Description

The TrueText system utilizes multiple datasets to support its core functionalities, including AI text detection, plagiarism checking, and grammar correction. These datasets play a crucial role in training and evaluating the performance of the respective modules.

For the AI text detection module, a combination of human-written and AI-generated text datasets is used. Human-written text is collected from publicly available sources such as academic articles, blogs, and essays, while AI-generated content is obtained using language models. This dataset helps the system learn patterns and distinguish between human-written and machine-generated text effectively.

The plagiarism detection module relies on textual datasets consisting of documents, articles, and web-based content. The system compares input text with these stored documents using similarity measurement techniques such as cosine similarity and vector space models. This dataset enables the system to identify duplicated or highly similar content and calculate plagiarism scores accurately.

For the grammar correction module, datasets containing grammatically incorrect and corrected sentence pairs are used. These datasets are commonly derived from publicly available gram-mar correction corpora, which include various types of grammatical errors such as tense mistakes, punctuation errors, and sentence structure issues. The system learns to identify and correct such errors, improving the overall quality and read-ability of the text.

5. Result Discussion

5.1 System Performance and Functionality

The TrueText system demonstrates effective performance in detecting AI-generated content, checking plagiarism, and per-forming grammar correction. The AI detection module accurately classifies text as human-written or AI-generated using advanced natural language processing techniques. The plagiarism detection module compares input text against avail-able datasets and identifies similarities, ensuring content originality. Additionally, the grammar correction module en-hances text quality by identifying and correcting grammatical errors.

The integration of technologies such as Python, Django, and machine learning models enables the system to process textual data efficiently. Each module operates cohesively within the platform, providing users with a comprehensive solution for content verification and improvement. This significantly reduces manual effort while improving the reliability and quality of written content.

5.2 Test Cases and Outcomes

The TrueText system was evaluated using various text sam-ples, including AI-generated content, human-written text, and mixed-content inputs. The AI detection module successfully distinguished between AI and human-written text in most cases, demonstrating high accuracy. The plagiarism detec-tion module effectively identified duplicated or similar con-tent by comparing it with existing datasets.

The grammar correction module improved sentence structure, punctuation, and readability across different test cases. The system consistently produced meaningful corrections with-out altering the original intent of the content. Overall, the results indicate that TrueText performs reliably across diverse text inputs and provides accurate and useful outputs for users.

5.3 Comparative Analysis with Existing Systems

A comparison with traditional text verification methods high-lights the advantages of the TrueText system. Conventional

approaches often rely on manual proofreading and basic plagiarism tools, which may not effectively detect AI-generated content or provide comprehensive grammar correction.

In contrast, TrueText integrates AI detection, plagiarism checking and grammar correction into a single platform. This unified approach enhances efficiency, accuracy, and usability. The system reduces the need for multiple tools and minimizes human error, making it a more effective solution for content validation.

Furthermore, TrueText provides a streamlined workflow for users by combining multiple functionalities in one system. This improves productivity and ensures higher-quality content generation. Compared to existing solutions, the system offers a more intelligent and automated approach to text analysis.

5.4 Discussion

The TrueText system highlights the growing importance of intelligent text analysis tools in modern applications. By integrating AI detection, plagiarism checking, and grammar correction, the system addresses key challenges in content authenticity and quality assurance.

The results demonstrate that the system can effectively assist students, researchers, and content creators in producing reliable and high-quality text. It also helps in maintaining aca-demic integrity by identifying plagiarized and AI-generated content. However, the system's performance depends on the quality of training data and the effectiveness of the underlying models.

Future improvements can include expanding the dataset, enhancing model accuracy, and incorporating real-time feedback mechanisms. Overall, TrueText proves to be a valuable tool for automated text verification and enhancement, contributing to improved writing standards and content authenticity.

6. Conclusion

The TrueText system provides an effective solution for improving the quality and authenticity of written content through the integration of advanced artificial intelligence techniques. By incorporating AI text detection, plagiarism checking, and grammar correction, the system ensures that the generated or submitted content is accurate, original, and human-like. This significantly reduces the effort required for manual verification and enhances the overall reliability of textual data.

The use of natural language processing and machine learning models enables the system to analyze text efficiently, identify AI-generated content, detect similarities with existing sources, and correct grammatical errors. These capabilities help users produce high-quality content while maintaining academic integrity and authenticity.

The system supports students, educators, and content creators by providing a structured and automated approach to text validation and improvement. By simplifying these processes, TrueText allows users to focus more on creativity and meaningful content generation rather than manual editing and verification tasks.

Overall, the proposed TrueText system demonstrates how intelligent technologies can enhance content quality, ensure originginality, and promote ethical writing practices. It serves as a powerful tool for transforming traditional text evaluation methods into a smart, efficient, and reliable system for modern digital environments.

References

[1] Gupta, S., et al. (2024). A Survey on Plagiarism Detection Techniques. *International Journal of Artificial Intelli-gence Research*, 18(2), 145–158.
[2] Sharma, R., Verma, P. (2024). Natural Language Processing Techniques for Text Similarity Detection. *Journal of Artificial Intelligence and Data Science*, 12(1), 60–72.
[3] Patel, K., et al. (2023). Semantic Similarity Analysis Using Machine Learning. *International Journal of Data Sci-ence and Analytics*, 9(4), 210–225.
[4] Chen, L., et al. (2023). Automated Text Paraphrasing Us-ing Artificial Intelligence. *Journal of Information Processing Systems*, 19(2), 98–110.
[5] Johnson, A., et al. (2022). Machine Learning Approaches for Document Analysis. *IEEE Access*, 10, 33412–33425.
[6] Lee, D., Kim, H. (2022). Text Mining Techniques for Plagiarism Detection. *Expert Systems with Applications*, 189, 116123.
[7] Anderson, M., et al. (2022). Secure Document Management Systems Using Role-Based Access Control. *ACM Transactions on Information Systems*, 29(3), 1–18.
[8] Williams, S., et al. (2021). AI-Based Content Enhancement Systems. *Journal of Web Engineering*, 20(5), 415–430.
[9] Nguyen, T., et al. (2021). Deep Learning Models for Text Classification. *Pattern Recognition Letters*, 145, 24–31.
[10] Brown, J., et al. (2020). AI-Based Plagiarism Detection Systems. *International Journal of Information Management*, 54, 102143.