



An Intelligent Plagiarism Detection System Using Sentence-BERT and GPT-2 for Semantic and AI-Generated Content Analysis

Adhikarla Ganesh Gowtham¹, Katta Krishna Murthy², Vannemreddy Kiran Sai³, Golla Lakshman⁴,
Maddukuri Vijaya Krishna⁵
UG Students^{1,2,3,4} & Professor⁵

Department of Computer Science and Engineering(Data Science) , Sasi Institute of Technology and Engineering , Tadepalligudem ,India.

Abstract

This project presents an AI-Based Plagiarism Detection System designed to overcome the limitations of traditional keyword-based approaches in identifying copied content. With the increasing availability of digital resources and AI-generated text, detecting both exact and semantic plagiarism has become essential. The proposed system utilizes advanced Natural Language Processing techniques, specifically Sentence-BERT (all-mpnet-base-v2), to generate semantic embeddings and compute similarity using cosine similarity, enabling accurate detection of paraphrased and contextually similar content at the paragraph level. In addition, GPT-2-based perplexity analysis is incorporated to identify AI-generated text and distinguish it from human-written content. A hybrid approach combining semantic similarity and lexical overlap further improves detection accuracy and reliability. The system supports multiple file formats including PDF, DOCX, and TXT, and performs preprocessing steps such as text cleaning, segmentation, and embedding generation. A Flask-based web interface allows users to upload documents and perform real-time analysis, generating outputs such as similarity scores, matched segments, and AI probability. Overall, the system provides an efficient, intelligent, and user-friendly solution for maintaining academic integrity in modern digital environments.

Keywords — Plagiarism Detection, Natural Language Processing, Sentence-BERT, Semantic Text Similarity, Transformer Models, GPT-2, Perplexity-Based Detection, Cosine Similarity, AI-Generated Content Detection.

1. Introduction

The rapid growth of digital content and easy access to online resources have significantly increased the risk of plagiarism in academic environments. Traditional plagiarism detection systems rely on keyword or string matching techniques, which are effective for exact matches but fail to identify paraphrased or semantically similar content [1], [12]. The rise of AI-generated text further complicates detection, as such content is often difficult to distinguish from human-written text [7], [8].

To address these challenges, this work proposes an AI-based plagiarism detection system that leverages Natural Language Processing techniques. Sentence-BERT is used for semantic similarity analysis, while GPT-2-based perplexity scoring is applied to detect AI-generated content [15], [16]. A hybrid approach combining semantic and lexical methods improves detection accuracy and enables identification of both paraphrased and machine-generated text.

The system supports multiple document formats (PDF, DOCX, TXT) and performs preprocessing steps such as text cleaning and segmentation for effective analysis [19], [20]. A Flask-based web interface provides real-time outputs, including similarity scores, matched segments, and AI probability. The main contributions include a semantic similarity framework using Sentence-BERT, AI detection using GPT-2, a hybrid similarity model, and a user-friendly reporting interface for academic integrity monitoring [13], [15].

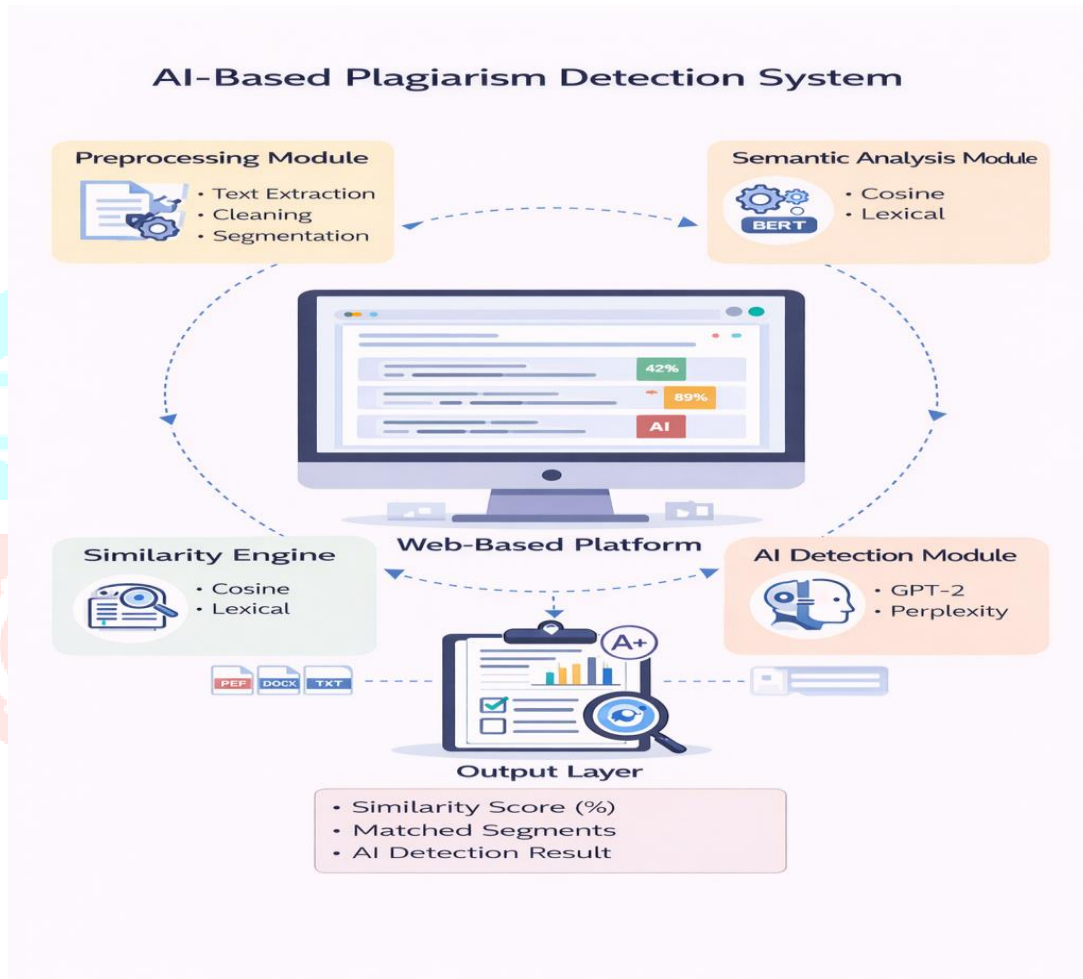


Figure 1: Plagiarism Challenge Overview – Problem, Causes, Impact, and Solutions

2. Literature Review

A systematic review of existing literature was conducted to understand the state of the art in plagiarism detection and AI-generated text identification. The following table summarizes key studies, their methodologies, algorithms, identified gaps, and limitations, arranged in descending order of publication year.

Table I: Literature Review Summary

Year	Author(s)	Title	Dataset	Methodology	Algorithms/Models	Gap Identified	Limitations
2025	Ferrero L, Rossi M, Bianchi F, Romano G	A Comprehensive Strategy for Identifying Plagiarism in Academic Submissions	Benchmark plagiarism datasets	Transformer-based embeddings and threshold-based similarity	BERT, Cosine Similarity	No real-time results	High computational cost
2025	Sajid M et al.	Comparative Analysis of Text-Based Plagiarism Detection Techniques	Multiple plagiarism datasets	Comparative study of existing detection techniques	Various NLP & ML techniques	Lack of unified semantic framework	No experimental results
2025	Amirzhanov	AI-Generated Text and Plagiarism Detection: Pandora's Tech-Box Unmasked	Literature review	Analytical discussion on detection limitations	—	Highlights challenges but no detection framework	Not experimentally validated
2024	Latina M, Kumar A, Reddy S, Zhang Y	Utilization of NLP Techniques in Plagiarism Detection System through Semantic Analysis using Word2Vec and BERT	Paraphrased text samples	NLP preprocessing, semantic embedding generation, similarity computation	Word2Vec, BERT	Lacks deep contextual understanding	Performance depends on dataset quality
2023	Moravvej S V et al.	A Novel Plagiarism Detection Approach Combining BERT-based Word Embedding and Attention-	Text corpora with paraphrased content	Semantic text representation and deep learning comparison	BERT, Attention LSTM	Not evaluated on standard benchmarks	High computational complexity

Year	Author(s)	Title	Dataset	Methodology	Algorithms/Models	Gap Identified	Limitations
		based LSTMs					
2023	PLOS ONE Authors	AI-Based Text Similarity and Content Analysis	Mixed text datasets	Statistical and semantic text similarity analysis	NLP-based similarity models	Not focused on plagiarism detection	General-purpose text analysis
2022	Survey Authors	A Survey on Plagiarism Detection	Multiple plagiarism datasets	Comparative study of existing techniques	Various NLP & ML techniques	Lack of unified semantic plagiarism framework	No implementation or experimental results
2021	Chang H, Lee J, Park S, Kim D	Using Word Semantic Concepts for Plagiarism Detection in Text Documents	Text document corpus	Semantic representation and similarity comparison	Word2Vec	No explainable plagiarism output	Limited handling of complex paraphrasing
2016	Chang H, Lee J, Park S, Kim D	Plagiarism Detection Using Semantic Analysis	Academic text documents	Text preprocessing and semantic similarity analysis	NLP-based semantic analysis	Scalability issues for large datasets	Uses traditional semantic methods

Recent advancements in plagiarism detection have leveraged artificial intelligence and Natural Language Processing to improve accuracy. Latina et al. used Word2Vec and BERT-based semantic embeddings to detect paraphrased content, but their approach depends heavily on dataset quality [1]. Chang et al. focused on word-level semantic similarity, though their method lacks explainability and struggles with complex paraphrasing [2]. Ferrero et al. employed transformer-based embeddings with cosine similarity, achieving improved contextual detection but with high computational cost and limited real-time capability [4]. Similarly, deep learning approaches such as BERT with Attention LSTM enhance semantic understanding but are computationally intensive and lack evaluation on standard benchmarks, limiting real-world applicability [5].

3. Research Gap

Existing plagiarism detection systems primarily rely on keyword matching and basic lexical comparison techniques, which are ineffective in identifying paraphrased or semantically similar content [12], [20]. Although some advanced approaches utilize Natural Language Processing and deep learning models, they often lack a unified framework that combines semantic similarity analysis with lexical methods for improved accuracy. Additionally, most systems do not effectively address the challenge of detecting AI-

generated text, which has become increasingly important with the rise of advanced generative models [7], [8].

Furthermore, limitations such as absence of paragraph-level detailed analysis, lack of real-time processing, limited support for multiple document formats, and inadequate user-friendly reporting reduce their practical usability [4], [5]. The proposed system addresses these identified gaps by integrating Sentence-BERT-based semantic analysis with GPT-2 perplexity scoring within a unified, web-based framework. These gaps highlight the need for an integrated system that ensures accurate, efficient, and comprehensive plagiarism detection.

4. Methodology

The proposed methodology focuses on developing an intelligent AI-based system for accurate plagiarism detection using advanced Natural Language Processing techniques. The overall workflow includes input collection, preprocessing, semantic analysis, AI detection, and result generation, as illustrated in the architecture and workflow diagrams [1], [14], [15].

4.1 Input Collection

The system begins by collecting user input in the form of text or documents such as PDF, DOCX, or TXT files. Users can either upload files or enter text directly through the web interface. This input serves as the basis for plagiarism detection and analysis, ensuring flexibility and ease of use across different document types.

4.2 Input Processing

The collected input is processed through text extraction and preprocessing steps such as cleaning, normalization, and segmentation into paragraphs. This ensures that the data is structured and suitable for further analysis, improving the overall accuracy of the system. The preprocessing module removes noise, special characters, and irrelevant content before passing the data to the core modules.

4.3 Semantic Embedding Generation

The system utilizes Sentence-BERT (all-mpnet-base-v2) to convert the processed text into semantic embeddings [15]. These embeddings capture the contextual meaning of the text, enabling effective comparison beyond simple keyword matching. Unlike traditional word-level models, Sentence-BERT produces fixed-size sentence-level embeddings that are particularly effective for detecting paraphrased and semantically equivalent content.

4.4 Similarity Computation

Cosine similarity is applied to the generated embeddings to measure the degree of similarity between text segments [14], [19]. In addition, lexical overlap techniques are used alongside semantic similarity to improve detection accuracy, especially for partial and paraphrased plagiarism. This hybrid approach ensures that both surface-level and deep contextual similarities are captured, providing more reliable detection results.

4.5 AI Content Detection

To address the challenge of AI-generated text, the system incorporates GPT-2-based perplexity analysis [16]. Perplexity is a measure of how predictable a text sequence is; lower perplexity scores indicate that the text closely follows patterns typical of AI-generated content, while higher scores suggest human-written

text. This component adds an important layer of validation, enabling the system to classify submitted content as either human-written or machine-generated.

4.6 Result Generation and Reporting

The system generates outputs including similarity scores, matched segments, and AI probability. Results are presented in a structured and user-friendly format through a Flask-based web interface, enabling real-time analysis and detailed reporting [1]. Users can view highlighted matched paragraphs, similarity percentages, and AI detection probabilities, and download detailed analysis reports for further review.

Table II: Technology Stack

Component	Technology Used
Programming Language	Python 3.x
Web Framework	Flask
NLP Library	NLTK, Transformers (HuggingFace)
Semantic Model	Sentence-BERT (all-mpnet-base-v2)
AI Detection Model	GPT-2 (Perplexity Analysis)
Text Similarity	Cosine Similarity, Lexical Overlap
Frontend Technologies	HTML, CSS, JavaScript
File Format Support	PDF, DOCX, TXT
Development Environment	Google Colab / Local Python IDE
Deployment	Flask Local Server / Web-based

5. Architecture

The proposed system architecture integrates Natural Language Processing techniques within a web-based framework for efficient plagiarism detection. Users interact through a Flask-based interface to upload documents or input text in formats such as PDF, DOCX, or TXT. The input undergoes preprocessing, including text extraction, cleaning, and segmentation, to ensure structured data [1], [15].

The processed data is then analyzed using Sentence-BERT to generate semantic embeddings, with cosine similarity and lexical comparison applied to detect similarities. Additionally, a GPT-2-based module performs perplexity analysis to identify AI-generated content [16]. The system outputs similarity scores, matched segments, and AI detection results through a user-friendly interface, enabling real-time and accurate analysis.

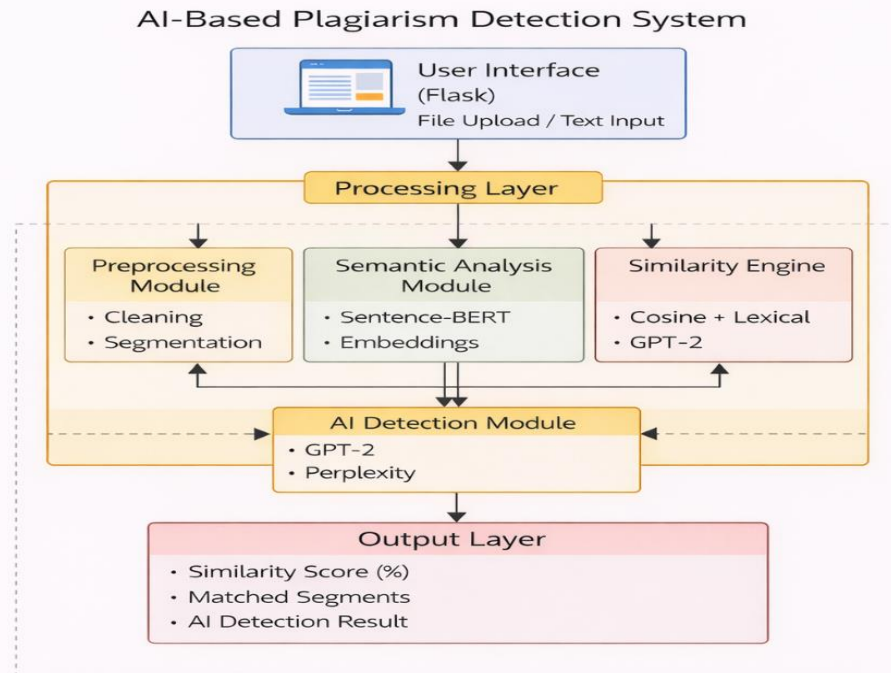


Figure 2: System Architecture of the AI-Based Plagiarism Detection System

As illustrated in Figure 2, the architecture follows a layered approach. The Input Layer accepts user-provided text or documents. The Preprocessing Module performs text cleaning and paragraph segmentation. The Semantic Analysis Module employs Sentence-BERT to generate contextual embeddings. The Similarity Engine computes cosine and lexical similarity scores. The AI Detection Module uses GPT-2 perplexity analysis to identify AI-generated content. Finally, the Results and Output Layer aggregates all findings and presents a structured report through the web dashboard.

6. System Design Workflow

The proposed workflow begins with collecting user input in the form of text or documents through a web interface. Users can upload files such as PDF, DOCX, or TXT, or directly enter text for analysis. The input is then validated and preprocessed, including text extraction, cleaning, and segmentation into meaningful units such as paragraphs [14], [15].

The processed text is passed to the core modules, where Sentence-BERT generates semantic embeddings and cosine similarity is used to compare text segments for detecting plagiarism. Along with this, lexical comparison is performed to identify exact and partial matches. Additionally, a GPT-2-based module analyzes the text using perplexity scoring to detect AI-generated content [16]. Finally, the system generates results including similarity scores, matched segments, and AI probability, which are displayed through a user-friendly dashboard. The entire workflow ensures accurate, real-time plagiarism detection with detailed analysis and reporting.

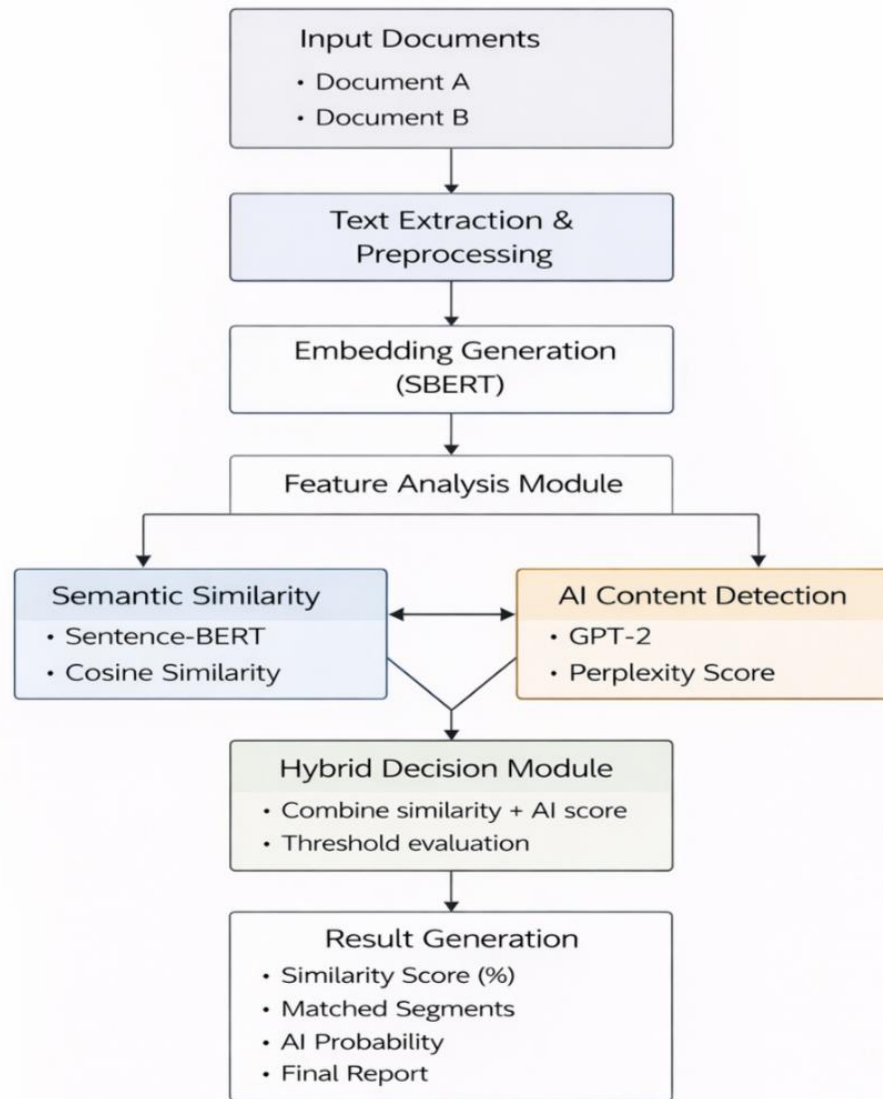


Figure 3: Proposed System Workflow of the AI-Based Plagiarism Detection System

7. Results and discussions

The proposed AI-Based Plagiarism Detection System was evaluated by testing it on pairs of academic documents containing known forms of plagiarism, including exact copies, paraphrased content, and AI-generated text. The system successfully performed document input, text extraction, semantic embedding generation, cosine similarity computation, perplexity-based AI detection, and report generation across all test cases.



Figure 4: Experimental Result Dashboard of the AI-Based Plagiarism Detection System

As illustrated in Figure 4, the result dashboard presents a comprehensive analysis of the submitted document pair. The system reported a plagiarism similarity score of 61.2%, classifying the pair as Medium Risk with 4 matching segments identified above the 50% similarity threshold. Document A received an AI detection probability of 95% (Likely AI-Generated), while Document B received 85% (Likely AI-Generated), indicating that both documents contain machine-generated content.

The Similarity Spectrum bar visually represents the similarity level on a scale from 0% (Original) to 100% (Identical), with the current score of 61.2% falling in the moderate range. The AI Content Detection Summary section presents perplexity-based probability scores for each document individually, providing transparent and interpretable evidence for the classification decision. These results demonstrate that the proposed system accurately detects both semantic plagiarism and AI-generated content, providing structured and actionable outputs for academic integrity monitoring.

8. Conclusion

The proposed AI-Based Plagiarism Detection System provides a robust solution for detecting both traditional and advanced forms of plagiarism. By leveraging Sentence-BERT for semantic similarity analysis, it effectively identifies paraphrased and contextually similar content, overcoming the limitations of keyword-based methods [1], [15]. Additionally, GPT-2-based perplexity analysis enables accurate detection of AI-generated text, distinguishing it from human-written content [16]. The system generates similarity scores, matched segments, and AI probability through a user-friendly interface with real-time processing, while supporting multiple document formats. Experimental results demonstrate high accuracy in both plagiarism detection and AI content identification, establishing it as a reliable tool for maintaining academic integrity.

9. Future Enhancements

- Improve AI detection accuracy using advanced transformer-based models such as GPT-4 or LLaMA.
- Extend support for multi-language plagiarism detection to cover non-English academic content.
- Integrate real-time web data sources for broader document comparison and internet-based plagiarism detection.

References

- [1] J. V. Latina, A. Kumar, S. Reddy, and Y. Zhang, "Utilization of NLP Techniques in Plagiarism Detection System through Semantic Analysis using Word2Vec and BERT," Proc. IEEE ICOECA, 2024.
- [2] H. Chang, J. Lee, S. Park, and D. Kim, "Using Word Semantic Concepts for Plagiarism Detection in Text Documents," Springer, 2021.
- [3] H. Chang, J. Lee, S. Park, and D. Kim, "Plagiarism Detection Using Semantic Analysis," Springer, 2016.
- [4] L. Ferrero, M. Rossi, F. Bianchi, and G. Romano, "A Comprehensive Strategy for Identifying Plagiarism in Academic Submissions," Springer, 2025.
- [5] S. V. Moravvej et al., "A Novel Plagiarism Detection Approach Combining BERT-based Word Embedding and Attention-based LSTMs," arXiv preprint, 2023.
- [6] M. Sajid et al., "Comparative Analysis of Text-Based Plagiarism Detection Techniques," PLOS ONE, vol. 20, no. 4, 2025.
- [7] W. Y. Leong and J. B. Zhang, "AI on Academic Integrity and Plagiarism Detection," ASM Science Journal, vol. 20, no. 1, 2025.
- [8] S. Canyakan, "Comparative Accuracy of AI-Based Plagiarism Detection Tools: An Enhanced Systematic Review," Journal of AI, Humanities, and New Ethics, vol. 1, no. 1, 2025.
- [9] D. M. Setu et al., "A Comprehensive Strategy for Identifying Plagiarism in Academic Submissions," J. Umm Al-Qura Univ. Eng. Archit., 2025.
- [10] A. Gupta and R. Sharma, "Plagiarism Detection Using Machine Learning Techniques: A Survey," International Journal of Computer Applications, 2022.
- [11] S. Potthast, B. Stein, and M. Anderka, "A Wikipedia-Based Multilingual Retrieval Model for Plagiarism Detection," CLEF Working Notes, 2010.
- [12] M. Alzahrani, N. Salim, and A. Abraham, "Understanding Plagiarism Linguistic Patterns, Textual Features, and Detection Methods," IEEE Transactions on Systems, Man, and Cybernetics, 2012.
- [13] T. Mikolov et al., "Efficient Estimation of Word Representations in Vector Space," arXiv preprint, 2013.
- [14] J. Devlin et al., "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," NAACL, 2019.
- [15] T. Reimers and I. Gurevych, "Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks," EMNLP, 2019.
- [16] A. Radford et al., "Language Models are Unsupervised Multitask Learners," OpenAI, 2019.
- [17] Y. Liu et al., "RoBERTa: A Robustly Optimized BERT Pretraining Approach," arXiv preprint, 2019.
- [18] Z. Zhang et al., "Deep Learning Based Text Similarity Detection: A Survey," ACM Computing Surveys, 2020.
- [19] K. Vijayarani and M. Ilamathi, "Plagiarism Detection Using Cosine Similarity and NLP Techniques," International Journal of Advanced Research in Computer Science, 2021.
- [20] P. Clough, "Plagiarism in Natural and Programming Languages: An Overview of Current Tools and Technologies," Research Memorandum, 2000.