IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

Information Extraction: Plagiarism Detection Between Text

[1]Abhishek Kumar[st] , [2]Sanoj Kumar[nd] , [3]Prince Kumar[rd]

[1]B-Tech CSE[st], [2]B-Tech CSE[nd] , [3]B-Tech CSE[rd]

[1]Department of Computer Science & Engineering[st]

[1]Centurion University of Technology and Management, Paralakhemundi[st], Odisha, India

Abstract: Academic plagiarism detection is a critical challenge in educational and research institutions. Existing automated systems suffer from three major limitations: (1) whole-document comparison produces artificially low similarity scores (~0.3 combined), making any fixed threshold ineffective; (2) hardcoded detection thresholds (typically 0.7) are never triggered at these score levels; and (3) systems provide only binary YES/NO verdicts with no passage-level evidence. This paper presents a novel plagiarism detection system trained on the PAN 2011 External Detection Corpus of 11,093 source documents. Our approach introduces three key innovations: overlapping 200-word chunk segmentation, a three-metric similarity fusion combining semantic embeddings (all-MiniLM-L6-v2, weight 0.5), TF-IDF cosine similarity (weight 0.3), and Longest Common Subsequence analysis (weight 0.2), and a data-driven threshold calibrated as mean + 1.5 x standard deviation of the actual score distribution. Additionally, an Agentic AI layer powered by Llama 3.3 70B via Groq API implements a ReAct reasoning loop with 10 specialised tools for natural-language interaction. Experimental evaluation on PAN 2011 suspicious documents demonstrates 86% detection rate compared to 0% baseline, with combined similarity scores of 0.65–0.92 for confirmed plagiarism cases.

Index Terms **—** Plagiarism Detection; Sentence Transformers; Chunk-Based Similarity; Auto-Calibrated Threshold; Agentic AI; ReAct Framework; PAN 2011 Corpus; TF-IDF; LCS; Natural Language Processing

I. INTRODUCTION

Plagiarism — the act of using another person's work without proper attribution — represents a significant ethical and academic integrity concern. With the rapid growth of digital content, distinguishing between independently written text and copied or paraphrased content has become increasingly difficult for both human reviewers and automated systems.

Traditional plagiarism detection tools compare entire documents against reference collections, a strategy that fundamentally underperforms when plagiarism occurs in isolated paragraphs. A 5,000-word essay containing only 300 words of copied text will exhibit a document-level similarity score close to 0.3, far below any commonly used detection threshold.

The PAN shared task series [1] has established the benchmark for plagiarism detection evaluation, providing the PAN 2011 External Detection Corpus of 11,093 source documents and 2,000+ annotated suspicious documents. This corpus remains the standard benchmark for external plagiarism detection research.

This paper identifies three fundamental flaws in existing systems and proposes targeted solutions: (1) Whole-document comparison dilutes the similarity signal — we address this with chunk-based analysis; (2) Hardcoded thresholds fail across different document distributions — we address this with auto-calibration; (3) Binary verdicts lack actionable evidence — we address this with passage-level reporting and an Agentic AI explanation layer.

Our contributions are summarised as: (a) Chunk-based segmentation using 200-word overlapping windows with 50-word stride; (b) Tri-metric similarity fusion with learned weights; (c) Data-driven threshold calibration; (d) ReAct-based Agentic AI interface with 10 specialised tools.

II. RELATED WORK

Plagiarism detection research has evolved across multiple paradigms. Early systems employed character n-gram fingerprinting using Winnowing [2], which generates compact document signatures via rolling hash functions. While computationally efficient, fingerprint-based methods detect only verbatim copying.

Potthast et al. [1] formalised the PAN evaluation framework and distinguished three plagiarism types: verbatim, paraphrase, and summary. Their analysis showed that character overlap features, while effective for verbatim cases, achieve near random performance on paraphrase detection.

The advent of neural sentence embeddings transformed semantic similarity measurement. Reimers and Gurevych [3] introduced Sentence-BERT, using siamese BERT networks trained on NLI and STS datasets. The distilled all-MiniLM-L6-v2 variant produces 384-dimensional embeddings with 80% of BERT's performance at 5x the speed, making it ideal for large-scale corpus comparison.

Large Language Model agents have recently demonstrated capability in multi-step reasoning tasks. Yao et al. [4] proposed ReAct, which interleaves chain-of-thought reasoning and environment interaction in a unified loop. Building on this, we implement a plagiarism-specific agent with specialised tool calls.

III. SYSTEM ARCHITECTURE

The proposed system consists of two tightly integrated components operating in a pipeline architecture, as illustrated conceptually below:

User Query → PlagiarismAgent [Llama 3.3 — ReAct Loop] → (10 tool calls available) → PANPlagiarismDetector → Chunk Splitter → Embedding Engine → Score Fusion (SEM + COS + LCS) → Threshold Calibration → Report

A. PANPlagiarismDetector

The core detection engine manages: (1) corpus loading from PAN 2011 directory structure (part1–part22, 11,093 documents); (2) batch embedding generation using all-MiniLM-L6-v2 with output serialised to disk; (3) chunk-based query processing; (4) three-metric similarity computation; (5) threshold calibration; and (6) passage-level report generation with word-position indices.

B. PlagiarismAgent

The agent layer uses Llama 3.3 70B via Groq API in a ReAct [4] loop. At each iteration: the model receives full conversation history plus tool results, reasons about the next step, emits a structured TOOL_CALL directive, and the dispatcher executes the corresponding tool. The loop terminates when the finish tool is called or the maximum iteration limit (14) is reached.

IV. METHODOLOGY

A. Dataset

The PAN 2011 External Detection Corpus [1] is downloaded from Zenodo (DOI: 10.5281/zenodo.3250095). After RAR extraction, source documents are loaded from the external-detection-corpus/source-document directory, yielding 11,093 text and XML files. Suspicious documents (2,000+) are loaded from the suspicious-document directory for evaluation.

B. Embedding Generation

All source documents are encoded using all-MiniLM-L6-v2 [3], producing 384-dimensional L2-normalised dense vectors. Encoding is batched (batch_size=64) with text truncated to 512 tokens. The full embedding matrix E ∈ ℝ^(11093x384) is serialised to disk via Pickle for reuse without recomputation on subsequent runs.

C. Chunk Segmentation

A suspicious document D of W words is segmented into overlapping chunks $C = \{c_1, c_2, ..., c_n\}$ using: $n = \lceil (W - 50) / 150 \rceil$, stride = 150 words. Each chunk spans 200 words with 50-word overlap into the preceding chunk, ensuring no cross-boundary plagiarism is missed. For a 1,000-word document, this yields approximately 14 chunks.

D. Similarity Fusion

For chunk ce against source document s_j, three scores are computed:

Semantic Score (SEM): Cosine similarity between L2-normalised embeddings ee and E_j, capturing paraphrase-level semantic equivalence. $SEM(ce, s_j) = ee \cdot E_j / (\|ee\| \|E_j\|)$

TF-IDF Cosine Score (COS): Cosine similarity of TF-IDF vectors from Scikit-learn TfidfVectorizer with English stop-word removal and L2 normalisation, capturing keyword-level overlap.

LCS Score: Character-level similarity ratio from Python SequenceMatcher ($2 \cdot M / T$ where M = matched characters, T = total characters), capturing direct copy-paste patterns.

The fused combined score is: $Combined = 0.5 \cdot SEM + 0.3 \cdot COS + 0.2 \cdot LCS$. The document-level score retains the best chunk: $Score(D, s_j) = \max_i Combined(ce, s_j)$

E. Auto-Calibrated Threshold

The detection threshold τ is computed dynamically from the score distribution of all N source documents rather than using a fixed value: $\tau = clip(\mu + 1.5 \cdot \sigma, 0.40, 0.85)$ where μ and σ are the mean and standard deviation of all combined scores. A source document s_j is flagged as plagiarised if $Score(D, s_j) > \tau$. The clip bounds (0.40, 0.85) prevent degenerate thresholds in edge-case distributions.

F. Agent Tools

The PlagiarismAgent exposes 10 tools to the LLM: check_plagiarism (run chunk-based detection on file), explain_match (extract overlapping phrases), compare_passages (sentence-level side-by-side comparison), generate_report (produce formatted detection report), plot_results (bar chart of similarity scores), train_model (train detector on PAN corpus), load_model (deserialise saved detector model), model_status (query training status), show_pan_samples (list real corpus test files), finish (conclude with user summary).

V. EXPERIMENTAL RESULTS

We evaluated the system on 50 randomly selected suspicious documents from the PAN 2011 corpus with confirmed plagiarism annotations. Table I compares the baseline (whole-document, fixed threshold) against our proposed approach.

TABLE I: PERFORMANCE COMPARISON

Metric	Baseline	Proposed
Avg. top score	0.31	0.79
Threshold	0.70 (fixed)	0.48 (auto)
Detection rate	0%	86%
False positive	—	7%
Passage located	No	Yes
LLM explanation	No	Yes

VI. DISCUSSION

The fundamental problem with whole-document comparison is that plagiarism is a localised phenomenon. A 5,000-word document containing 300 plagiarised words exhibits a document-level similarity score of approximately 0.3 because the similarity signal is averaged across all word positions. Our chunk-based approach concentrates the comparison to the specific region of copying, amplifying the similarity signal by a factor proportional to (document_length / chunk_length). For the above example this is approximately 25x, which explains why our scores reach 0.79 on average compared to the baseline's 0.31.

The auto-calibrated threshold addresses distribution shift between different subject domains. A physics essay compared against a general corpus produces systematically lower baseline scores than a history essay, because domain-specific terminology has lower overlap with the general vocabulary distribution. A fixed threshold is inappropriate for both. Our calibration formula $\tau = \mu + 1.5\sigma$ adapts to each query independently, maintaining a consistent false positive rate regardless of domain.

The Agentic AI layer introduces a paradigm shift from passive batch processing to interactive investigation. The LLM reasons about which tools to call and in what order, adapts its strategy based on intermediate results, and provides a natural language explanation of findings. This is particularly valuable for academic administrators who may not have programming expertise.

The 7% false positive rate arises primarily from legitimately similar documents in the same domain — for example, two documents both citing the same historical event in similar phrasing. Future work will incorporate citation-aware filtering to suppress known-source overlaps.

VII. CONCLUSION

This paper presented a chunk-based plagiarism detection system that addresses three fundamental limitations of whole-document comparison approaches. The proposed system achieves an 86% detection rate on PAN 2011 evaluation data — compared to 0% for the baseline — through a combination of overlapping window segmentation, three-metric similarity fusion, and auto-calibrated threshold estimation.

The integration of an Agentic AI layer using the ReAct framework and Llama 3.3 70B enables natural language interaction, making sophisticated plagiarism analysis accessible to non-technical users while providing detailed, passage-level evidence for each detected case.

Directions for future work include: (1) Cross-lingual detection using multilingual sentence transformers. (2) Real-time web document crawling to extend the reference corpus. (3) Fine-tuning sentence transformers on academic domain corpora. (4) Integration with institutional Learning Management Systems. (5) Obfuscation-resistant detection for back-translation and GPT-paraphrase cases.

REFERENCES

[1] M. Potthast, B. Stein, A. Barron-Cedeno, and P. Rosso, "An Evaluation Framework for Plagiarism Detection," in Proc. 23rd Int. Conf. on Computational Linguistics (COLING 2010), pp. 997–1005, 2010.

[2] S. Schleimer, D. S. Wilkerson, and A. Aiken, "Winnowing: Local Algorithms for Document Fingerprinting," in Proc. ACM SIGMOD Int. Conf. on Management of Data, pp. 76–85, 2003.

[3] N. Reimers and I. Gurevych, "Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks," in Proc. 2019 Conf. on Empirical Methods in NLP (EMNLP), pp. 3982–3992, 2019.

[4] S. Yao, J. Zhao, D. Yu, N. Du, I. Shafran, K. Narasimhan, and Y. Cao, "ReAct: Synergizing Reasoning and Acting in Language Models," in Proc. Int. Conf. on Learning Representations (ICLR), 2023.

[5] F. Pedregosa et al., "Scikit-learn: Machine Learning in Python," J. Machine Learning Research, vol. 12, pp. 2825–2830, 2011.

[6] M. Potthast, M. Hagen, T. Gollub, M. Tippmann, J. Kiesel, P. Rosso, E. Stamatatos, and B. Stein, "Overview of the 5th International Competition on Plagiarism Detection," in CLEF (Online Working Notes/Labs/Workshop), 2013.

[7] Meta AI, "Llama 3: Open Foundation and Fine-Tuned Chat Models," Technical Report, Meta AI Research, 2024. [Online]. Available: https://ai.meta.com/llama/

[8] Groq Inc., "Groq LPU Inference Engine," 2024. [Online]. Available: https://console.groq.com

[9] S. Bird, E. Klein, and E. Loper, "Natural Language Processing with Python," O'Reilly Media, 2009.

[10] PAN 2011 Plagiarism Detection Corpus, Zenodo, DOI: 10.5281/zenodo.3250095, 2019.

[11] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," in Proc. NAACL-HLT, pp. 4171–4186, 2019.

[12] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient Estimation of Word Representations in Vector Space," in Proc. ICLR Workshop, 2013.