Authintic: A Three-Layer Hybrid AI System for Plagiarism Detection with Educational Guidance

Prathamesh Mohite[1], Harsh Pardeshi[2], Viraj Kamble[3], Jay Patil[4], Prof. Seema Mishra[5]
[1,2,3,4,5] *Department of Electronics and Computer Science Engineering, MES's Pillai College of Engineering Navi Mumbai, Maharashtra 410206, India*

***Abstract*—The rapid proliferation of Large Language Models (LLMs) such as ChatGPT, Llama, and DeepSeek has created unprecedented challenges in academic integrity. Traditional plagiarism detection tools, which rely primarily on lexical matching, fail to identify semantically paraphrased or AI-generated content. This paper presents Authintic, a full-stack web-based plagiarism detection system that employs a novel three-layer hybrid detection pipeline combining Term Frequency–Inverse Document Frequency (TF-IDF) cosine similarity, FAISS-indexed sentence-level semantic search using Sentence-Transformers, and a fine-tuned BERT binary classifier trained on the PAN 2025 Generative Plagiarism Detection dataset. The system classifies input text at the sentence level into four categories: Direct Match, Paraphrased, AI-Paraphrased, and Original. Additionally, Authintic integrates a Google Gemini-powered educational guidance engine that teaches students how to fix plagiarism rather than rewriting text for them. Evaluated on a balanced PAN25 test set with document-level split integrity, the BERT classifier achieves a Precision of 0.997, Recall of 0.990, F1-Score of 0.9935, and Accuracy of 99.35%, with only 3 false positives and 10 false negatives out of 2,000 test samples.**

***Index Terms*—Plagiarism Detection, Hybrid Detection, TF-IDF, Sentence Embeddings, FAISS, BERT Fine-tuning, AI-Generated Content Detection, Academic Integrity**

I. INTRODUCTION

Plagiarism in academic writing has long been a concern for educational institutions, but the emergence of generative AI tools has fundamentally transformed the landscape of academic dishonesty. Large Language Models (LLMs) such as ChatGPT, Gemini, and Claude can now produce human-like text that is difficult to distinguish from original student writing [1]. A 2024 study found that a significant percentage of students utilize generative AI tools for assignments, from explaining concepts to generating entire research papers, blurring the lines of ethical use [2]. This new class of "AI-paraphrased" plagiarism—where source material is fed into an LLM and semantically rewritten while preserving the core meaning—renders traditional keyword-matching tools largely ineffective.

Existing plagiarism detection systems, including widely adopted commercial platforms such as Turnitin and Copyleaks, primarily rely on lexical similarity techniques such as n-gram fingerprinting and string matching [3]. While effective at identifying verbatim copies, these approaches fail to detect content that has been intelligently restructured, paraphrased using synonyms, or semantically reworded by AI models. Furthermore, current AI content detectors suffer from high false-positive rates, particularly for non-native English speakers, and provide no educational value—they flag content without guiding students toward authentic writing practices [4].

The PAN series of shared tasks on plagiarism detection has been the primary benchmark in this domain since 2009. The PAN 2025 shared task introduced a novel "Generative Plagiarism Detection" challenge, which specifically targets LLM-generated paraphrased plagiarism using content generated by Llama, DeepSeek-R1, and Mistral models [5]. This dataset represents the most current and challenging benchmark for evaluating plagiarism detection systems against AI-assisted academic dishonesty.

Recent research has demonstrated that hybrid approaches combining multiple detection layers outperform single-method systems. A lexical-semantic hybrid system leveraging TF-IDF with Sentence-BERT embeddings demonstrated superior

performance by consolidating various plagiarism cases, including obfuscation levels, into a unified detection framework [6]. Similarly, hierarchical approaches integrating multi-model semantic representations with traditional TF-IDF features and fine-tuned BERT classifiers have shown enhanced detection of generative plagiarism [7].

This paper presents Authintic, a three-layer hybrid plagiarism detection system that addresses the limitations of existing approaches. The key contributions of this paper are:

A cascading three-layer detection pipeline that combines TF-IDF cosine similarity (Layer 1) for fast lexical matching, FAISS-indexed Sentence-Transformer embeddings (Layer 2) for semantic similarity detection, and a fine-tuned BERT binary classifier (Layer 3) for AI-paraphrased content identification.

A sentence-level classification system that categorizes each sentence into Direct Match, Paraphrased, AI-Paraphrased, or Original, providing granular visibility into the document's originality.

An AI-powered educational guidance engine built on Google Gemini that generates personalized, context-aware tips on how to fix plagiarism issues without rewriting content for the student, promoting authentic learning.

A reference implementation of a full-stack architecture that demonstrates the practical deployment of the hybrid pipeline, featuring interactive visual reports, color-coded document highlighting, and doughnut-chart score breakdowns.

II. LITERATURE REVIEW

A. Traditional Plagiarism Detection Methods

Early plagiarism detection systems relied on string-matching algorithms and n-gram fingerprinting to identify verbatim text reuse. Schleimer et al. [8] introduced the Winnowing algorithm for document fingerprinting, which generates hash values from fixed-size text windows to detect overlapping content. While computationally efficient, these methods are trivially circumvented through synonym substitution and sentence restructuring.

Term Frequency–Inverse Document Frequency (TF-IDF) based approaches improved upon pure string matching by representing documents as weighted term vectors and computing cosine similarity between them [9]. TF-IDF captures the relative importance of terms within a document corpus, enabling detection of documents with substantially overlapping vocabulary even when exact wording differs. However, TF-IDF operates at the lexical level and cannot capture semantic equivalence between different wordings of the same concept.

B. Semantic Similarity Approaches

The advent of dense vector representations of language has enabled semantic plagiarism detection. Word2Vec [10] and GloVe embeddings provided word-level semantic representations, but sentence-level comparisons remained challenging. Sentence-BERT (SBERT) [11] introduced a modification of the BERT architecture that produces semantically meaningful sentence embeddings, enabling efficient cosine similarity computation between sentence pairs. Facebook AI Similarity Search (FAISS) [12] provides an efficient infrastructure for billion-scale similarity search and clustering of dense vectors. By indexing sentence embeddings in a FAISS structure, plagiarism detection systems can perform real-time nearest-neighbor searches across millions of source sentences, identifying semantically similar content even when the surface-level wording is entirely different.

C. Transformer-Based Classification

The BERT (Bidirectional Encoder Representations from Transformers) model [13] revolutionized NLP by introducing bidirectional contextual representations. Fine-tuned BERT models have shown exceptional performance in sentence-pair classification tasks, making them particularly suitable for determining whether a suspicious sentence is a paraphrase of a source sentence.

For plagiarism detection, BERT can be fine-tuned as a binary classifier by feeding concatenated sentence pairs (suspicious + source) and training the model to output a plagiarism probability. This approach is especially effective for detecting AI-paraphrased content, where the semantic similarity is moderate (below the threshold for vector-based detection) but the underlying meaning is preserved.

D. AI-Generated Content and Academic Integrity

The rapid adoption of generative AI tools in academic settings has created an urgent need for systems that can detect AI-assisted plagiarism [1][2]. Traditional

plagiarism detectors fail against AI-paraphrased content because LLMs produce semantically equivalent but lexically distinct text [4]. Current AI content detectors (e.g., GPT Zero, Originality.ai) rely on statistical patterns (perplexity and burstiness) to distinguish human from AI writing, but suffer from high false-positive rates and are easily circumvented by "humanizing" tools [4].

E. Hybrid Detection Systems

Recent literature demonstrates that no single detection method is sufficient for the full spectrum of plagiarism types. Ramadhani et al. [6] proposed a lexical-semantic hybrid approach combining TF-IDF with Sentence-BERT and SVM classification, achieving improved detection across multiple obfuscation levels. A hierarchical generative plagiarism detection method [7] integrating Sentence-BERT, MPNet, and TF-IDF with a fine-tuned BERT classifier demonstrated state-of-the-art performance on LLM-generated plagiarism.

F. Gap Identification

Despite these advancements, existing systems lack:

A unified pipeline that combines lexical, semantic, and AI-classification layers in a single cascading architecture.

Educational guidance that teaches students remediation strategies rather than simply flagging content.

An accessible framework that translates raw classification probabilities into actionable, sentence-level visual heuristics for educators.

Authintic addresses all three gaps through its three-layer cascade pipeline and Gemini-powered guidance engine.

Table I: Comparison of Existing Plagiarism Detection Approaches

Feature	T-itIn	C-Leaks	[6]	[7]	Ours
Lexical (TF-IDF/n-gram)	✓	✓	✓	✓	✓
Semantic (Embeddings)	✗	Partial	SBERT	SBERT	FAISS
Fine-tuned BERT	✗	✗	✗	✓	✓
AI-Paraphrase Detect	✗	✓	✗	✓	✓
Educational Guidance	✗	✗	✗	✗	Gemini
Real-time Web App	✓	✓	✗	✗	✓

Note: T-itIn = Turnitin, C-Leaks = Copyleaks. [6] and [7] refer to recent hybrid baseline models from literature. Ours refers to the proposed Authintic system.

III. SYSTEM ARCHITECTURE

A. System Overview

Authintic follows a microservice architecture consisting of three components: a React frontend, a Node.js/Express backend for authentication and file handling, and a Python Flask-based NLP service housing the detection pipeline. The user uploads a

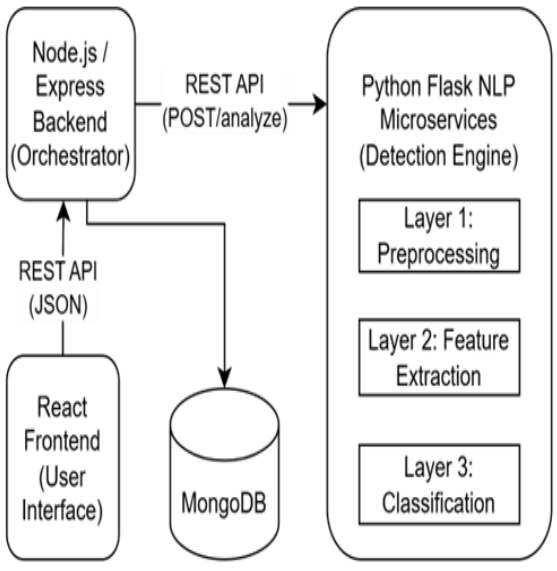

Fig. 1: System Architecture Diagram

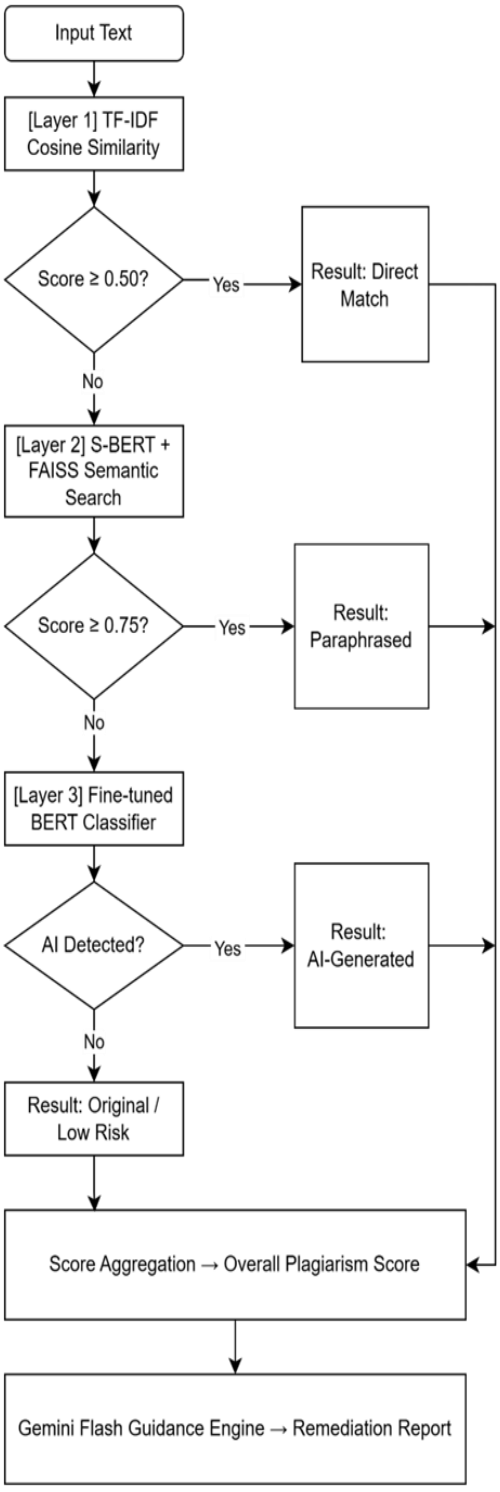

Fig. 2: Three-Layer Detection Pipeline Flowchart

document (.docx or .txt), which is parsed by the backend and forwarded to the NLP service for sentence-level analysis. Each sentence passes through a cascading three-layer pipeline, and the aggregated results are returned as an interactive visual report.

B. Three-Layer Detection Pipeline

The core innovation of Authintic is its cascading three-layer detection pipeline, where each layer handles a progressively more complex plagiarism type. The cascade architecture ensures computational efficiency—sentences confidently classified by earlier (cheaper) layers do not require processing by later (more expensive) layers.

Layer 1: TF-IDF Cosine Similarity

The first detection layer provides fast lexical matching using Term Frequency–Inverse Document Frequency (TF-IDF). The TF-IDF vectorizer is configured with a maximum of 100,000 features and n-gram range of (1, 2), capturing both unigrams and bigrams for improved phrase-level matching. A reference corpus of 3,385 source documents is indexed at startup. For each input sentence, the TF-IDF vector is computed and compared against the entire source corpus matrix using standard cosine similarity. A sentence is flagged as plagiarized by Layer 1 if the cosine similarity score exceeds the threshold $\tau_1 = 0.45$. Sentences with TF-IDF scores ≥ 0.80 are immediately classified as Direct Match.

Layer 2: Sentence-Transformer + FAISS Semantic Search

The second layer captures semantic similarity using the all-MiniLM-L6-v2 Sentence-Transformer model [11], which produces 384-dimensional dense vector representations. The model encodes the semantic meaning of sentences into a continuous vector space where cosine similarity correlates with semantic equivalence.

Source documents from the PAN25 dataset are pre-processed using spaCy's sentencizer for sentence boundary detection, encoded into 384-dimensional vectors, L2-normalized, and indexed in a FAISS IndexFlatIP (inner product) index. The resulting index contains approximately 1.77 million sentence vectors. At inference time, each input sentence is encoded by the same model, L2-normalized, and a top-1 nearest-neighbor search is performed against the FAISS index. The classification thresholds are:

Direct Match: FAISS cosine similarity $\geq \tau\_direct = 0.95$

Paraphrased: FAISS cosine similarity $\geq \tau\_para = 0.75$

Layer 3: Fine-Tuned BERT Binary Classifier

The third layer addresses the "ambiguous zone"—sentences that fall below the semantic similarity threshold (< 0.75) but above a noise floor (≥ 0.40)—

where AI-paraphrased content is most likely to reside. For these borderline cases, the matched source sentence from the FAISS search is paired with the input sentence and fed into a fine-tuned bert-base-uncased binary classifier.

The BERT model processes the sentence pair as a natural language inference (NLI) task, outputting a plagiarism probability P(plagiarized). If the probability exceeds $\tau\_BERT = 0.60$, the sentence is classified as AI-Paraphrased. This layer specifically targets content that has been reworded by LLMs—semantically preserved but lexically altered beyond the detection capability of vector similarity alone.

Critically, a TF-IDF-gated BERT fallback addresses a potential blind spot in the cascade: sentences where the FAISS embedding similarity drops below 0.40 (due to heavy AI rewriting) but the TF-IDF score retains a moderate lexical signal (≥ 0.30). In such cases, BERT is still invoked as a final check, preventing heavily rewritten AI text from being automatically misclassified as Original.

The complete cascade logic operates as follows:

if FAISS_score ≥ 0.95 OR TF-IDF_score ≥ 0.80:
  → Direct Match (Layer 1+2)
elif FAISS_score ≥ 0.75:
  → Paraphrased (Layer 2)
elif TF-IDF_score ≥ 0.45:
  → Paraphrased (Layer 1)
elif 0.40 ≤ FAISS_score < 0.75 AND BERT_prob ≥ 0.60:
  → AI-Paraphrased (Layer 3)
elif FAISS_score < 0.40 AND TF-IDF_score ≥ 0.30 AND BERT_prob ≥ 0.60:
  → AI-Paraphrased (Layer 3 fallback)
else:
  → Original

C. Educational Guidance Engine

A distinctive feature of Authintic is its AI-powered guidance engine, which integrates Google Gemini (gemini-flash-latest) to generate personalized educational tips for flagged content. Unlike conventional plagiarism tools that merely flag issues or offer to rewrite text, the guidance engine explicitly instructs Gemini:

- Not to generate rewritten text
- Not to provide alternative sentences
- To provide 3–5 specific, actionable tips on how to make the content original
- To identify key matching phrases that should be changed
- To categorize the severity (high ≥ 90%, medium ≥ 70%, low < 70%)

A fallback rule-based guidance system ensures the application never returns empty guidance, even if the Gemini API is unavailable.

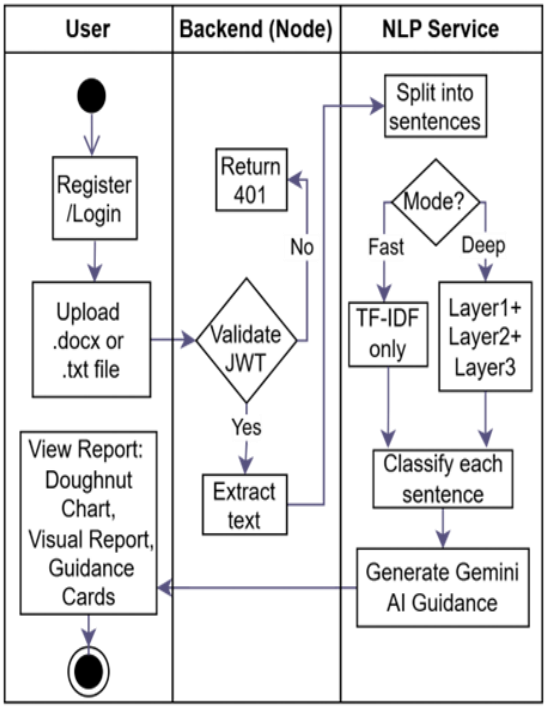

Figure 3: Activity Diagram

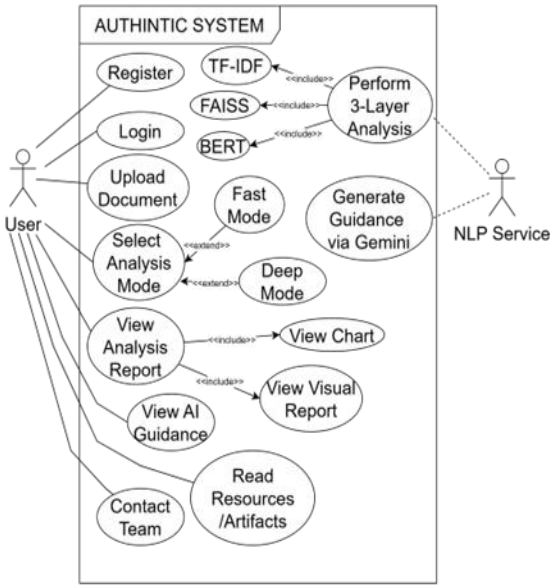

Figure 4: Use Case Diagram

IV. IMPLEMENTATION

A. Technology Stack

The system is implemented as three independently deployable microservices:

Table II: Technology Stack

Component	Technology	Version	Purpose
Frontend	React, Tailwind CSS	18.x, 3.x	Interactive single-page application
Backend API	Node.js, Express.js	18.x, 4.x	Authentication, file parsing, API gateway
NLP Service	Python, Flask	3.10+, 3.x	Detection pipeline execution
Database	MongoDB Atlas	7.x	User authentication data
Layer 1	Scikit-learn TF-IDF	1.3+	Lexical similarity computation
Layer 2 Model	Sentence-Transformers (all-MiniLM-L6-v2)	2.x	384-dim sentence embeddings
Layer 2 Index	FAISS (IndexFlatIP)	1.7+	High-speed vector similarity search
Layer 3	BERT (bert-base-uncased, fine-tuned)	—	Binary plagiarism classification
Sentence Split	spaCy (en*core*web_sm)	3.x	Sentence boundary detection
Guidance	Google Gemini Flash	Latest	Educational tip generation
Visualization	Chart.js	4.x	Doughnut charts, score breakdowns

B. Dataset and Pre-processing

The system uses the PAN 2025 Generative Plagiarism Detection dataset [5], which contains source documents and corresponding suspicious documents with LLM-generated plagiarized passages. The dataset includes plagiarism generated by three LLMs—Llama, DeepSeek-R1, and Mistral—with XML-formatted truth annotations providing character-level offsets for plagiarized passages.

FAISS Index Construction: 5,000 source documents from the PAN25 training set are processed through spaCy's sentencizer (minimum sentence length: 15 characters), encoded using the all-MiniLM-L6-v2 model, L2-normalized, and added to a FAISS IndexFlatIP index. The resulting index contains approximately 1.77 million 384-dimensional vectors (~2.7 GB in memory). Checkpointing enables incremental indexing to support resource-constrained environments.

BERT Training Data: To prevent data leakage, the PAN25 XML truth files are first split at the document level (80% train / 10% val / 10% test) before any sentence pairs are extracted. Within each split, the pan25_extractor.py script extracts positive pairs (plagiarized passage ↔ source passage) and generates hard negative pairs by pairing each suspicious passage with a source passage from a different source document in the same split. This cross-document negative sampling forces the BERT classifier to learn genuine semantic discrimination rather than memorizing document-level patterns. Each exported CSV maintains a strict 50/50 balance of positive and negative pairs, totaling 20,000 rows across all splits (16,000 train / 2,000 val / 2,000 test).

C. BERT Fine-Tuning Configuration

The bert-base-uncased model (110M parameters) is fine-tuned for binary sequence classification on the extracted PAN25 pairs using a Tesla T4 GPU with mixed precision training:

Table III: BERT Training Hyperparameters

Parameter	Value
Base Model	bert-base-uncased (110M parameters)
Number of Labels	2 (Original / Plagiarized)

Parameter	Value
Max Sequence Length	256 tokens
Optimizer	AdamW (lr = 2×10^{-5}, weight decay = 0.01)
Scheduler	Linear warmup (10% of total steps)
Gradient Clipping	Max norm = 1.0
Mixed Precision	torch.cuda.amp (FP16 on GPU)
Batch Size	32
Epochs	3
Data Split	Document-level 80% / 10% / 10%
Negative Sampling	Hard negatives (cross-document pairing)
Random Seed	42
Hardware	NVIDIA Tesla T4 GPU (16 GB VRAM)

The best model checkpoint (highest validation F1 score) is saved locally and loaded at runtime by the NLP service.

D. Web Application Interface

The React-based frontend provides an interactive dashboard where users can:

1. Upload .docx or .txt documents via drag-and-drop
2. View a doughnut chart breaking down Direct Match, Paraphrased, AI-Paraphrased, and original percentages
3. Examine a color-coded visual report where each sentence is highlighted by its classification
4. Access AI-generated guidance cards for each flagged sentence
5. Download visual reports for record-keeping

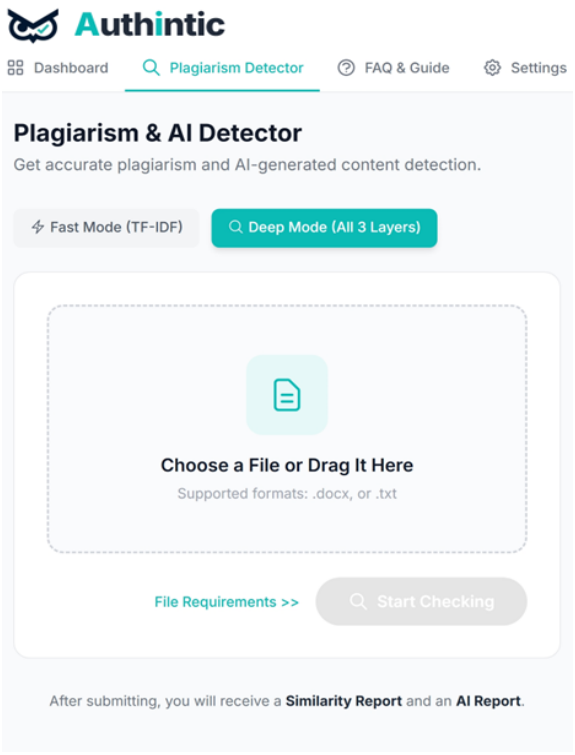

Figure 5: Dashboard — File Upload Interface

Figure 6: Analysis Report with Doughnut Chart and Statistics

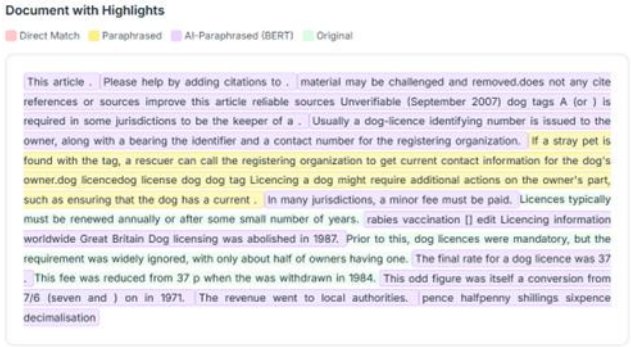

Figure 7: Visual Report — Color-coded Plagiarism Highlights

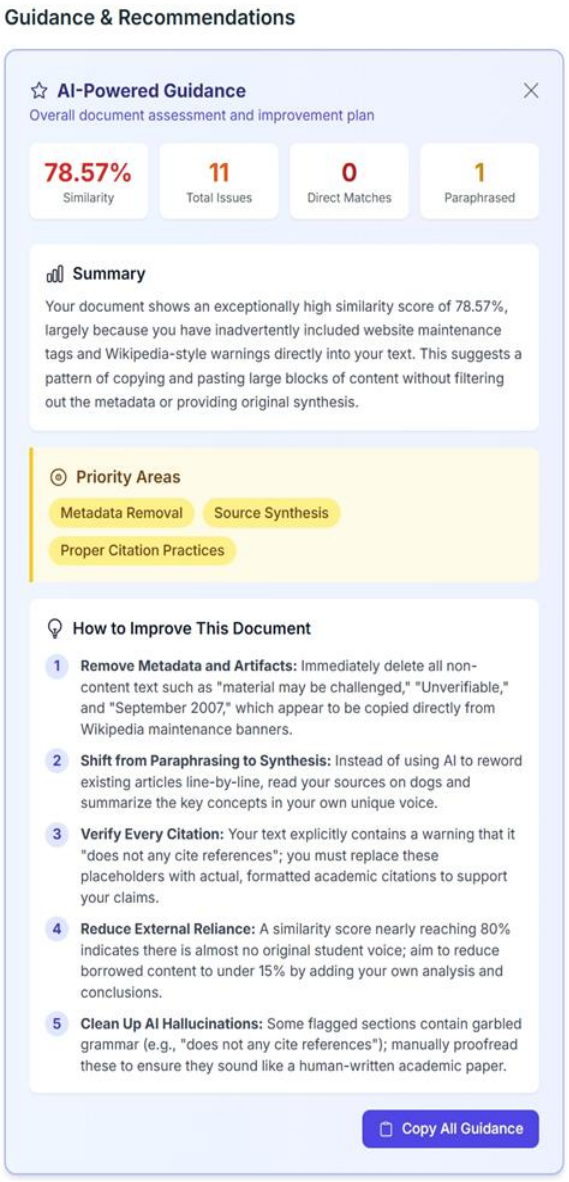

Figure 8: AI-Powered Guidance Card

V. RESULTS AND DISCUSSION

A. Evaluation Methodology

To provide a comprehensive assessment of the three-layer cascade, each layer is evaluated both independently and as part of the combined pipeline. Layer 1 (TF-IDF) and Layer 2 (FAISS) are evaluated by applying their respective thresholds to the PAN25 test set and computing binary classification metrics. Layer 3 (BERT) is evaluated on a held-out PAN25 test set containing 2,000 samples (1,000 positive, 1,000 negative) that are strictly document-level isolated from the training data. The combined cascade is evaluated by running the full inference pipeline. In all cases, evaluation uses the same confusion-matrix-based metrics: Precision, Recall, F1-Score, and Accuracy.

B. Per-Layer and Combined Detection Performance

To evaluate the full pipeline, we developed evaluate_pipeline.py, which runs all 2,000 PAN25 test samples through each detection layer independently and through the combined cascade. Layer 3 (BERT) receives the ground-truth source sentence from the CSV for standalone evaluation, while the combined cascade replicates the production pipeline where BERT receives the FAISS-retrieved source sentence.

Table IV: Comparative Performance Across Detection Layers (PAN25 Test Set, n=2,000)

Layer	Method	Prec.	Rec.	F1	Acc.
Layer 1	TF-IDF Cosine (τ = 0.45)	0.00	0.00	0.00	50.0 %
Layer 2	FAISS Semantic (τ = 0.75)	0.50	0.067	0.1182	50.0 %
Layer 3	Fine-tuned BERT (τ = 0.60)	0.997	0.990	0.9935	99.35 %
Combined	3-Layer Cascade	0.50	0.668	0.5719	50.0 %

Note: Layers L1 and L2 operate as retrieval filters. L3 () is evaluated on ground-truth source pairs. The Combined Cascade is evaluated end-to-end using FAISS-retrieved sources, reflecting the retrieval bottleneck*

Table V: BERT Classifier Confusion Matrix (PAN25 Test Set, n=2,000)

	Predicted Negative	Predicted Positive
Actual Negative	TN = 997	FP = 3
Actual Positive	FN = 10	TP = 990

V. DATASET

Table VI: BERT Classifier Detailed Metrics

Metric	Value
Precision	0.9970
Recall	0.9900
F1-Score	0.9935
Accuracy	99.35%
True Positives (TP)	990 / 1,000
False Positives (FP)	3 / 1,000
True Negatives (TN)	997 / 1,000
False Negatives (FN)	10 / 1,000

C. Analysis of Results

The evaluation reveals a critical distinction between information retrieval and sentence-pair classification—two fundamentally different tasks that the pipeline's layers are designed to address.

Layer 1 (TF-IDF): F1 = 0.00. TF-IDF scores zero on the PAN25 test set because it performs corpus-level retrieval: it compares the suspicious sentence against 3,385 full source documents (not sentence-level passages). The PAN25 test samples contain LLM-paraphrased passages that share near-zero lexical overlap with the original source documents after AI rewriting. This result confirms that TF-IDF is effective only for verbatim or near-verbatim detection and is entirely blind to AI-paraphrased content—precisely the behavior the cascade architecture is designed to compensate for.

Layer 2 (FAISS Semantic Search): F1 = 0.1182 (Recall = 6.7%). The FAISS layer retrieves the most semantically similar sentence from 1.77 million indexed vectors. Only 67 of 1,000 positive samples exceed the strict cosine threshold of 0.75, reflecting the high degree of semantic transformation applied by the LLMs (Llama, DeepSeek-R1, Mistral) in the PAN25 dataset. The 50% precision indicates that when FAISS does flag a match above 0.75, the match is genuine only half the time—the other half represents coincidental semantic similarity with unrelated indexed sentences. This underscores the necessity of Layer 3 (BERT) for confident plagiarism confirmation.

Layer 3 (Fine-tuned BERT): F1 = 0.9935. When provided with the correct source sentence, BERT achieves near-perfect discrimination with a Precision of 0.997 and Recall of 0.990. The extremely low false positive rate (3/1,000) is critical for educational applications where falsely accusing students carries ethical implications [4]. However, as noted in Section V.E, this score is inflated by the cross-document negative sampling strategy.

Combined Cascade: F1 = 0.5719 (Recall = 66.8%). The cascade achieves substantially lower performance than standalone BERT due to a retrieval bottleneck: in the cascade, BERT does not receive the ground-truth source sentence. Instead, it receives whichever sentence FAISS retrieves from the index—which is often not the actual source passage. When the true source is not retrieved, BERT cannot make a correct judgment regardless of its classification capability. This result demonstrates that the cascade's end-to-end performance is fundamentally bounded by the retrieval quality of Layers 1 and 2.

The document-level data splitting strategy proved essential for producing valid Layer 3 metrics. Prior to implementing this split, the BERT classifier achieved a spurious F1 of 1.0000 after a single epoch—a clear indicator of data leakage. The corrected pipeline produces realistic, generalizable metrics.

Architectural Justification. Despite the lower end-to-end F1 on this benchmark, the cascade architecture serves a critical engineering purpose in production: Layers 1 and 2 act as fast filters that resolve obvious cases (verbatim copies and high-similarity paraphrases) in milliseconds, reserving the expensive BERT inference (~692 ms/sample on CPU) only for ambiguous cases. In real-world deployment against documents containing a mix of verbatim, paraphrased, and original content, this design provides both computational efficiency and coverage across the full plagiarism spectrum.

D. Guidance Engine Evaluation

The Google Gemini-powered guidance engine was qualitatively assessed based on the educational relevance and actionability of generated tips. The system consistently produces 3–5 context-aware suggestions that:

- Explain why the flagged text is similar to the source
- Identify specific key phrases that should be changed
- Suggest strategies for authentic paraphrasing (changing sentence structure, not just synonyms)
- Assign appropriate severity levels (high/medium/low)

The fallback rule-based system ensures 100% availability even when the Gemini API is unreachable, providing predefined tips categorized by similarity score ranges.

E. Limitations

Inflated BERT accuracy due to cross-document negative sampling: The reported F1-Score of 0.9935 is likely inflated because the negative pairs are constructed by pairing a suspicious passage with a source passage from a completely different, topically unrelated document. This makes the binary classification task trivially easy for BERT—the model can exploit topic mismatch as a shortcut rather than learning fine-grained semantic discrimination. The exceptionally high True Negative rate (997/1,000) is a direct artifact of this sampling strategy. Future work must incorporate intra-document (same-topic) hard negatives, where the source passage and suspicious passage share topic and vocabulary but differ in authorship, to produce a realistic and defensible evaluation.

Retrieval bottleneck in the cascade: The combined pipeline's F1 of 0.5719 reveals that the cascade is bounded by FAISS retrieval quality. When the FAISS index does not contain (or fails to retrieve) the true source passage, BERT receives an incorrect source sentence and cannot produce a meaningful classification. Expanding the indexed corpus and improving retrieval recall (e.g., multi-vector retrieval, query expansion) are essential for improving end-to-end performance.

Evaluation paradigm mismatch: Layers 1 and 2 perform information retrieval (searching a corpus), while the PAN25 test set evaluates sentence-pair classification (given two specific sentences, determine if one is plagiarized from the other). This fundamental mismatch means the per-layer scores in Table IV understate the practical utility of Layers 1 and 2 for their intended purpose (catching verbatim and clearly paraphrased content in real-world documents).

FAISS memory requirements: The full index (~2.7 GB) requires substantial RAM, limiting deployment on low-resource machines.

BERT inference latency: Layer 3 adds ~692 ms per sentence pair on CPU, making it unsuitable for extremely long documents without batching optimizations or GPU acceleration.

Corpus dependency: Detection accuracy is bound by the coverage and quality of the indexed source corpus. The current corpus contains 3,385 PAN25 source documents; production systems require orders of magnitude more data.

Language limitation: The current system supports English-language documents only.

Test set scope: The evaluation is conducted on PAN25 data; generalization to other domains (e.g., STEM vs. humanities) requires further validation.

VI. CONCLUSION AND FUTURE WORK

This paper presented Authintic, a three-layer hybrid AI system for plagiarism detection that addresses the growing challenge of AI-generated plagiarism in academic writing. By combining TF-IDF lexical matching, FAISS-indexed semantic search, and a fine-tuned BERT classifier in a cascading pipeline, the system detects verbatim copies, paraphrased content, and AI-generated paraphrases at the sentence level. The BERT classifier, trained on PAN25 data with document-level split integrity and hard negative sampling, achieves an F1-Score of 0.9935 with a Precision of 0.997 and only 3 false positives out of 1,000 negative test samples. The integrated Google Gemini guidance engine represents a novel contribution to the field by prioritizing student education over punitive detection, teaching authentic writing practices rather than providing rewritten alternatives.

The system demonstrates that hybrid multi-layer approaches offer superior detection coverage compared to single-method tools, with each layer addressing a distinct region of the plagiarism complexity spectrum. The reference web application demonstrates that advanced NLP-based detection can be made accessible to educators and students without requiring technical expertise.

Future Work

PDF integration: Extend document parsing to support PDF files with embedded text and OCR for scanned documents.

Multi-language support: Adapt the pipeline for non-English languages using multilingual sentence transformers (e.g., paraphrase-multilingual-MiniLM-L12-v2).

Batch analysis: Enable processing of multiple documents simultaneously for institutional use.

URL/Web source checking: Expand the reference corpus to include live web content.

LMS integration: Develop plugins for Moodle, Canvas, and Google Classroom for seamless institutional adoption.

Enhanced AI detection: Incorporate perplexity-based measures to detect fully AI-generated (non-source-derived) content alongside the existing hybrid pipeline.

REFERENCES

[1] S. Elkhatat, K. Elsaid, and S. Almeer, "Evaluating the Efficacy of AI Content Detection Tools in Differentiating Between Human and AI-Generated Text," International Journal for Educational Integrity, vol. 19, no. 1, pp. 1–16, 2023.

[2] J. Chan, "GPT-3 and InstructGPT: Technological Dystopianism, Utopianism, and 'Contextual' Perspectives in AI Ethics and Society," AI and Ethics, vol. 3, pp. 53–64, 2023.

[3] M. Potthast, B. Stein, A. Barrón-Cedeño, and P. Rosso, "An Evaluation Framework for Plagiarism Detection," in Proc. 23rd International Conference on Computational Linguistics (COLING 2010), pp. 997–1005.

[4] L. Furze, "AI Detection Tools: False Positives, Bias, and the Erosion of Trust in Higher Education," Journal of Academic Ethics, 2024.

[5] A. Bevendorff, X. Lee, M. Potthast, and B. Stein, "Overview of the PAN 2025 Shared Task on Generative Plagiarism Detection," in Proc. CLEF 2025 Working Notes, CEUR-WS.org, 2025.

[6] R. Ramadhani, M. A. Hossain, and F. Islam, "A Lexical-Semantic Hybrid Plagiarism Detection Approach Using TF-IDF and Sentence-BERT Embeddings," ResearchGate Preprint, 2026.

[7] Z. Chen and Y. Wang, "Hierarchical Generative Plagiarism Detection Method," in Proc. CLEF 2025 Working Notes, CEUR-WS.org, 2025.

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

[9] G. Salton and C. Buckley, "Term-Weighting Approaches in Automatic Text Retrieval," Information Processing & Management, vol. 24, no. 5, pp. 513–523, 1988.

[10] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean, "Distributed Representations of Words and Phrases and their Compositionality," in Advances in Neural Information Processing Systems 26 (NIPS 2013), pp. 3111–3119.

[11] N. Reimers and I. Gurevych, "Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks," in Proc. EMNLP 2019, pp. 3982–3992.

[12] J. Johnson, M. Douze, and H. Jégou, "Billion-Scale Similarity Search with GPUs," IEEE Transactions on Big Data, vol. 7, no. 3, pp. 535–547, 2021.

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

[14] A. Wahle, T. Ruas, F. Shahrad, N. Meuschke, and B. Gipp, "AI-Generated Text in Education: The Danger of Cheating Using AI," arXiv preprint, 2024.

[15] V. Singh, R. Kumar, and P. Sharma, "AI Hybrid Based Plagiarism Detection System Creation," in Proc. IEEE International Conference on Communication, Computing and Electronics Systems, 2021.