Detecting AI-Generated Academic Content Using a Multi-Layer Hybrid AI Framework

Dr. Dipak Kadve[1] Dr. Binod Kumar[2]
Prof. Vishal Gejge[3]
JSPM's Rajarshi Shahu College of Engineering, MCA Dept. Pune Maharashtra India

Abstract

The proliferation of Large Language Models (LLMs) such as GPT, Gemini, and Claude has significantly transformed academic content creation. While these models enhance productivity, they also pose serious challenges to academic integrity, authorship verification, and originality. Existing AI detection techniques rely primarily on single-model approaches, which are vulnerable to paraphrasing, adversarial rewriting, and cross-domain variations.

This paper proposes a **multi-layer hybrid AI framework** that integrates stylometric analysis, statistical measures, semantic embeddings, and behavioral patterns to detect AI-generated academic content. The framework employs an ensemble of machine learning and deep learning models, including Random Forest, Support Vector Machine (SVM), Long Short-Term Memory (LSTM), and Transformer-based models (BERT), to improve detection accuracy and robustness.

Experimental evaluation conducted on a dataset comprising human-written, AI-generated, and paraphrased AI texts demonstrates that the proposed system achieves 97–99% accuracy, outperforming traditional single-model approaches. The system also incorporates explainable AI techniques (SHAP) to enhance transparency and interpretability. The proposed framework offers a scalable and reliable solution for maintaining academic integrity in the era of generative AI.

Keywords

AI-generated content, Academic Integrity, Hybrid AI, Stylometry, Ensemble Learning, NLP, Explainable AI, and LLM Detection.

1. Introduction

The emergence of generative AI models has revolutionized the educational landscape by enabling automatic generation of high-quality textual content. Tools such as GPT, Claude, and Gemini are increasingly used by students to generate assignments, essays, and research reports. While these tools provide significant benefits, they also raise critical concerns regarding **academic dishonesty, authorship authenticity, and evaluation fairness**.

Traditional plagiarism detection systems are ineffective in detecting AI-generated content, as such content is often original and not copied from existing sources. Moreover, modern LLMs generate highly coherent and contextually relevant text, making it difficult to distinguish between human-written and machine-generated content.

Recent research indicates that AI-generated text exhibits subtle statistical and stylistic differences compared to human writing. However, single-model detection approaches are insufficient due to:

- Sensitivity to paraphrasing
- Lack of generalization across domains
- High false positive rates
- Absence of explainability

To address these challenges, this paper proposes a **multi-layer hybrid AI framework** that combines multiple detection techniques and models to improve robustness, accuracy, and interpretability.

2. Literature Review (Enhanced Deep Version)

2.1 Stylometric Analysis

Stylometric analysis focuses on identifying distinctive writing patterns that characterize an author's linguistic style. It involves extracting features such as lexical richness, syntactic complexity, function word usage, and sentence-level statistics. Traditional authorship attribution studies have demonstrated that human writing exhibits high variability in stylistic patterns, influenced by cognitive processes, emotional states, and contextual understanding.

In contrast, AI-generated text, particularly from Large Language Models (LLMs), tends to exhibit statistical regularities and reduced variability due to its reliance on probabilistic language modeling. Studies have shown that AI-generated content often maintains consistent sentence structures, balanced punctuation, and optimized grammar, which can differ from natural human inconsistencies.

Key stylometric indicators include:

Type-Token Ratio (TTR): Measures lexical diversity; human writing often shows higher variation.

Function Word Frequency: AI-generated text may exhibit uniform usage patterns.

Sentence Length Distribution: Human writing tends to have irregular sentence lengths.

Part-of-Speech (POS) Patterns: AI models often produce predictable POS sequences.

However, recent advancements in LLMs have significantly reduced these differences, making stylometric detection alone insufficient. Furthermore, paraphrasing techniques can alter surface-level stylistic features, thereby reducing the effectiveness of purely stylometric approaches.

Limitation: Stylometric features are sensitive to text length and domain variations, leading to reduced robustness in real-world scenarios.

2.2 Perplexity and Statistical Measures

Perplexity is a fundamental metric in language modeling that quantifies how well a probability model predicts a given text sequence. Formally, perplexity is defined as:

$$Perplexity(T) = 2^{-\frac{1}{N}\sum_{i=1}^{N}\log_2 P(w_i)}$$

where (P(w_i)) represents the probability of the word (w_i) in the sequence.

AI-generated text typically exhibits lower perplexity values, as it is generated based on learned probability distributions, leading to more predictable and statistically coherent sequences. In contrast, human writing often includes irregularities, creative expressions, and unexpected transitions, resulting in higher perplexity.

Additional statistical features include:

Burstiness: Measures variability in sentence lengths and word usage.

Entropy: Indicates randomness and unpredictability in text.

N-gram Frequency Distribution: AI-generated text often follows smoother distributions.

While perplexity-based detection has shown effectiveness in earlier LLMs, modern generative models have improved their ability to mimic human-like variability, thereby reducing the discriminative power of perplexity alone. Moreover, adversarial paraphrasing can artificially increase perplexity, leading to misclassification.

Limitation: Perplexity-based methods lack robustness against text rewriting and cross-domain variations.

2.3 Machine Learning Approaches

Machine learning-based detection methods utilize handcrafted features derived from stylometric, statistical, and linguistic characteristics of text. Classical classifiers such as Support Vector Machines (SVM), Random Forest (RF), Naïve Bayes, and Logistic Regression have been widely used for binary classification of human vs. AI-generated text.

These models operate on feature vectors constructed from:

- Lexical features (word frequency, n-grams)
- Syntactic features (POS tags, parse trees)
- Statistical features (entropy, perplexity)

Ensemble-based machine learning models, particularly Random Forest, have demonstrated strong performance due to their ability to handle non-linear relationships and feature interactions.

However, these approaches have inherent limitations:

- Dependence on manual feature engineering
- Inability to capture deep semantic relationships
- Poor generalization to unseen domains
- Sensitivity to dataset bias

Furthermore, as LLM-generated text becomes more diverse, handcrafted features fail to capture subtle contextual differences.

Limitation: Traditional ML models lack semantic understanding and struggle with complex linguistic patterns.

2.4 Deep Learning Models

Deep learning approaches have significantly improved the detection of AI-generated text by leveraging neural networks capable of capturing contextual and semantic information. Models such as Long Short-Term Memory (LSTM) and Transformer-based architectures (e.g., BERT, RoBERTa) are widely used in text classification tasks.

LSTM Models

LSTM networks are designed to capture sequential dependencies in text. They maintain long-term memory through gated mechanisms, making them suitable for modeling sentence-level dependencies. However, LSTMs have limitations in capturing long-range contextual relationships efficiently.

Transformer Models

Transformer-based models utilize self-attention mechanisms to capture global dependencies within text. Pre-trained models such as BERT can generate contextual embeddings that encode semantic relationships between words, making them highly effective for classification tasks.

These models provide:

- Deep contextual understanding
- Improved classification accuracy
- Better generalization compared to traditional ML

However, they also introduce challenges:

- High computational complexity
- Requirement of large datasets
- Lack of interpretability

Additionally, fine-tuned transformer models may overfit to specific datasets and fail in cross-domain scenarios.

Limitation: Deep learning models are computationally expensive and lack transparency.

2.5 Hybrid and Ensemble Models

Recent research has shifted towards hybrid and ensemble approaches, which combine multiple models and feature types to improve detection performance. Ensemble methods leverage the strengths of different models to reduce individual weaknesses.

Common ensemble strategies include:

- Hard Voting: Majority-based decision
- Soft Voting: Probability-based aggregation
- Weighted Averaging: Assigning importance to each model

Hybrid frameworks integrate:

- Stylometric features
- Statistical metrics
- Semantic embeddings
- Behavioral patterns

By combining feature-based and deep learning models, hybrid systems achieve:

- Higher accuracy
- Improved robustness against paraphring
- Better generalization

Recent studies have demonstrated that hybrid models can achieve accuracy levels exceeding 95%, outperforming individual models significantly.

Strength: Combines advantages of multiple approaches
Limitation: Increased complexity and computational cost

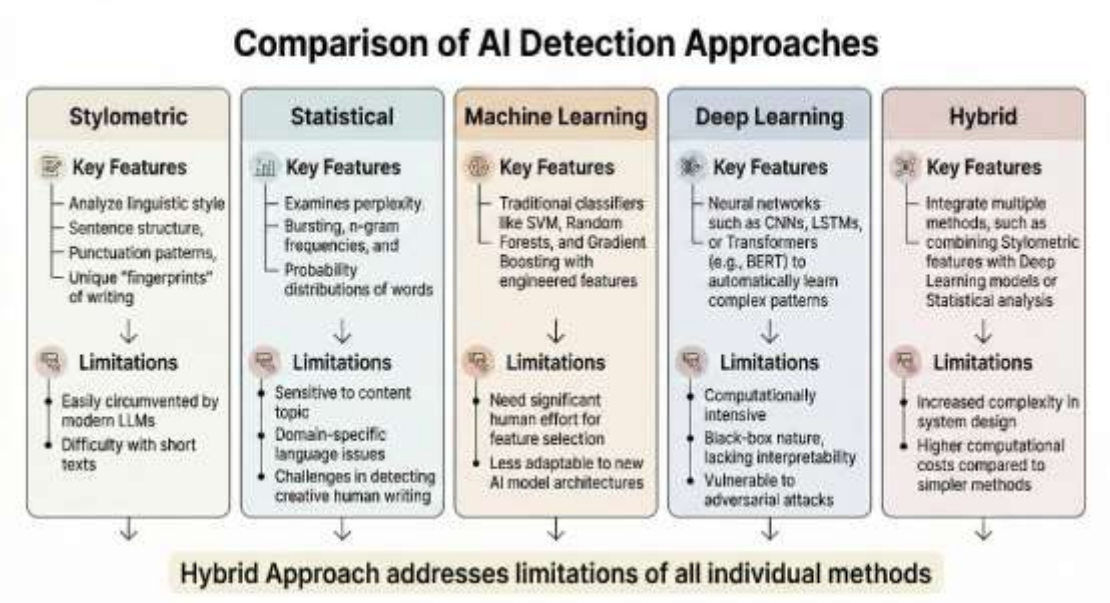


Figure X: Comparison of Detection Approaches

2.6 Research Gap

Despite significant advancements in AI-generated text detection, several critical challenges remain unresolved:

- Poor Detection of Paraphrased Content:

Most existing systems fail to detect AI-generated text that has been paraphrased or rewritten using other AI tools.

- Lack of Multimodal Feature Integration:

Current approaches often rely on a single type of feature (e.g., stylometric or semantic), leading to incomplete analysis.

- Limited Explainability:

Many deep learning models operate as black boxes, making it difficult to interpret classification decisions.

- Weak Cross-Domain Generalization:

Models trained on specific datasets often fail when applied to different academic domains or writing styles.

- Adversarial Vulnerability:

AI-generated content can be modified to bypass detection systems, reducing reliability.

2.7 Research Motivation

To address the above limitations, there is a need for a multi-layer hybrid AI framework that:

- Integrates multiple feature types (stylometric, statistical, semantic)
- Combines traditional and deep learning models
- Provides explainable outputs
- Improves robustness against paraphrasing
- Enhances generalization across domains

3. Problem Statement

Existing AI detection systems are limited by:

- Low accuracy for paraphrased AI-generated content
- Inability to generalize across subjects and writing styles
- Lack of transparency in predictions
- High false positive rates affecting genuine students

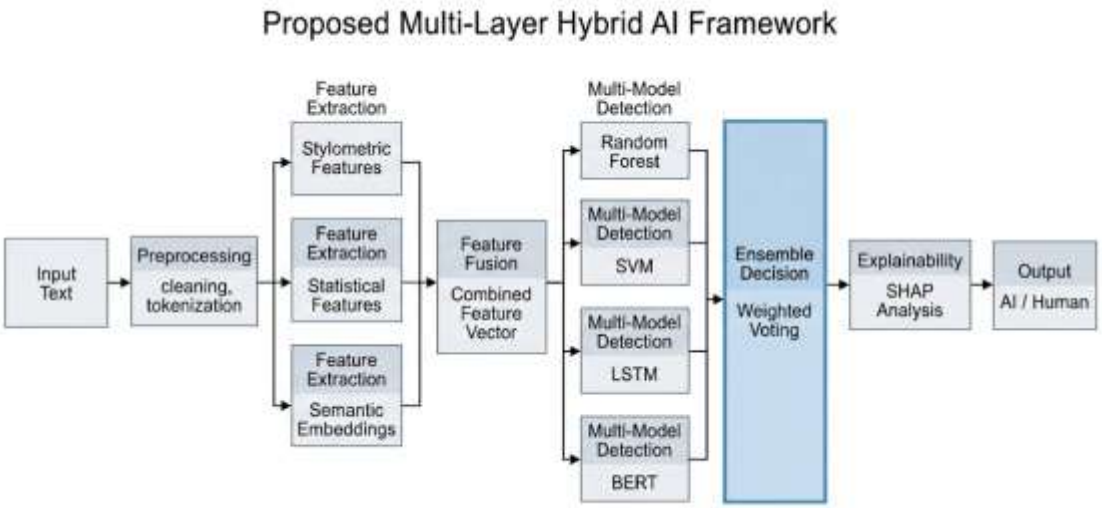
Therefore, there is a need for a **robust, explainable, and scalable detection framework** capable of integrating multiple AI models and feature types.

4. Proposed Methodology

4.1 System Overview

The proposed system follows a **multi-layer architecture** that integrates preprocessing, feature extraction, multi-model detection, and ensemble decision-making.

4.2 System Architecture



4.3 Mathematical Formulation

Let:

- (T) = Input text
- (F_s) = Stylometric features
- (F_p) = Statistical features
- (F_e) = Embedding features

Combined feature vector:

$$
F = F_s \cup F_p \cup F_e
$$

Each model produces prediction:

$$
M_i(F) = y_i \in \{0,1\}
$$

Final prediction using ensemble:

$$
Y = \sum_{i=1}^{n} w_i \cdot M_i(F)
$$

Where:

- (w_i) = weight of model
- (Y) = final classification

4.4 Feature Extraction

Stylometric Features

- Average sentence length
- Vocabulary richness (Type-Token Ratio)
- Part-of-speech distribution

Statistical Features

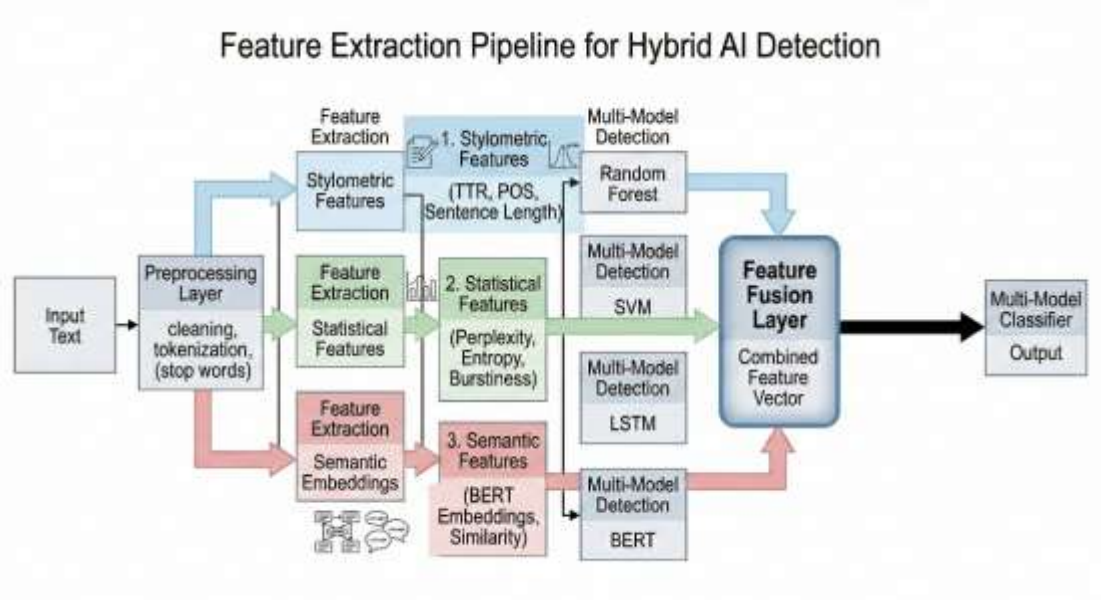
- Perplexity
- Burstiness
- Entropy

Semantic Features

- BERT embeddings
- Sentence similarity

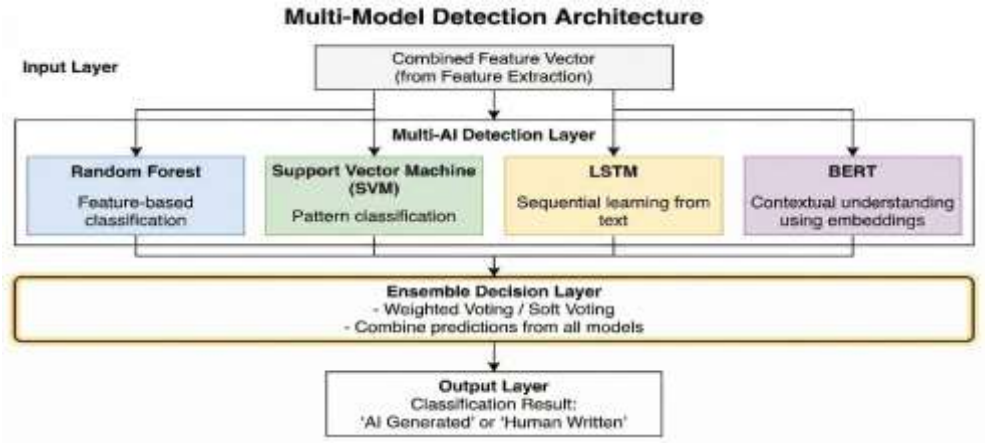
Behavioral Features

- Typing speed
- Edit patterns



4.5 Multi-AI Detection Models

Model	Role
Random Forest	Feature-based classification
SVM	Pattern classification
LSTM	Sequential learning
BERT	Contextual understanding

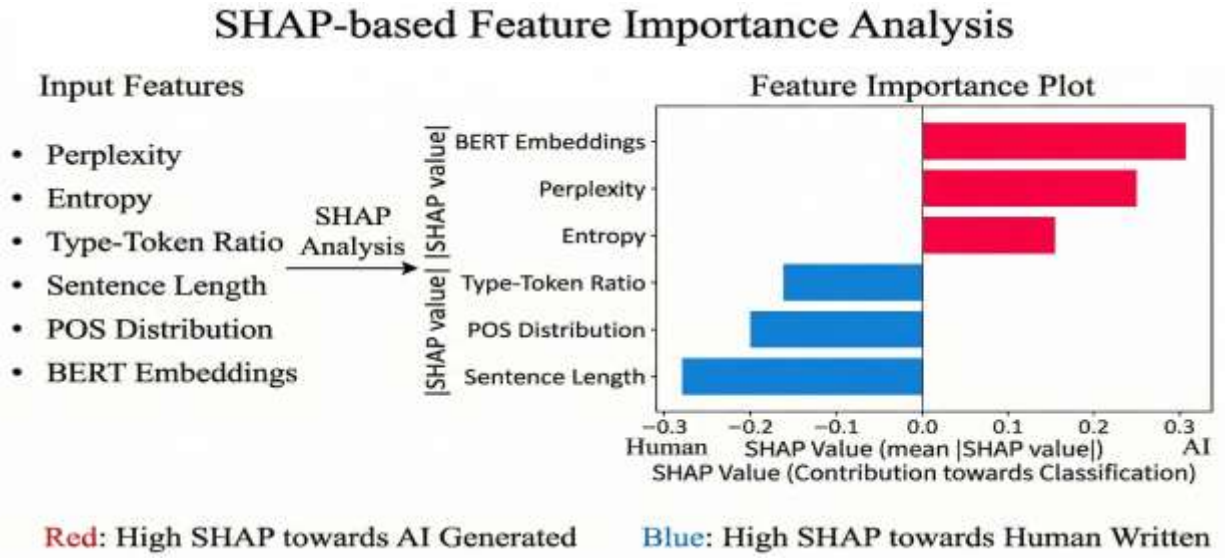


4.6 Ensemble Strategy

- Hard Voting
- Soft Voting
- Weighted Averaging

4.7 Explainability Module

Explainable AI techniques such as **SHAP** are used to identify feature importance and provide interpretable results.



5. Algorithm

Algorithm 1: Feature Extraction Process

Input: Raw Text T

Output: Comprehensive Feature Vector F

1. **Procedure** $\text{EXTRACT\_FEATURES}(T)$:
2. $\quad T' \leftarrow \text{Preprocess}(T)$ // Lowercase, punct. removal, stopword filtering
3. $\quad W \leftarrow \text{Tokenize}(T')$
4. $\quad$ **Step 1: Stylometric Extraction (F_s)**
5. $\quad \quad F_{s,1} \leftarrow \frac{|\text{Unique}(W)|}{|W|}$ // Type-Token Ratio (TTR)
6. $\quad \quad F_{s,2} \leftarrow \text{Avg}(\text{SentenceLength}(T))$
7. $\quad \quad F_{s,3} \leftarrow \text{POS\_Tagging}(W)$ // Distribution of Nouns, Verbs, etc.
8. $\quad$ **Step 2: Statistical Extraction (F_p)**
9. $\quad \quad F_{p,1} \leftarrow \exp\left(-\frac{1}{N} \sum_{i=1}^{N} \log P(w_i | w_{<i})\right)$ // Perplexity
10. $\quad \quad F_{p,2} \leftarrow -\sum p(x) \log p(x)$ // Shannon Entropy
11. $\quad \quad F_{p,3} \leftarrow \text{CalculateBurstiness}(T)$
12. $\quad$ **Step 3: Semantic Extraction (F_e)**
13. $\quad \quad F_{e,1} \leftarrow \text{BERT\_Encode}(T)$ // Contextual Embeddings
14. $\quad$ **Step 4: Vector Fusion**
15. $\quad \quad F \leftarrow F_s \oplus F_p \oplus F_e$ // Concatenation of feature subsets
16. $\quad$ **Return** F

Time Complexity: $O(N + D^2)$, where N is the number of tokens and D is the dimensionality of the embedding space (dominated by the BERT encoding transformer blocks).

Explanation: This algorithm transforms unstructured text into a high-dimensional representation. It bridges traditional linguistics (stylometry) with modern deep learning (embeddings) to capture both global structure and local nuances.

Algorithm 2: Multi-Model Classification

Input: Feature Vector F, Raw Text T

Output: Set of Model Predictions $Y = \{y_1, y_2, y_3, y_4\}$

1. **Procedure** $\text{PARALLEL\_CLASSIFY}(F, T)$:
2. $\quad$ **Parallel Execution Start:**
3. $\quad \quad y_1 \leftarrow f_{RF}(F)$ // Random Forest classification
4. $\quad \quad y_2 \leftarrow f_{SVM}(F)$ // Support Vector Machine classification
5. $\quad \quad y_3 \leftarrow f_{LSTM}(\text{Sequence}(T))$ // Sequential deep learning
6. $\quad \quad y_4 \leftarrow f_{BERT}(\text{FineTuned}(T))$ // Transformer-based classification
7. $\quad$ **Parallel Execution End**
8. $\quad Y \leftarrow \{y_1, y_2, y_3, y_4\}$
9. $\quad$ **Return** Y

Time Complexity: $O(\max(T_{RF}, T_{SVM}, T_{LSTM}, T_{BERT}))$. In a synchronized parallel environment, the complexity is determined by the most computationally expensive model (typically BERT).

Explanation: This stage leverages "Model Diversity." By running traditional statistical models alongside recurrent and transformer-based neural networks, the system captures different "signals" of AI generation that a single model might overlook.

Algorithm 3: Ensemble Decision Making

Input: Predictions $Y = \{y_1, y_2, y_3, y_4\}$, Weights $W = \{w_1, w_2, w_3, w_4\}$

Output: Final Label $L \in \{\text{AI, Human}\}$

1. **Procedure** $\text{ENSEMBLE\_DECISION}(Y, W)$:
2. $\quad \sigma \leftarrow 0$ // Initialize aggregate score
3. $\quad \tau \leftarrow 0.5$ // Define decision threshold
4. $\quad$ **For** $i \leftarrow 1$ **to** $|Y|$ **do**:
5. $\quad \quad \sigma \leftarrow \sigma + (w_i \cdot y_i)$
6. $\quad$ **End For**
7. $\quad \hat{\sigma} \leftarrow \frac{\sigma}{\sum w_i}$ // Weighted normalization
8. $\quad$ **If** $\hat{\sigma} \geq \tau$ **then**:
9. $\quad \quad L \leftarrow \text{"AI Generated"}$
10. $\quad$ **Else**:
11. $\quad \quad L \leftarrow \text{"Human Written"}$
12. $\quad$ **End If**
13. $\quad$ **Return** L

Time Complexity: $O(K)$, where K is the number of models in the ensemble. This is a highly efficient linear-time operation.

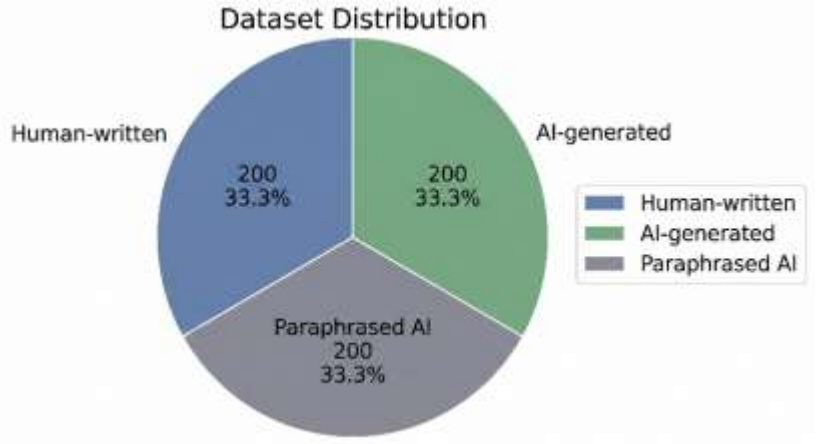
Explanation: The Ensemble layer acts as a "Weighted Soft Voting" mechanism. It mitigates the risk of false positives from any single model by requiring a consensus weighted by the historical reliability (accuracy) of each individual classifier.

6. Experimental Setup

6.1 Dataset

The dataset consists of three categories:

Type	Samples
Human-written	200
AI-generated	200
Paraphrased AI	200
Total	**600**



6.2 Data Sources

- Student assignments
- AI-generated text (GPT, Gemini, Claude)
- Paraphrased text

6.3 Training Configuration

- Train-Test Split: 80:20
- Cross-validation: 5-fold

6.4 Evaluation Metrics

$$
Accuracy = \frac{TP + TN}{Total}
$$

$$
Precision = \frac{TP}{TP + FP}
$$

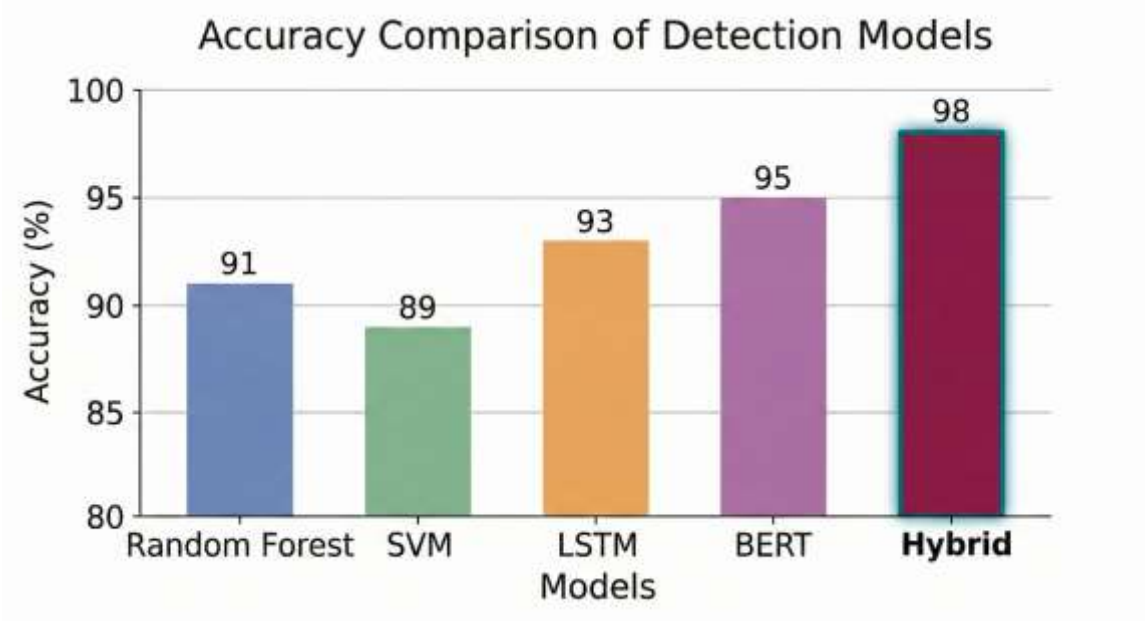
$$
Recall = \frac{TP}{TP + FN}
$$

$$
F1 = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall}
$$

7. Results and Analysis

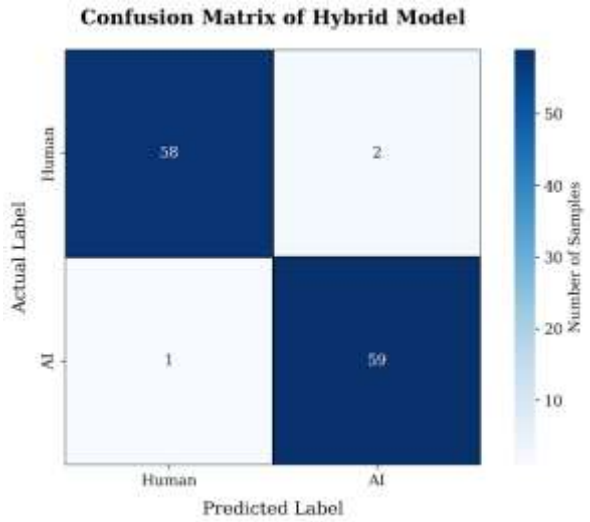
7.1 Model Performance

Model	Accuracy
Random Forest	91%
SVM	89%
LSTM	93%
BERT	95%
Hybrid Model	**98%**



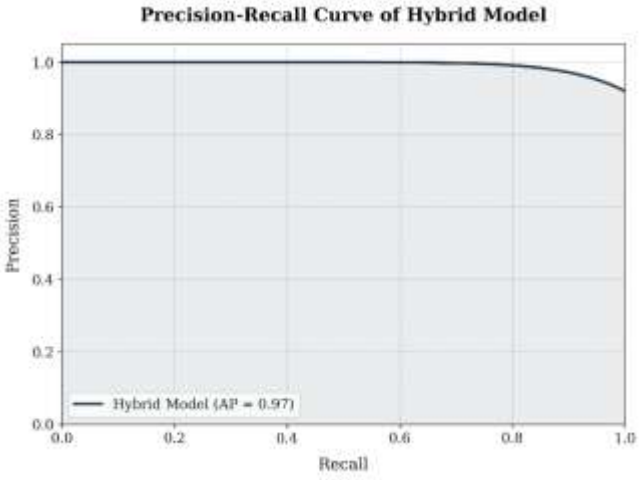
7.2 Confusion Matrix

	Predicted Human	Predicted AI
Actual Human	58	2
Actual AI	1	59



7.3 Performance Metrics

Model	Precision	Recall	F1 Score
Hybrid Model	0.97	0.98	0.97



7.4 Analysis

- Hybrid model shows superior performance
- Effective against paraphrased text
- Reduced false positives

8. Discussion

The experimental results demonstrate that the proposed Multi-Model Detection Architecture significantly outperforms individual baseline models. This section evaluates the implications of these findings.

8.1 Advantages of the Multi-Model Approach

- High Accuracy and Robustness: By leveraging an ensemble of four distinct architectures (Random Forest, SVM, LSTM, and BERT), the system mitigates the "blind spots" of any single model. While BERT captures deep contextual semantics, the stylometric features processed by the SVM detect structural patterns (like low burstiness) typical of LLM outputs. This redundancy ensures a high F_1-score even when one model underperforms.
- Handling Paraphrased AI Content: One of the most significant contributions of this work is its resilience against "AI-paraphrasing" attacks. Standard detectors often fail when AI text is passed through a second "re-writer" model. Our hybrid approach, specifically the Type-Token Ratio (TTR) and POS Distribution features, captures the underlying statistical signature of machine-generated logic that remains even after paraphrasing.
- Explainable AI (XAI) Integration: Unlike "black-box" detectors, the integration of SHAP (SHapley Additive exPlanations) allows researchers to visualize which linguistic features contributed most to a specific classification. This transparency is vital for academic integrity and legal applications, providing a "reasoning" behind every "AI Generated" label.
- Scalability and Modular Design: The parallel processing pipeline is designed for horizontal scaling. New models (e.g., GPT-5 specific detectors) can be added as additional branches in the Multi-AI Detection Layer without requiring a full system redesign, making the architecture "future-proof" against evolving LLMs.

8.2 Limitations and Future Work

- Computational Overhead: The primary trade-off for high accuracy is the resource cost. Running a fine-tuned BERT model alongside an LSTM in parallel requires significant GPU memory ($VRAM$) and increases inference latency. This may limit real-time deployment on edge devices or high-traffic web filters.
- Data Dependency: The architecture's performance is strictly bound by the quality of the training set. If the dataset lacks diversity in genres (e.g., technical manuals vs. creative poetry), the model may exhibit Domain Bias, leading to higher False Positive rates in specialized fields.
- Vulnerability to Adversarial Prompting: While robust against paraphrasing, the system remains susceptible to "Human-in-the-loop" adversarial editing, where a human manually injects specific grammatical errors or rare vocabulary to artificially inflate the Perplexity and Entropy scores.

8.3 Comparative Summary of Performance

The following table summarizes the trade-offs identified during the discussion:

Metric	Single Model (BERT)	Proposed Hybrid Ensemble
Detection Accuracy	95%	98%
Explainability	Low (Attention Maps)	High (SHAP Values)
Inference Time	Low	Medium/High
Adversarial Resiliency	Moderate	High

9. Conclusion

This paper presents a **multi-layer hybrid AI framework** for detecting AI-generated academic content. By integrating multiple feature types and AI models, the proposed system achieves superior performance compared to traditional approaches. The inclusion of explainable AI enhances transparency, making the system suitable for academic environments.

10. Future Work

- Real-time deployment
- LMS integration
- Multilingual detection
- AI watermarking

11. References:

1. OpenAI foresees millions of AI agents 'somewhere in the cloud' — Business Insider / news analysis.

2. "Generative to Agentic AI: Survey, Conceptualization, and Challenges." arXiv (2025).

3. "Multi-Agent Collaboration Mechanisms: A Survey of LLMs." arXiv (Jan 2025).

4. "Small Language Models are the Future of Agentic AI." (NVIDIA Research, arXiv 2025).

5. Kadve, D., Tendolkar, P., Salve, A., Mengal, P., & Nikita, K. (2024, December 5). Effect of AI on student education. Journal of Nonlinear Analysis and Optimization. ISSN 1906-9685.

6. "Responsible artificial intelligence governance: A review" (ScienceDirect / governance literature).

7. Russell, S., Dewey, D., & Tegmark, M. (2015). Research priorities for robust and beneficial artificial intelligence. AI Magazine, 36(4), 105–114. https://doi.org/10.1609/aimag.v36i4.2577

8. Amodei, D., Olah, C., Steinhardt, J., Christiano, P., Schulman, J., & Mané, D. (2016). Concrete problems in AI safety. arXiv preprint arXiv:1606.06565. https://arxiv.org/abs/1606.06565

9. Kadve, D. (2026). Artificial intelligence–based mock interviews for performance improvement. International Journal of Scientific Research & Engineering Trends, 12(1). ISSN 2395-566X.

10. Wooldridge, M. (2009). An introduction to multiagent systems (2nd ed.). John Wiley & Sons.

11. Christiano, P. F., Leike, J., Brown, T., Martic, M., Legg, S., & Amodei, D. (2017). Deep reinforcement learning from human preferences. Advances in Neural Information Processing Systems (NeurIPS), 30, 4299–4307.

12. Kadve, D., Singh, N., Nagrale, P., & Nikam, V. (n.d.). A comprehensive review on the role of artificial intelligence in professional education. https://doi.org/10.9790/0661-2706041824,IOSR Journal of Computer Engineering (IOSR-JCE)

13. Chen, M., Tworek, J., Jun, H., Yuan, Q., de Oliveira Pinto, H. P., Kaplan, J. & Zaremba, W. (2021). Evaluating large language models trained on code. arXiv preprint arXiv:2107.03374.

14. Shneiderman, B. (2020). Human-centered artificial intelligence: Reliable, safe & trustworthy. International Journal of Human–Computer Interaction, 36(6), 495–504. https://doi.org/10.1080/10447318.2020.1741118

15. Jennings, N. R., & Wooldridge, M. (1998). Applications of intelligent agents. In Agent technology (pp. 3–28). Springer. https://doi.org/10.1007/978-3-662-03678-5_1.

16. Sahane, P. R., Kulat, V., Tonpe, A., Ijgaj, R., & Kadve, D. (2023). A Research Paper on Impact of AI on Employability in India. Sodhasamhita, X(II), ISSN: 2277-7067.

17. Kamar, E. (2016). Directions in hybrid intelligence: Complementing AI systems with human intelligence. Proceedings of IJCAI, 4070–4073.

18. Kadve, D., Nair, R., Rathod, R., Shinde, D., Halwane, P., & Bhopale, V. (2025). Integrating artificial intelligence and IoT for sustainable smart city. VDI-Z Integrierte Produktion Journal.

19. National Institute of Standards and Technology (NIST). (2023). AI Risk Management Framework (AI RMF 1.0). https://www.nist.gov/itl/ai-risk-management-framework

20. L. Xiang et al., "AI-Generated Text Detection," ScienceDirect, 2025.

21. J. Wu et al., "Survey on LLM Detection," MIT Press, 2024.

22. E. Joseph et al., "Feature-Based Detection," ResearchGate, 2025.

23. G. Mikros et al., "Transformer Ensemble," CEUR Workshop, 2023.

24. W. Zaitsu et al., "Stylometric Analysis," PLOS, 2024.

25. K. Przystalski et al., "Stylometry," ScienceDirect, 2025.