VFAST Transactions on Software Engineering
https://vfast.org/journals/index.php/VTSE@ 2026, ISSN(e): 2309-3978, ISSN(p): 2411-6246
Volume 14, Number 1, January-March 2026 pp:108-123

Distinguishing Human-Generated and AI-Generated Academic Writing: A Machine Learning Benchmark Study

Ali Raza[1], **Mohib Ullah**[1], **Rafiullah Khan**[1*], **Adeem Ali Anwar**[2], **Muhammad Inam Ul Haq**[3], **Shazia Riaz**[4]

[1]The University of Agriculture, Peshawar, 25130, Pakistan; [2]Kent Institute, 10 Barrack St, Sydney, 2000, NSW, Australia; [3]Khushal Khan Khattak University, Karak, 27200, Pakistan; [4]Department of Computer Science, Government College Women University Faisalabad, Faisalabad, 38000, Pakistan

Keywords:
AI-authorship Detection, Academic Integrity, Machine Learning, Transformer Models, Text Classification.

Journal Info:
Submitted:
November 02, 2025
Accepted:
February 15, 2026
Published:
February 28, 2026

Abstract The rapid adoption of large language models (LLMs) such as ChatGPT has raised critical questions about authorship, originality, and integrity in academic writing. Unlike conventional plagiarism testing tools, AI-generated or AI-rephrased text can preserve the original meaning and context of the text while modifying the writing style, making it challenging to detect using standard similarity checks. This study addresses this challenge by creating a domain-specific corpus of postgraduate-level academic texts. The corpus contains 22,520 samples, equally divided between human-written text and AI-rephrased text. All samples were preprocessed and represented using two common techniques: TF-IDF and Word2Vec. The dataset was evaluated using well-known machine learning and deep learning models, including Logistic Regression, Support Vector Machines, Recurrent Neural Networks, and transformer-based models BERT and T5. The results show that linear and sequential models provide low baseline performance, with accuracy between 50-54%. While BERT significantly outperforms the other models, achieving 83% precision along with a high recall rate. Confusion matrix analysis further shows that traditional models tend to overpredict AI authorship, whereas BERT demonstrates strong reliability in distinguishing between human-written and AI-generated text. The results show that transformer-based models are more effective for authorship verification in academic settings. They also emphasize the trade-offs among interpretability, computational cost, and predictive performance. In general, this study offers some important recommendations for the creation of credible, transparent, and domain-sensitive AI detectors for academia.

***Correspondence author email address:** rafiyz@gmail.com
DOI: 10.21015/vtse.v14i1.2274

1 Introduction

Consider a situation in which the professor is reading an essay, which is apparently ideal, clear, tidy, and stylistically consistent. However, at the same time, the professor did not know whether it was written by the student or by a large language model (LLM) chatbot such as ChatGPT. Although this may appear to be a hypothetical scenario, such scenarios are increasingly becoming a reality. The contrast between human-generated and AI-generated or AI-rephrased text now presents a new challenge for the academic community. This development is likely to influence education, scholarly practice, and the

broader knowledge ecosystem badly.

The introduction of large language models (LLMs) has completely revolutionized the text writing process for both academic and professional documentation. New AI models can generate text that is consistent, stylistically similar, and often identical to human-generated text. Despite their undeniably positive effects, such as enhanced productivity, language acquisition, and automated routine writing, their application in academic environments raises serious ethical and practical concerns. The need to maintain originality and intellectual authorship remains a significant ethical and practical issue. [1, 2].

One of the main challenges is the ability of such systems to produce or rephrase text that cannot be detected by traditional plagiarism detection methods. Traditional tools use surface-level similarity measures, e.g., string matching or overlap of n-grams with known sources [3]. Conversely, AI-generated or AI-plagiarized texts seem to be stylistically different even in the case of copying ideas. This slight imitation blurs the distinction lines between genuine research and synthetic reproduction, increasing the distrust of the peer review and the reliability of knowledge production.

The issue is further intensified by the free, easy, and widespread availability and accessibility of these AI-based tools. Tools such as Copilot, Gemini, and ChatGPT are popular and capable of recreating entire documents with minimal effort. This makes it increasingly challenging for educators and publishers to determine and authenticate the authorship of a document. Previous studies have established that robust, data-efficient, interpretable, and versatile detection systems are required to address these issues [4, 5, 7, 8]. However, existing strategies are struggling to remain effective due to continuous improvements in the contextual reasoning and fluency of modern LLMs.

Another major limitation in current research is its focus on generic web text rather than formal academic writing. Scholarly texts at the postgraduate level are usually domain/discipline-specific, linguistically dense, and structurally complex. These characteristics create a more demanding evaluation setting for authorship detection and are directly relevant to high-stakes domains such as scientific publishing, higher education, policy development, and journalism. Therefore, addressing this gap is very important for preserving trust, especially in environments where authentic authorship is critical.

In this research, we systematically evaluate various machine learning and deep learning models to determine their capability to differentiate between human-generated academic writing and AI-generated text. Unlike proprietary "black-box" detection systems, we focus on methodological transparency and reproducibility of the experiments. We examine the performance of classical approaches (Logistic Regression and Support Vector Machines (SVM)), deep learning models (Recurrent Neural Networks), and a transformer-based model (BERT). The evaluation is carried out on a domain-specific dataset comprising genuine student-written text and AI-generated and rephrased text.

The results show that the transformer-based model BERT outperformed all other employed models, achieving an accuracy of 81.4% and a precision of 83%. The key contributions of the paper are as follows:

1. Development of an academic writing domain-specific dataset containing student-authored academic text and AI-generated and paraphrased text.
2. Comparison of deep learning and traditional machine learning models using two different feature extraction strategies: TF-IDF and Word2Vec.
3. Evaluation of trade-offs between interpretability, computational efficiency, and accuracy.
4. Empirical demonstration of the effectiveness of different models in recognizing AI-generated text.

The rest of this paper is structured as follows. Section 2 examines current methods of AI text detection. Section 3 explains our dataset and preprocessing pipeline. Section 4 defines the benchmarked models and evaluation metrics. Section 5 outlines the experimental settings. The findings and discussion of the experiments are presented in Section 6. Finally, Section 7 concludes with implications for research, practice, and policy.

2 Literature Review

The challenge of effectively differentiating human-generated text from AI-generated text has gained attention since the launch of ChatGPT at the end of 2022.

This surge in research interest is largely due to the LLM's ability to generate text that is not only grammatically correct but also contextually relevant and stylistically coherent across various fields [8].

2.1 Early Statistical and Lexical Feature-Based Approaches

Initial detection methods used basic statistical and lexical features of text, including the frequency of words used, the probability of tokens, or unusual syntactic structure. Such as Gehrmann et al, applied logistic regression to word likelihood distributions based on GPT-2 outputs and obtained fairly good results on smaller models [4]. Nonetheless, it did not work with larger and more fluent LLMs. Similarly, Huang et al. found that the AI text generation tends to have semantic flow, particularly when considering lengthy text passages, i.e., a flaw that some early detectors occasionally exploited [3].

During this period, Kehkashan et al. proposed the Giant Language Model Test Room (GLTR), created by [8], a visualization-oriented tool that enabled users to inspect documents at the token-level probability distribution of varying likelihood levels. It improved non-expert human performance on the task of human detection to 72%. Its effectiveness, however, did not take long to collapse when applied to newer or fine-tuned LLMs due to its dependence on the output distributions of a fixed model [10].

2.2 Transformer-Based Detection Models

Transformer-based architecture was introduced, and this was a turning point. BERT models and RoBERTa have become much more effective at detection, with contextual and semantic features more fully represented in the model. The value of contextual embeddings in generating and detecting fake news was demonstrated by [11], showcasing Grover, a transformer that performs both generation and detection of fake news. Subsequently, RoBERTa models have been fine-tuned by [4] and others and have reached almost state-of-the-art performance in GPT-2 detection.

Extensive reviews [8] verify that, under controlled conditions, transformer-based detectors can achieve F1-scores greater than 0.99. [12] applied this research to multilingual environments and trained RoBERTa and ELECTRA on text detection in English and French. These models showed strong in-domain results (more than 99% and 94% F1-scores) but were weak with out-of-domain texts, which points to a continuing problem with generalization.

Another major direction is the inclusion of explainable AI (xAI). Tools such as SHAP or LIME could be superimposed on transformer pipelines to identify which stylistic features (such as lexical diversity or readability indices) have the highest association with AI authorship, as shown in [16]. This not only enhances accuracy but also provides greater transparency.

2.3 Perplexity-Based and Model-Intrinsic Methods

Another strong line of research focuses on perplexity. Because AI text generation is capable of producing fewer random tokens, HowkGPT [20] is one of the systems that uses distribution- and context-based thresholds to identify suspicious sentences, particularly in scholarly writing. Though useful with original content, these techniques are unsuccessful when texts are changed or paraphrased [8]. White-box methods go further by reflectively analyzing model internals. DetectGPT, which detects passages with the help of a model based on log-probability curvature, is an example. Despite the promise of these approaches, they tend to be difficult to scale across architectures and are strongly tied to the specific model on which they are constructed [10].

2.4 Sequential and Hybrid Architectures

Along with transformers, there are also sequence models. LSTMs and BiLSTMs have been used to represent the flow of text [13], e.g., suggested a TSA-LSTMRNN, which was a composite of a sequence model with simulated annealing-based optimization, and reached a level of accuracy of generating more than 93 percent on human and ChatGPT datasets. But, observing, as it is seen in [8] note, even optimized RNNs are likely to perform worse than transformers with hints of style in longer or more semantically material writing.

Combination techniques are attempting to fuse the strengths of both paradigms. As an example, [14] added GLTR contextualised, inspired statistical features. RoBERTa has embeddings that are capable of giving the model will to be made more successful in detecting antagonistic paraphrasing of inputs. These kinds of

hybrids mean that the strategy of combining the two resistances would be made worse by shallow and deep cues to paraphrasing attacks.

2.5 Feature Engineering and Explainable AI

In addition to architecture, feature engineering is useful in improving interpretability. Krishna et al. demonstrated that standard linguistic measures, such as sentence length or punctuation rule, vocabulary richness (e.g., Yule's K), can be used as effective indicators in conjunction with modern embeddings [9]. Such features not only enhance the predictive performance of the tool but also give human explanations and the reasons why a text is marked as an AI-generated text.

Explainability is not a choice, especially in areas that matter, such as education [8]. Educators and schools should be in a position to explain the reason why a piece of reading was flagged to prevent the unfair demotivation of learners. The techniques, such as SHAP [16], allow defining the difference between acceptable stylistic features and actual AI indications.

2.6 Limitations of the State of the Art

Although there have been significant improvements seen recently, there are still gaps. A significant part of the current literature has been experimented with general web text, and this restricts its use to academic writing. Cross-domain robustness is poor: detectors that have been trained in one domain tend to fail when used on another. There are numerous commercial tools that continue to exhibit both high false positive and false negative rates in academia, including ZeroGPT [8].

Both reviews and empirical literature point out that the future in this area will be based on the development of domain-specific repositories and adaptive architecture. Free-flowing evaluation frameworks that can keep pace with generative frameworks are the best option for AI-generated text detection problems.

The focus of this study is to fill these gaps by:

1. Creating a domain-specific and relevant data repository of human-generated academic text.
2. Comparison between the traditional classifiers (Logistic Regression and SVM) with deep learning models (RNN and BERT).
3. Comparison of the effect of feature extraction techniques (TF-IDF vs. Word2Vec) on model performance.
4. Recommending the methods and techniques that are best suited to educational integrity environments.

In short, although previous studies exhibit that transformer-based strategies bear promising results, there is an urgent need to conduct task-specific and context-focused evaluations. This paper is based on this premise, where several machine learning pipelines are put to the test in the scholarly sector, where themes of originality and authorship count are especially elevated.

To investigate the usefulness of various machine learning models in the process of detecting human-generated and AI-generated academic writing. We designed a customized binary classification dataset [30]. The dataset is constructed to mirror the actual-life features of postgraduate academic writing, including linguistic and stylistic patterns of academic writing.

3 Dataset and Preprocessing

3.1 Dataset Collection

To support empirical evaluation, a domain-specific and extensive dataset is prepared by compiling post-graduate academic reports that are submitted by the post-graduate students of ICS/IT, the University of Agriculture, Peshawar, as indicated in Table 1 and presented in Figure 1. Based on these reports, 11,260 paragraphs written by humans on different topics are selected. In total, these texts included 1,989,000 words, 49,673 sentences, and 535 separate sections.

Every original paragraph is rewritten using ChatGPT (version GPT-3.5 Turbo). This generated 11,260 AI-generated paragraphs, 2,073,000 words, 62,190 sentences, and 691 sections. The slight difference in the number of paragraphs was caused by the limitations in rephrasing, including the use of tokens or the formatting of the materials.

The choice of postgraduate-level writing is deliberate. As the academic texts written on this level are more complicated, contain more difficult vocabulary, longer sentences, and are strict to disciplinary terminologies. Such attributes are much harder to realistically imitate and, thus, offer a more potent testbed to assess AI detection

systems. Advanced academic writing will also be used to make sure that the dataset is directly applicable to the high-stakes educational settings, where the authorship and originality will be very important.

3.2 Dataset Labelling

The dataset consists of binary labelling of each paragraph, i.e., each paragraph is either a human-generated academic text or rephrased by an AI tool. Overall, the dataset is composed of 22,520 annotated paragraphs, which provide an adequate base for training and testing of machine learning classifiers.

Table 1. Dataset composition by category.

Category	Academic (Human)	Rephrased (AI)
Paragraphs	11,260	11,260
Words	1,989,000	2,073,000
Sentences	49,673	62,190
Sections	535	691

The table presents dataset statistics comparing Human-Written and AI-Rephrased academic texts.

3.3 Text Preprocessing

All the samples are subjected to a standardized preprocessing pipeline before training, which aims to normalize linguistic variability and reduce noise. This helps the models focus on substantive patterns rather than superficial variations in the texts. Some important steps involved in the pipeline are as follows:

In the first step, all text is converted to lowercase to remove case sensitivity. This has the advantage of treating words like "education" and "Education" as the same word and eliminating redundant feature representations. In the second step, all punctuation marks are removed to minimize format discrepancies that add little to semantic meaning but can enlarge the feature space, particularly in vector representations.

Lastly, we tokenize the text using the tokenizer library of NLTK. In this step, the text is divided into sentences and words for feature extraction. The features are extracted using Word2Vec and TF-IDF techniques.

Two feature extraction strategies are used after preprocessing:

3.3.1 TF-IDF Vectorization

TF-IDF detects differences in surface-level word usage and allows models to detect lexical choices that may be used to differentiate between human-generated and AI-generated text. Figure 2 shows TF-IDF features of human-generated text, and the text is rephrased by AI. The fact is justified by these discrepancies that TF-IDF is able to hold discriminative surface characteristics which can be used to affect classification [3].

3.3.2 Word2Vec Embeddings

Word2Vec embeddings are generated using the skip-gram architecture with a window size of 5 and a vector size of 100. The embedding of every paragraph symbolizes each paragraph and enables models to gain semantic, not merely the patterns of word frequency.

TF-IDF and Word2Vec can be utilized as a complementary view: the former is much more concerned with the stylistic and lays stress on semantic coherence. This combination can be utilized for stronger detection models assessment.

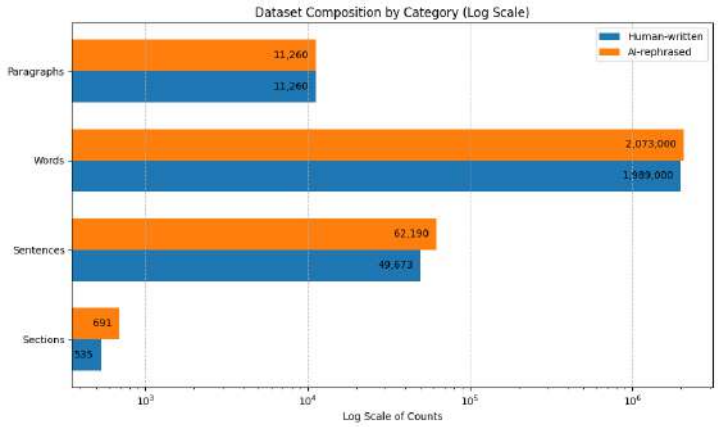

Figure 1. Dataset Composition by Category

3.4 Dataset Partitioning

The dataset is split into training, testing, and validation subsets for fair evaluation. The data sampling ensures that the 50:50 class distribution is preserved in all three subsets. 80% data of the dataset is used for training purposes; the rest of the data is used for testing and validation equally. This partitioning of the data ensures that both AI-generated and human-written texts are equally represented in all the subsets, thus allowing models to be trained and evaluated regularly.

The stratification was used to guarantee that human and AI writing were equally distributed in every subset to have their models trained and evaluated regularly. Figure 1 shows this partitioning, in which the majority of the samples were used in the training set, and the same number was used in the validation and testing sets. Such stratified splits are not only widely suggested in text classification research to stabilize performance assessment and extrapolation, but also to offer an equitable contrast of the two [17].

On the whole, this data can be utilized to perform successful testing of authorship detection models. Authentic postgraduate-level writing and AI-rephrased variants of the same can be used to rigorously compare the datasets, as well as the semantic (Word2Vec) and lexical (TF-IDF) feature representations. It helps in the analysis of the traditional, sequential, and transformer techniques. The benchmark methodology of these models is provided in the next section.

4 Methodology

In this paper, we compared various traditional and deep learning classifiers to understand whether an academic paragraph has been written or rephrased by a human or by an AI tool. We make a comparison between Logistic Regression (LR), Support Vector Machines (SVM), Recurrent Neural Networks (RNN), BERT, and T5, which are different in terms of model complexity, explainable features, and reasoning ability. All the models are optimized using TF-IDF features and Word2Vec embeddings to analyze the performance impact of encoding features. The motivations for choosing this combination are as follows:

- It has two extremes, shallow learning (efficiency, interpretability, and base performance) and deep learning (nuanced semantic patterns).
- It provides knowledge on how different model families work in the process of AI detection in academic domain.
- It allows the trade-off evaluation of the computational cost, interpretability, and accuracy, which is critical to be deployed in academic institutions with alternative infrastructure.

4.1 Framework

The experimental framework, summarized in Algorithm 1, establishes a reproducible pipeline for distinguishing human-authored academic text from AI-generated rephrasings. The process begins with a labeled dataset $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{N}$, where x_i denotes the i-th paragraph and $y_i \in \{0, 1\}$ indicates whether the paragraph is human-written (0) or AI-generated (1).

Each paragraph is normalized through a preprocessing operator $\Pi(\cdot)$ that performs lowercasing, punctuation removal, stopword filtering, and tokenization, producing the cleaned representation $\tilde{x}_i$. The normalized text is then transformed into numerical features using two mappings:

- ϕ_{tfidf}, which computes TF-IDF weights $w_t(\tilde{x}_i)$ across vocabulary V;
- ϕ_{w2v}, which generates semantic paragraph embeddings by averaging pretrained word vectors $\mathbf{e}(w) \in \mathbb{R}^{100}$.

For each feature representation, models from the family $\mathcal{F} = \{\text{LR}, \text{SVM}, \text{RNN}, \text{BERT}, \text{T5}\}$ are trained by minimizing the empirical loss $\ell(f_\theta(\mathbf{x}_i), y_i)$ with respect to parameters θ, yielding fitted parameters $\hat{\theta}$. Predictions on the held-out test data produce evaluation metrics including accuracy (A), precision (P), recall (R), and F1-score (F).

Finally, comparative analysis incorporates confusion-matrix inspection together with assessment of interpretability, computational complexity, throughput, and predictive reliability. This structured workflow enables consistent benchmarking across classical, sequential, and transformer-based detection strategies.

The pipeline can be summarized into four stages: preprocessing, feature representation, model training, and evaluation.

4.1.1 Preprocessing

All paragraphs undergo normalization through lowercasing, punctuation removal, stopword elimination, and tokenization. These operations suppress irrelevant stylistic variation while preserving linguistically meaningful structure required for classification.

4.1.2 Feature Representation

Each normalized paragraph is encoded using two complementary approaches. TF-IDF produces sparse, high-dimensional vectors that capture lexical and stylistic vari-

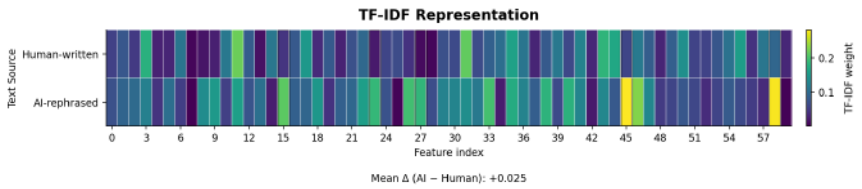

Figure 2. TF-IDF representation of human-written and AI-rephrased text, showing distinct lexical distributions.

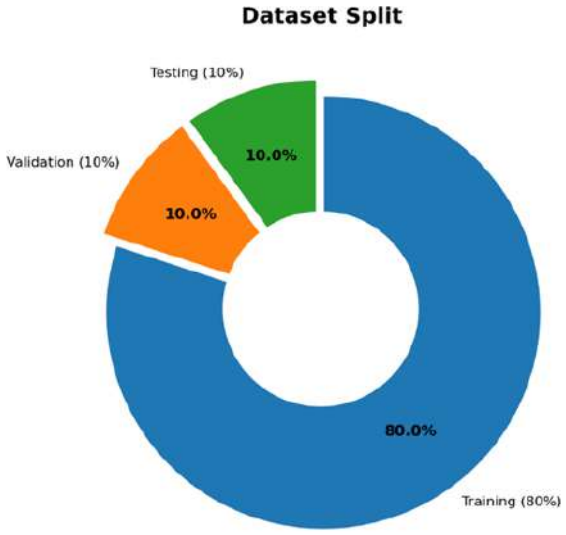

Figure 3. Dataset split into training, validation, and testing subsets

ation, whereas Word2Vec yields dense semantic embeddings reflecting contextual similarity. Comparing these representations allows investigation of whether stylistic or semantic signals better distinguish AI-generated writing.

4.1.3 Model Training

For this research, we selected five benchmark classification models, including Logistic Regression (LR), Support Vector Machine (SVM), Recurrent Neural Network (RNN), Bidirectional Encoder Representations from Transformers (BERT), and Text-to-Text Transfer Transformer (T5). LR and SVM are simple machine learning models that are used as baseline models. RNN model captures the dependencies between words and sequences. Where as BERT and T5 are used to detect deeper authorship patterns using their contextual embedding features. Each model is trained separately using TF-IDF and Word2Vec feature extraction mechanism (where applicable). The training efficiency and resource utilization of these models are summarized in Table 2.

Table 2. Training efficiency and resource utilization of models evaluated on Google Colab (Tesla T4 GPU, 15 GB VRAM, 2 vCPUs, 12 GB RAM).

Model	Training Time (mins)	CPU (GB)	GPU (GB)	Throughput (tokens/s)
LR	0.5	0.6	0.0	42,000
SVM	1.6	1.2	0.0	3,200
RNN	9.2	1.8	2.5	32
BERT (base)	22.0	3.5	7.3	1,450
T5	41.0	4.5	9.2	1,120

4.2 Computational Complexity

A comparison of computational trade-offs of the models in terms of training time, memory usage, and throughput is presented in Figure 4. The results show the dissimilarity between lightweight linear models (Logistic Regression and SVM) and computationally complex deep learning models (RNN, BERT, and T5). Resource-performance trade-offs of the same nature models have

been reported in earlier literature, whereas the capacity to apply large-scale models is mostly dependent on the availability of computational resources and the budget limits [17, 18].

The logistic regression model was the best-performing model in terms of throughput (42,000 tokens/sec) and training time (0.5 minutes), showing its efficiency in high-throughput text classification. SVM had similar performance, requiring 3,200 tokens/sec throughput and 1.6 minutes training time, which made it a feasible lightweight alternative to LR in terms of scale. RNN model took more time as compared to the LR and SVM (9.2 minutes) to train and was the least efficient (32 tokens/sec) because it is sequential and more expensive to compute.

Transformer-based models had notably higher computational demands. BERT (base) needed about 22 minutes of training and had a throughput of 1,450 tokens/sec, which is a good balance of resource-performance ratio. While T5 (small/base) used 9.2 GB of GPU memory and had the slowest training time (41 minutes), it had a consistent throughput (1,120 tokens/sec), which may open up as a solution in situations where a lot of computational resources are accessible and high-quality text generation tasks are required.

In general, these results highlight the significance of model selection based on the computational constraints and task-related performance needs. The lightweight models (LR and SVM) can be used when it is necessary to make fast results and when the resources are limited, and on the other hand, the models based on transformers are more suitable when it is possible to invest more in computing resources and when high accuracy is needed.

5 Experimental Settings

This section outlines the experimental setup used to evaluate authorship detection models. We first describe the datasets employed, including their collection, preprocessing, and feature representation. Next, we present the algorithms selected for comparative study, spanning both traditional and deep learning approaches. Finally, we detail the performance indicators used to assess model effectiveness, ensuring a comprehensive and fair evaluation across methods.

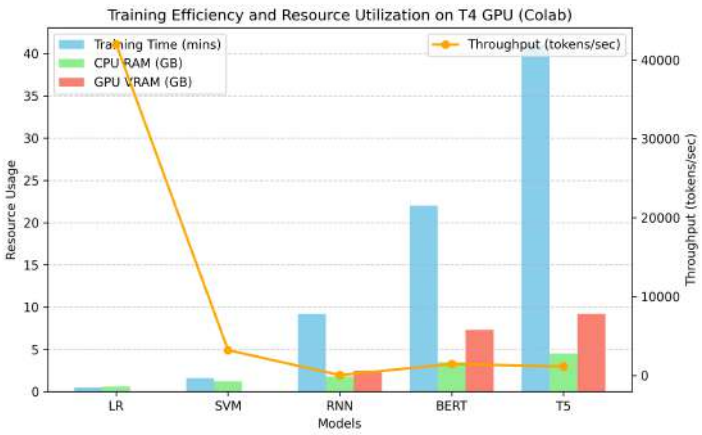

Figure 4. Computational performance comparison of models across three dimensions: training time, GPU memory usage, and throughput

Algorithm 1. Framework for Human- vs. AI-Generated Academic Writing

Input: Corpus $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{N}$ with labels $y_i \in \{0, 1\}$, feature maps Φ, models $\mathcal{F}$

Output: Evaluation metrics $\mathcal{M}(f, \phi)$ for each model–feature pair

Step 1: Preprocessing

foreach $x_i \in \mathcal{D}$ **do**

$\tilde{x}_i \leftarrow \Pi(x_i) = \text{Tok}(\text{Stop}(\text{PuncRm}(\text{Lower}(x_i))))$

Step 2: Feature Representation

for $\phi \in \{\phi_{tfidf}, \phi_{w2v}\}$ **do**

if $\phi = \phi_{tfidf}$ **then**

$\mathbf{x}_i = \phi_{\text{tfidf}}(\tilde{x}_i) = [w_t(\tilde{x}_i)]_{t \in V}$,

$w_t(\tilde{x}_i) = \frac{\text{tf}(t, \tilde{x}_i)}{|\tilde{x}_i|} \cdot \log \frac{N}{1+\text{df}(t)}$

else if $\phi = \phi_{w2v}$ **then**

$\mathbf{x}_i = \frac{1}{|\tilde{x}_i|} \sum_{w \in \tilde{x}_i} \mathbf{e}(w),\ \mathbf{e}(w) \in \mathbb{R}^{100}$

Step 3: Model Training and Evaluation

for $f \in \{LR, SVM, RNN, BERT, T5\},\ \phi \in \{\phi_{tfidf}, \phi_{w2v}\}$ **do**

Train by solving $\hat{\theta} = \arg\min_\theta \sum_{i \in \mathcal{D}_{\text{train}}} \ell(f_\theta(\mathbf{x}_i), y_i)$ Predict $\hat{y}$ on $\mathcal{D}_{\text{test}}$ compute metrics (A, R, P, F)

Step 4: Comparative Analysis

Generate confusion matrices for each (f, ϕ)

Compare trade-offs: interpretability, complexity, throughput, and accuracy

5.1 Corpus Utilization in Experiments

In our experiments, we used our custom-built domain-specific dataset comprised of 22,520 postgraduate-level

academic paragraphs (both human-written and AI-rephrased). Original student reports were used as the source of human-written texts, while their analogies were obtained with the help of ChatGPT (GPT-3.5 Turbo).

Each sample was subject to standard preprocessing steps to provide consistency and minimize the noise. This was followed by two feature extraction approaches: TF-IDF, which was used to extract lexical and stylistic differences, and Word2Vec to learn semantic relationships. To facilitate the process of fair assessment, the dataset was divided into training (80%), validation (10%), and testing (10%) subsets with the distribution of the classes maintained at 50:50 on all the sets. This tedious training made sure that models had been trained and evaluated against balanced and domain-relevant academic writing, so the results can be directly applied to authorship-determination in education.

5.2 Algorithms for Comparative Studies

5.2.1 Logistic Regression

Logistic Regression is a powerful and simple model that is used as a baseline model to predict binary classification. Its linear decision boundary and interpretation coefficient can be used to determine the most influential terms, or features, to be used in academic integrity aspects that require explanation [19].

The complexity of training is $O(nd)$ where n is the number of samples, and d is the number of features, and so LR can scale to large corpora. $O(d)$/sample inference, which makes real-time or large-scale document-level screening possible.

5.2.2 Support Vector Machines

SVMs handle high-dimensional representations and model non-linear class boundaries via kernel functions, often outperforming LR when stylistic distinctions are subtle [19, 20]. Computational cost varies with kernel selection but remains manageable for linear configurations.

5.2.3 Recurrent Neural Networks

RNNs model sequential token dependencies, capturing discourse-level stylistic signals relevant to authorship [21, 22]. Although more expressive than linear models, their sequential computation reduces throughput. An LSTM-based two-layer architecture with dropout of 0.3 is employed to enhance generalization.

5.2.4 BERT

BERT leverages bidirectional self-attention to encode deep semantic and syntactic context, making it well-suited for detecting subtle AI-induced stylistic variation [23, 27]. Computational complexity scales with sequence length and attention layers, necessitating GPU acceleration.

5.2.5 T5

T5 views classification as a text generation task, which may encode the finer contextual relationships. This is beneficial in scenarios where the detection tasks are going to be eventually generalized to the explanatory or multi-task cases [26].

It is as complex as BERT and, more generally, due to its encoder–decoder architecture, requires substantial resources for processing both input and output sequences. T5 is also resource-intensive, requiring large GPU memory and compute time; unless optimized, it is not very practical in resource-constrained environments.

5.3 Performance Indicators

Model effectiveness is evaluated using standard classification metrics [25, 26]:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

$$\text{Precision} = \frac{TP}{TP + FP} \tag{2}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{3}$$

$$\text{F1} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{4}$$

6 Results and Analyses

In this section, we analyze and discuss the performance of all the machine learning and deep learning models used in this research. Models are evaluated in terms of accuracy, precision, recall, and F1-score.

we also present a confusion matrix analysis, examine the effect of training and testing splits, provide a detailed discussion of all models, and compare our findings with previous studies.

6.1 Experimental Results

Table 3 summarizes the performance of all the models and sets of features. Logistic Regression and SVM models as an interpretable baseline, RNN captures sequential dependencies, while T5 and BERT represent transformer-based contextual architectures.

The performance of the Logistic Regression was modest compared to the other employed models. However, the accuracy of LR models with TF-IDF was slightly higher compared to Word2Vec (53.92% and 51.17% respectively). The sparse version of TF-IDF with the frequency-based representation appears to be more appropriate to represent surface-level stylistic and lexical differences, whereas Word2Vec embeddings can be more prone to generalizing their semantics. Even though the accuracy of LR is low, it has great interpretability and thus can serve as a good baseline model in academic integrity applications where transparency is vital.

As compared to LR, the performance of SVM was a bit lower both in TF-IDF (50.12% accuracy) and Word2Vec (49.88% accuracy). However, the recall rate of Word2Vec (97.38%) was way higher than the recall rate of LR Word2Vec and TF-IDF. The Word2Vec variant was therefore more likely to consider almost everything as AI-generated, which overfits false positives. The high-dimensional sparse features of TF-IDF capitalized more on the ability of hyperplane separation of SVM.

The RNNs reached a perfect recall with the text generated by AI (100%) but failed to recognize any paragraphs written by humans (TN = 0) and had a 100% perfect recall, with an apparent specificity of 0. TF-IDF and Word2Vec inputs gave a similar result. This extreme bias implies that the learned decision boundaries are unstable, probably because of representational imbalance. Although the high recall can be helpful when the AI content is being filtered (no AI version is overlooked), the absolute absence of precision makes RNN impractical for academic evaluation.

BERT has an accuracy of 81.4%, precision of 83%, recall of 80%, and F1-score of 79%, which is significantly better than all employed baseline and neural network models. This implies a high capability of differentiating human and AI-written text by bidirectional semantic relationship modeling over sentence structures. Contextual representations of BERT also represent subtle elements of style that are not well represented by manually designed features or sequential models.

Based on these evaluation results, the BERT model is fit and recommended for academic evaluation.

T5 model exhibits irregular behavior by demonstrating higher recall for human-generated text (69%) compared to AI-generated text (53%). This implies the bias of the model for human-generated text. Although it attained a decent balance precision (62%). Overall performance was worse than that of BERT, which indicates that task-specific fine-tuning of generative architectures is necessary when reusing them on classification.

6.2 Confusion Matrix Analysis

To better understand classification behavior, confusion matrices were examined for each model (Fig. 5). Logistic Regression showed consistent baseline behavior but produced relatively high false-positive counts, particularly when paired with Word2Vec. SVM slightly reduced false positives with TF-IDF, yet again showed over-prediction of the AI class when dense embeddings were used.

RNN results revealed complete bias toward AI classification, achieving perfect recall but zero specificity. Such an imbalance confirms that sequential modeling alone is insufficient for reliable authorship discrimination in academic text.

BERT model provided the balanced confusion matrix as compared to the rest of the employed models, with a number of true negatives and a low number of false positives. Despite some false negatives and false positives, the model achieved a favorable balance between fairness and sensitivity. This makes it more suitable for academic evaluation.

T5 model, on the other hand, successfully identified many AI-generated text samples but did not perform well in detecting human-generated text. This behavior shows that fine-tuning and additional calibration of the model are needed when reusing them on classification.

In a nutshell, the process of changing a traditional linear model into sophisticated deep learning structures depicts the trade-off between interpretability and predictive power. Although the transparency of simpler models, such as the LR and SVM, is provided, deep learning models such as BERT provide much better performance

Table 3. Performance of models across feature sets (TF-IDF and Word2Vec). Metrics are reported on the held-out test set. Best results per column are highlighted in bold.

Model & Feature Set	Accuracy	Precision	Recall	F1-score
Logistic Regression (Word2Vec)	0.5117	0.5327	0.8142	0.6440
Logistic Regression (TF-IDF)	0.5392	0.5181	0.7142	0.6060
SVM (Word2Vec)	0.4988	0.5000	0.9738	0.6607
SVM (TF-IDF)	0.5012	0.5029	0.6000	0.5472
RNN (TF-IDF)	0.5024	0.5023	**1.0000**	0.6687
RNN (Word2Vec)	0.5012	0.5023	**1.0000**	0.6687
BERT	**0.8140**	**0.8300**	0.8000	**0.7900**
T5 (Human class)	0.6300	0.6200	0.6900	0.6500
T5 (AI class)	0.5800	0.5900	0.5300	0.5600

in all metrics of classification. Model selection should, however, be done carefully, especially in sensitive fields such as education and publishing, where false positives have serious ethical and reputation implications.

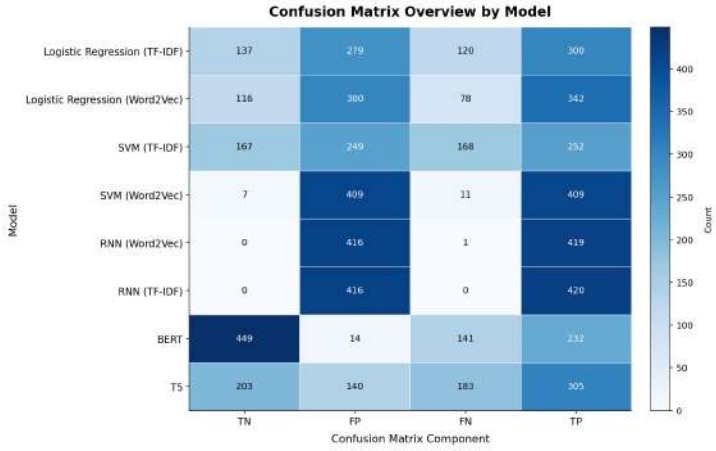

Figure 5. Confusion matrix analysis of all models.

6.3 Accuracy Across Train–Test Splits

In this section, we discuss the performance of the selected models under 2 different variations of training and testing data ratios. We conducted experiments with 70:30 and 80:20 training and testing split ratios. The results exhibit that the 70:30 split accuracy is a bit lower, or somewhat like an 80:20 split, which is understandable since a larger test set is hard to generalize. Performance comparison of all selected models under 70:30 and 80:20 training and testing split ratios is shown in Figure 6.

BERT was the most accurate in both splits, achieving over 81% accuracy, indicating its strong ability to perform the specified classification task. The accuracy of the Logistic Regression and SVM models was relatively similar, and differences among splits were minimal. However, larger fluctuations were recorded in RNN models with Word2Vec. It is concluded that bigger training sets are necessary to achieve better performance. Moreover, the tendencies also prove that model architecture is a more decisive factor in the accuracy results.

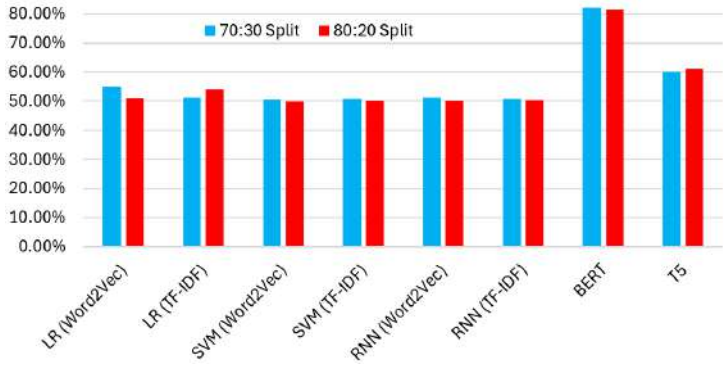

Figure 6. Performance comparison of all selected models under 70:30 and 80:20 training and testing split ratios.

6.4 Comparative Insights

Figure 7 exhibits the performance of the selected models in terms of precision, recall, f1-score, and accuracy. The performance of the BERT model was superior compared to the rest of the models. The accuracy of BERT was substantially higher than traditional and sequential baselines, confirming the weaknesses of older architec-

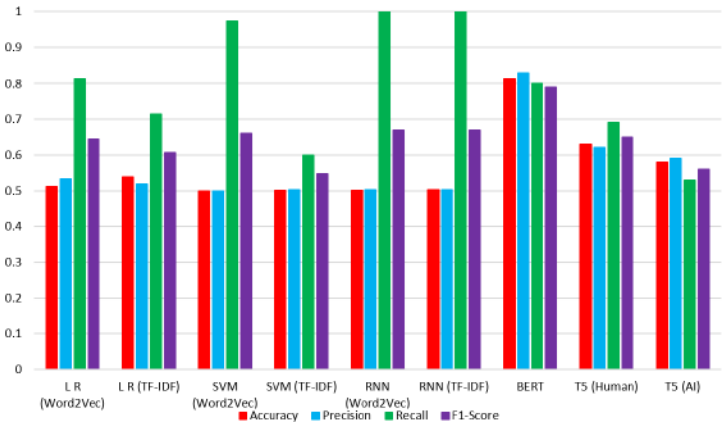

Figure 7. Performance comparison of machine learning and deep learning models.

tures faced with semantically similar but syntactically different academic text.

Overall, linear and sequential models served as useful points of baseline comparison and to show the relevance of feature representations, but compared to transformer-based models, especially BERT, they proved to be much more effective. Their capability to differentiate between artificial and AI-rephrased academic text was outstanding. However, even in the educational field, it might be necessary to balance interpretability (as in LR and SVM), computational limits (as in RNN and T5), and robustness (as in BERT).

6.5 Discussion

The findings of the current research highlight the presence of obvious differences in the work of classical machine learning models, deep neural architectures, and transformer-based systems. These variations are specific, especially in the identification of academic text that has been rewritten using AI. Such variations not only exist in numbers but also represent underlying differences in the way the models process languages, feature textual characteristics, and extrapolate to stylistic changeovers. When placed in a larger research context. These results contribute to the current computational linguistics and natural language processing discussions about the sufficiency of the traditional text classifier and the versatility of the contextualized language models.

6.5.1 Classical Models: Logistic Regression and SVM

Traditional algorithms like the Logistic Regression (LR) and Support Vector Machines (SVM) with TF-IDF features scored relatively low. LR achieved 53.92% accuracy while SVM achieved 50.12% accuracy. The result with Word2Vec embeddings was even worse. LR accuracy decreased to 51.17%, and SVM to 49.88%.

These are the results that imply that although TF-IDF still has some discriminating power in stylometric analysis, due to its sparseness. Frequency-based representation is soon surpassed by the even more subtle manipulations generated by the modern generative AI. This finding is in line with Solaiman et al. [4], who found that the frequency-based stylometric methods are only useful in short or shallow anomaly detection. This finding is also in line with Wahle et al. [3], who stated that linear classifiers do not work when semantic coherence is maintained, like in the case of the AI paraphrasing.

Word2Vec embeddings are semantically rich but seem to blur style boundaries by condensing token-level distinctions into high-density vector space. Consequently, they are prone to losing important cues that can be used by classifiers constructed on them to indicate any subtle differences between human and AI-generated text. This finding also supports the results of Jawahar et al. [31], who showed that conventional classifiers are not designed to understand complex structures and contexts and thus are not fit for authorship detection.

6.5.2 Sequential Model: RNN

The RNNs have shown little but significant progress compared to the selected linear models. The RNN trained on Word2Vec embeddings had a 50.12% accuracy on perfect recall (100%) with AI-generated content. In the same case, the RNN with TF-IDF features got an accuracy of 50.24% with a recall rate of 100%. The models, in both cases, however, misclassified almost all human-written texts, and thus achieved 0% specificity.

This behavior implies increased sensitivity to AI-generated stylistic cues at the expense of generalization to human writing. These results are consistent with Aldeen et al. [32], who reported the challenge of RNNs in long-range dependencies. These results are also aligned

with a study conducted by Belinkov and Glass [2], which noted that RNN-based models do not have the ability to generalize between semantically similar but stylistically different inputs.

6.5.3 Transformer Models: BERT and T5

Transformer-based architecture, especially BERT, delivered substantially better results. BERT attained 81.4% accuracy, 83% precision, and 80% recall, which is much higher than other models. Its success shows that pre-trained models that utilize self-attention mechanisms to encode local and global context can produce good results.

The study by Uchendu et al. [21] also indicated the effectiveness of BERT in distinguishing between AI-written and human-written news articles. Moreover, by using deep bidirectional embeddings, BERT can identify even subtle semantic anomalies and structural anomalies that are characteristic of text elements generated by AI, which shallow and sequential models can often fail to notice.

The confusion matrix also proves the credibility of BERT. The number of false positives and false negatives for BERT was 14 and 141, respectively, which means that the model was sensitive enough to the text generated by AI and was strong enough to avoid classifying the text generated by humans. This ratio is especially important in educational institutions that should not unfairly punish human labor.

Another transformer-based model, T5, shows a more nuanced performance profile. It had a total accuracy of 63% with 69% recall for human-generated texts and 53% recall for AI-generated texts. The pattern of T5 was different than BERT. The number of false positive cases in T5 is 183, which shows that it often misclassified human generated text and AI generated. This behavior shows bias towards predicting the AI class. These results point to the difficulty of optimizing fine-tuning of generative encoder-decoder models such as T5, which are more focused on text generation than on correct discrimination between classes. The same fact has been reported by Krishna et al. [9] who state that if we frame the classification problem as a generative problem, it will affect the decision boundary and reduce the accuracy.

However, the ability of the T5 model to capture subtle patterns in human-generated text makes it a potential candidate for hybrid detection systems. For example, T5 can help BERT in capturing a broader range of stylistic patterns in a hybrid system. This combination can more effectively perform the classification tasks.

6.5.4 Implications

In conclusion, the results show that classical models are computationally efficient but were found ineffective to detect text generated by modern generative linguistic tools. Deep neural models like the RNNs provide slight improvements in recall but are far-reaching due to the model structure. Transformer-based models, especially BERT, however, provide far better results due to their contextual modeling along with syntax and semantics features. Therefore, the BERT model is recommended for the detection of human and AI-generated text in academia. However, further research may be conducted on hybrid approaches for more balanced structures and to get better precision, recall, and accuracy.

6.5.5 Ethical and Computational Considerations

Considering the findings of this study, it is also important to consider the ethical implications and computational trade-offs related to deploying machine learning models for AI authorship detection. Models with recall rates (SVM and RNN) are less likely to misclassify AI-generated text. However, they may have false positive cases, which can affect the fairness of the model, academic integrity, and user trust. On the other hand, models like BERT are more accurate, though they are so complex in terms of computational resources, with greater training costs, greater inference times, and might not be accessible in low-resource settings. Furthermore, since transformer models are pre-trained on large datasets, they might become overfitted during fine-tuning. Therefore, continuous model management and monitoring are very important. To ensure that the text classification system operates accurately and effectively, a balance must be maintained between computational efficiency and ethical considerations.

7 Conclusion and Future Work

In this paper, we presented a detailed analysis of deep learning and machine learning models for the identification of human-generated, AI-generated, and

AI-rewritten academic texts. The data set used in this research comprises postgraduate-level reports submitted by students at the University of Agriculture, Peshawar, and AI-rephrased versions of the same reports. Five different models were used for experiments, including LR, SVM, RNN, BERT, and T5.

The results show that conventional machine learning (LR and SVM) models were struggling to understand the complex changes made by modern AI-based generative linguistics tools. The implementation of dependency features does not provide the rich semantic and syntactic information required for high-accuracy classification.

Whereas the performance of the Recurrent Neural Network (RNN) model was slightly better compared to SVM and LR because RNN can model dependencies in text. However, it is also not recommended due to its limitation in capturing long-range syntactic relationships. Transformer-based models, particularly BERT, performed better than the rest of the selected models by a significant margin.

BERT model was found to be the most successful model with an accuracy of 81.4% highest among all models. Its bidirectional pre-training model helps to understand the context of the text and to effectively verify the authorship. The T5 model, on the other hand, did not produce good results compared to BERT; however, it was sensitive to human-generated text, making it a potential model for ensemble systems.

Overall, the results show the importance of advanced context-aware approaches for the detection of AI-generated text. Although this study pointed out some useful insights for the identification of AI-generated text, especially in academics, there are many gaps that future research can address.

1. **Cross-model generalization:** This research is conducted using ChatGPT-generated texts only. Future studies ought to test the external validity of the performance of different AI systems, including Claude, Gemini, LLaMA, and other open-source models such as Mistral and Falcon, in conditions of performance regarding various stylistic and architectural diversification.
2. **Hybrid and ensemble detection:** The hybrid approach of BERT (discriminative transformers) with stylistically sensitive models (T5) may produce good results. These hybrid frameworks may give the best performance by balancing the weaknesses of one model with the strengths of another model.
3. **Explainable AI integration:** Using explainability tools (such as SHAP, LIME) or heatmap attention would enhance the trust in these systems, particularly in academic and legal contexts where transparency is vital. The use of explainable AI (XAI) tools would ensure that the outcome of the detection is auditable and justified.
4. **Real-time and scalable deployment:** The real-time deployment of these tools, such as text editing applications, AI plagiarism detection software, and browser extensions for runtime testing, must be created to bridge the gap between theory and practice. Moreover, these tools should be tuned to manage a high volume of data without compromising quality.
5. **Ethical and institutional governance:** In addition to the above technical improvements, institutions should establish clear policies to balance the detection abilities with fairness and due process. The results of the current study show that transformer-based models (especially BERT) are found to be a prominent model to detect AI-generated text with higher accuracy. However, the AI-based text generation tools are evolving constantly; therefore, the text detection system should also be improved continuously. Future efforts should consider ethical awareness and technical advancements to ensure a trustworthy, fair, and effective system to defend academic integrity.

In summary, transformer-based models, especially BERT, are the most effective models in terms of accuracy to detect AI-generated text with higher accuracy compared to other selected models. However, AI-based text generative models evolve quickly; therefore, the text detection model must also be improved and updated on a regular basis. Moreover, hybrid models, when deployed under proper ethical and deployment strategy consideration, will help ensure effective authorship verification and user trust in the academic domain.

Author Contributions

All authors have equal contributions.

Compliance with Ethical Standards

This document has been prepared under the ethical guidelines of the University of Agriculture, Peshawar.

Code and Dataset availability

The code and dataset used in this study are available on: https://github.com/alirazabangash1/Sample-Dataset

Funding Information

No funding has been received for this research.

References

[1] E. Clark, T. August, S. Serrano, N. Haduong, S. Gururangan, and N. A. Smith, "All that's 'human' is not gold: Evaluating human evaluation of generated text," in *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (ACL-IJCNLP 2021)*, pp. 7282–7296, 2021. doi: 10.18653/v1/2021.acl-long.565.

[2] Y. Belinkov and J. Glass, "Analysis Methods in Neural Language Processing: A Survey," *Transactions of the Association for Computational Linguistics*, vol. 7, pp. 49–72, 2019.

[3] H. Huang, N. Sun, M. Tani, Y. Zhang, J. Jiang, and S. Jha, "Can LLM-generated misinformation be detected: A study on Cyber Threat Intelligence," *Future Generation Computer Systems*, vol. 173, p. 107877, 2025. doi: 10.1016/j.future.2025.107877.

[4] S. Gehrmann, H. Strobelt, and A. Rush, "GLTR: Statistical detection and visualization of generated text," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, Florence, Italy, Jul. 2019, pp. 111–116. doi: 10.18653/v1/P19-3019.

[5] F. Alqasemi, M. F. Aldafer, N. F. Assarwie, and Y. K. Ahmed, "A comparative study for Yemeni poets detection using TEXT-CNN and RNN-LSTM text classification," in *Proceedings of the 2025 5th International Conference on Emerging Smart Technologies and Applications (eSmarTA)*, pp. 1–8, 2025. doi: 10.1109/eSmarTA.2025.XXXXXXX.

[6] E. Al-Buraihy, D. Wang, R. Khan, and M. Ullah, "An ML-based classification scheme for analyzing the social network reviews of Yemeni people," *International Arab Journal of Information Technology*, vol. 19, no. 6, pp. 904–914, 2022. doi: 10.34028/iajit/19/6/8.

[7] Z. Iqbal, S. Murtaza, H. Y. Chan, M. R. Ghori, N. Ahmed, and H. Ayub, "Handling Illusive Text in Document to Improve Accuracy of Plagiarism Detection Algorithm," *Proceedings of the 11th International Conference on Robotics, Vision, Signal Processing and Power Applications*, Lecture Notes in Electrical Engineering, vol. 829, pp. 93–100, Springer, 2022.

[8] T. Kehkashan, R. A. Riaz, A. S. Al-Shamayleh, A. Akhunzada, N. Ali, M. Hamza, and F. Akbar, "AI-generated text detection: A comprehensive review of methods, datasets, and applications," *Computer Science Review*, vol. 58, p. 100793, 2025. doi: 10.1016/j.cosrev.2025.100793.

[9] K. Krishna, Y. Song, M. Karpinska, J. Wieting, and M. Iyyer, "Paraphrasing evades detectors of AI-generated text, but retrieval is an effective defense," *Advances in Neural Information Processing Systems*, vol. 36, pp. 27469–27500, 2023.

[10] N. Selwyn, "The future of AI and education: Some cautionary notes," *European Journal of Education*, vol. 57, no. 4, pp. 622–634, 2022.

[11] I. Solaiman, M. Brundage, J. Clark, A. Askell, A. Herbert-Voss, J. Wu, and J. Wang, "Release strategies and the social impacts of language models," *arXiv preprint arXiv:1908.09203*, 2019.

[12] H. Stiff and F. Johansson, "Detecting computer-generated disinformation," *International Journal of Data Science and Analytics*, vol. 13, no. 4, pp. 363–383, 2022.

[13] A. de Santana Correia and E. L. Colombini, "Attention, please! A survey of neural attention models in deep learning," *Artificial Intelligence Review*, vol. 55, pp. 6037–6124, 2022.

[14] J. P. Wahle, T. Ruas, F. Kirstein, and B. Gipp, "How large language models are transforming machine-paraphrase plagiarism," in *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Abu Dhabi, United Arab Emirates, pp. 952–963, Dec. 2022. doi: 10.18653/v1/2022.emnlp-main.62.

[15] R. Zellers, A. Holtzman, H. Rashkin, Y. Bisk, A. Farhadi, F. Roesner, and Y. Choi, "Defending against neural fake news," *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[16] A. M. Salih, Z. Raisi-Estabragh, I. B. Galazzo, P. Radeva, S. E. Petersen, K. Lekadir, and G. Menegaz, "A perspective on explainable artificial intelligence methods: SHAP and LIME," *Advanced Intelligent Systems*, vol. 7, no. 1, p. 2400304, 2025.

[17] W. Antoun, V. Mouilleron, B. Sagot, and D. Seddah, "Towards a Robust Detection of Language Model-Generated Text: Is ChatGPT that Easy to Detect?," *Actes de la 30e Conférence sur le Traitement Automatique des Langues Naturelles (TALN)*, p. 14, 2023.

[18] I. Katib, F. Y. Assiri, H. A. Abdushkour, D. Hamed, and M. Ragab, "Differentiating Chat Generative Pretrained Transformer from Humans: Detecting ChatGPT-generated text and human text using machine learning," *Mathematics*, vol. 11, no. 15, p. 3400, 2023.

[19] Z. Su, X. Wu, W. Zhou, G. Ma, and S. Hu, "HC3 Plus: A semantic-invariant human–ChatGPT comparison corpus," *arXiv preprint arXiv:2309.02731*, 2023. doi: 10.48550/arXiv.2309.02731.

[20] C. Vasilatos, M. Alam, T. Rahwan, Y. Zaki, and M. Maniatakos, "HowkGPT: Investigating the Detection of ChatGPT-generated University Student Homework through Context-Aware Perplexity Analysis," *arXiv preprint arXiv:2305.18226*, 2023.

[21] A. Uchendu, Z. Ma, T. Le, R. Zhang, and D. Lee, "TURINGBENCH: A benchmark environment for Turing test in the age of neural text generation," in *Findings of the Association for Computational Linguistics: EMNLP 2021*, Punta Cana, Dominican Republic, pp. 2001–2016, Nov. 2021. doi: 10.18653/v1/2021.findings-emnlp.172.

[22] H. P. Nguyen *et al.*, "Logistic regression on guard of students' academic performance," in *Artificial Intelligence and System Engineering (CoMeSySo 2024)*, R. Silhavy and P. Silhavy, Eds., Lecture Notes in Networks and Systems, vol. 1490. Cham, Switzerland: Springer, 2025. doi: 10.1007/978-3-031-96759-7_26.

[23] F. Zhao and F. Yu, "Enhancing multi-class news classification through BERT-augmented prompt engineering in large language models: A novel approach," in *Proc. 10th Int. Sci. and Practical Conf. "Problems and Prospects of Modern Science and Education"*, Stockholm, Sweden, Mar. 12–15, 2024, pp. 297. International Science Group, 2024.

[24] S. Gul, R. U. Khan, M. Ullah, R. Aftab, A. Waheed, and T.-Y. Wu, "Tanz-Indicator: A novel framework for detection of Perso-Arabic-scripted Urdu sarcastic opinions," *Wireless Communications and Mobile Computing*, vol. 2022, no. 1, p. 9151890, 2022. doi: 10.1155/2022/9151890.

[25] R. Khan, M. Ullah, and B. Shafi, "Web search privacy evaluation metrics," in *Protecting User Privacy in Web Search Utilization*, IGI Global, 2023, pp. 46–62.

[26] N. Fatima, S. Riaz, S. Ali, R. Khan, M. Ullah, and D. Kwak, "Sensors faults classification and faulty signals reconstruction using deep learning," *IEEE Access*, vol. 12, pp. 1–10, 2024. doi: 10.1109/ACCESS.2024.3425408.

[27] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," *Proceedings of NAACL-HLT*, pp. 4171–4186, 2019.

[28] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu, "Exploring the limits of transfer learning with a unified text-to-text transformer," *Journal of Machine Learning Research*, vol. 21, no. 140, pp. 1–67, 2020.

[29] G. Canbek, T. Taskaya Temizel, and S. Sagiroglu, "PToPI: A comprehensive review, analysis, and knowledge representation of binary classification performance measures/-metrics," *SN Computer Science*, vol. 4, no. 1, pp. 13, 2022.

[30] A. R. Bangash, "Sample Dataset: Human vs AI Authored Academic Writing," *GitHub Repository*, 2025. [Online]. Available: https://github.com/alirazabangash1/Sample-Dataset.

[31] G. Jawahar, B. Sagot, and D. Seddah, "What does BERT learn about the structure of language?," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL)*, Florence, Italy, pp. 3651–3657, Jul. 2019. doi: 10.18653/v1/P19-1356.

[32] S. D. Aldeen, T. Abbas, and A. R. Abbas, "Review of detecting text generated by ChatGPT using machine and deep-learning models: A tools and methods analysis," *Diyala Journal of Engineering Sciences*, pp. 34–54, 2025.