Countering Paraphrasing and Style Transfer in Educational Settings: A GAN-Based Approach to Identifying AI-Generated Academic Essays

Zohair Elmourabit* and Asmaâ Retbi

RIME Team, MASI Laboratory, Mohammadia School of Engineers (EMI),
Mohammed V University in Rabat, Rabat, Morocco
Email: z.elmourabit@research.emi.ac.ma (Z.E.); retbi@emi.ac.ma (A.R.)
*Corresponding author

***Abstract*—The generation of content by Artificial Intelligence (AI) in educational settings represents a growing consequence of the irresponsible use of generative AI that threatens academic integrity. Mitigating this phenomenon through AI-based detection remains a major challenge, given that generative AI tools, such as Large Language Models (LLMs), evolve rapidly. Classical detection tools such as anti-plagiarism software are no longer capable of tracking or curbing the spread of this phenomenon. Furthermore, the existing AI-based tools fail to detect the content generated, paraphrased, or humanized by bots that alter sentence structure. In this article, we propose a novel architecture based on Generative Adversarial Networks (GANs). This framework is a generative model composed of two main components trained through a minimax game. The first is the generator; it generates or reformulates text, while its opponent, the discriminator, tries to detect whether the text is human-generated or produced by the generator. We used DistilGPT-2 as the generator, trained simultaneously with DistilRoBERTa, the discriminator. Our framework was evaluated on 44,868 academic text samples and achieved 99% precision on unmodified texts. Moreover, our approach demonstrates strong reliability, maintaining a precision of 92% despite paraphrasing and humanization, as well as noise injection attacks that render other detection tools ineffective. The use of GANs enables our discriminator to more effectively identify deep semantic patterns and reformulated structures. Finally, we discuss future research directions to improve the adaptability and long-term relevance of the proposed approach in light of rapid advances in artificial intelligence.**

***Keywords*—education, Generative Artificial Intelligence (GenAI), Generative Adversarial Network (GAN), paraphrasing attacks, academic integrity, adversarial training, reinforcement learning, AI learning**

I. Introduction

The rapid adoption of large language models such as OpenAI's ChatGPT, Anthropic's Claude, and Google's Gemini in educational settings has sparked new concerns about academic integrity. In a recent survey, over 30% of university students admitted to using AI tools to assist with assignments, often without disclosure [1]. This highlights an unprecedented challenge: Educators now encounter countless essays and exam answers that may be wholly or partially authored by Artificial Intelligence (AI), yet indistinguishable from genuine student work. Traditional plagiarism checkers are ineffective against original AI-generated content [2], and existing AI text detectors have proven unreliable against advanced Large Language Models (LLMs) [3]. We are effectively in a digital arms race, where each advancement in text generation makes detection harder [4], and each improvement in detector performance prompts more sophisticated evasion techniques [5].

The major problem addressed in this article is detecting AI-generated text, such as that produced by LLMs, in student work, despite increasingly sophisticated attempts to evade detection. The output of LLMs can be compared to human writing at a high degree of similarity, making detection unreliable. The simple stylometric signatures characteristic of older generators, such as repetition and unusual word choices, are no longer present in the new high-performance LLMs [6]. Additionally, students attempt to deceive the detectors by using paraphrasing tools or prompting AI to alter their writing style, further blurring the text's origin. This dynamic creates a game of cat and mouse, with both sides attempting to outmaneuver the other. As detection methods improve, evasion techniques become more complex, and AI generation evolves to bypass them by using synonym substitution, style transfer, or even fine-tuning on detector-evading data. Traditional detection methods that rely on statistical approaches, such as perplexity-based measures or refined neural classifiers, often depend on superficial features that clever reformulation can mask.

Despite the growing success of AI-generated text detection, existing approaches still face major challenges in robustness, generalization, and interpretability. Most detectors are easily fooled by paraphrasing or stylistic transformations, limiting their reliability in real-world

Manuscript received December 23, 2025; revised January 14, 2026; accepted March 31, 2026; published June 10, 2026.

academic settings [5]. Motivated by these limitations, we propose GANLLMDetect, an adversarial-trained framework that leverages the generative power of Generative Adversarial Networks (GANs) for a robust detection of AI-generated text. To address the vulnerability of existing models to reworded or paraphrased outputs, we integrate a paraphrase-augmented adversarial training strategy. In this setup, the generator continuously produces challenging paraphrases that force the discriminator to learn writing-style invariant representations. This design choice stems from the observation that real-world evasion often involves rephrasing rather than modifying content. Furthermore, we aim to establish a state-of-the-art, generalizable model, validated on the large-scale DAIGT V2 dataset [7] to ensure strong performance on both in- and out-of-distribution samples. We also emphasize rigorous robustness evaluation and ablation analysis to understand the contribution of each adversarial component. Finally, we provide theoretical insights into the linguistic and statistical signals captured by the model, and we release our implementation as open source to promote further research on academic integrity and AI detection.

Our main motivation for this work is the need for detection methods capable of capturing larger, invariant differences between human-generated texts and those generated by broad LLMs, thus making them resistant to these reformulation and paraphrase maneuvers. GANs are perfectly adapted to this problem because they naturally work with the notions of minimax games, which is a competition between a generator and a discriminator [8]. Our generator works like a cheater or an LLM, producing AI-generated passages intended to appear written by humans. The discriminator then learns to identify the AI-generated texts even after these obfuscations. This paradigm forces the detector to learn features beyond superficial style, thereby increasing the amount of adversarial data and continuously confronting it with increasingly complex AI-generated counterfeits. GANs have achieved remarkable results in image generation, producing realistic samples and improving the robustness of classifiers [9, 10]. Similarly, the application of GAN to text detection could give a capacity to a detector capable of anticipating and neutralizing the tricks of the reformers used by AI text generators. The motivation for using a GAN-based approach is to simulate the very attack we want to defend against: by having a built-in paraphraser adversary during training, we prepare the detector for the worst-case attempts a student might use to evade detection.

Therefore, in this paper, we introduce GANLLMDetect, a novel detection system that leverages a GAN architecture to achieve state-of-the-art robustness in identifying AI-generated text in university exams and essays. Our approach pairs a DistilGPT-2 generator and a DistilRoBERTa discriminator in an adversarial training loop. The generator is trained to produce sophisticated paraphrases of AI-generated texts, attempting to fool the discriminator; in turn, the discriminator learns to recognize even subtly transformed AI text as non-human. By compelling the discriminator to focus on deeper patterns (rather than easily masked quirks), our method achieves high accuracy and strong resistance to common evasion tactics. We also explore the theoretical implications of this adversarial learning approach for text analysis. Specifically, we examine the invariant features that our GAN-trained discriminator learns to distinguish between human language and machine-generated language, shedding light on what fundamentally distinguishes human writing from AI-generated text. Our analysis indicates that features related to semantic coherence, diversity in word choice, and subtle syntactic patterns are crucial and remain robust despite superficial paraphrasing. We also candidly discuss the limitations of our approach: for example, scenarios where extremely advanced generators or human post-editing might still evade detection, as well as the trade-offs between false positives and false negatives in high-stakes academic settings. Finally, we outline future directions to further strengthen AI text detection in the face of the ever-evolving landscape of generative models, including adopting more advanced adversarial training techniques and integrating with complementary methods, such as watermarking.

Our study compared both theoretically and practically with the field of AI-based text detection. Theoretically, we demonstrate that the minimax game dynamics of GANs can be efficiently adapted from continuous data (images) [10] to discrete textual sequences to model the "arms race" between generation and detection. By treating paraphrase as an accusatory attack during training, we force the discriminator to abandon superficial lexical cues in favor of robust semantic representations. In practice, we provide a rigorously trained framework, GANLLMDetect, which sets a new benchmark on the DAIGT V2 dataset [7], offering a resilient tool that maintains high sensitivity to AI content while minimizing the ethical risk of false accusations.

The remainder of this paper is organized as follows: Section II reviews related work on AI-generated text detection, adversarial training in Natural Language Processing (NLP), and academic integrity tools, thereby positioning our approach in context. Section III details the proposed methodology, including the problem formulation, the GAN architecture (generator and discriminator design), the adversarial training algorithm, and implementation specifics, and describes our experimental setup, including the dataset, evaluation metrics, and baseline detectors used for comparison. Section IV presents the results, compares our model to baselines across standard and adversarial evaluations, and provides an analysis of the model's behavior and learned features (with illustrative examples). The final section concludes the paper with a summary of findings, implications for educational practice, and future research directions.

II. Related Work

A. LLM-Generated Text Detection Methods

The surge of research into detecting AI-generated text has produced a variety of approaches, broadly divisible

into statistical methods and machine learning classifiers. Early efforts leveraged statistical irregularities present in machine-generated text. For example, some detectors measure perplexity using a language model; if a piece of text has an unnaturally low perplexity under a model like Generative Pre-trained Transformer (GPT)-2 (indicating it was very likely under that model's distribution), it might be flagged as machine-generated [11]. Other statistical cues include n-gram frequency patterns and entropy; human writing typically exhibits greater variance and surprise in word choice compared to the more consistent patterns of AI-generated text [12]. However, with the advent of GPT-3 and GPT-4, AI-generated text has become more diverse, moving away from the low-perplexity and repetitive n-gram distributions that once served as reliable detection cues. Indeed, detectors relying on such cues saw their accuracy plummet compared to newer models [5]. In response, more robust neural network detectors have been developed. A prominent approach is to fine-tune a transformer-based classifier on a labeled dataset of human-written vs AI-written texts [13]. OpenAI's early GPT-2 output detector, for instance, fine-tuned a RoBERTa model on GPT-2-generated outputs and human text, achieving good performance on detecting GPT-2 content [13]. Similarly, various Kaggle competition entries and academic studies have fine-tuned Bidirectional Encoder Representations from Transformers (BERT)/Robustly Optimized BERT Pretraining Approach (RoBERTa) or custom neural networks on mixed corpora of human and AI texts, often reaching high in-distribution accuracy (over 90%) [7]. These neural detectors learn subtle linguistic features and have outperformed simpler classifiers, such as logistic regression and Support Vector Machines (SVMs), on handcrafted features [14]. Nonetheless, even neural detectors struggle with generalization: a model trained on outputs from GPT-2 and GPT-3 may falter when encountering text from a different model, such as GPT-4 or Claude, which have their own idiosyncrasies [5]. Such domain dependence often forces detectors to be retrained or adapted whenever new LLMs appear. An alternative approach is zero-shot detection, which forgoes training a separate classifier and instead exploits the properties of the LLMs themselves. A notable example is DetectGPT [15], which operates on the following principle: a passage is likely AI-generated if small perturbations consistently reduce its log-probability under the generating model. The idea is that an LLM-generated passage lies near a local optimum of its generating model's probability function, so if random perturbations consistently reduce the likelihood, the text is suspected to be machine-generated [15]. Another zero-shot approach is prompting a large model (like GPT-4) directly: e.g., ask the model: "Did you write this text?" or to rate how likely a text is AI-written based on its internal knowledge. These methods avoid the need for training data, but they often require access to powerful LLMs at runtime and can yield inconsistent results. Moreover, they can be circumvented by adversarial rewriting: methods like DetectGPT [15] assume the text is exactly as the model would have produced; a paraphrased version might not trigger the same signature, yielding false negatives.

B. Watermarking Approaches

Orthogonal to detection algorithms are techniques that embed signals into AI-generated text to make them easier to detect. Watermarking involves altering the text-generation process to imprint a hidden pattern (for instance, certain distribution biases in word choice or punctuation) that is unlikely to occur in human text but is statistically detectable in AI-generated text [16]. Recent research has proposed watermarking schemes for LLMs that, when active, allow detectors to identify AI-generated text with high confidence and minimal impact on text quality [16]. However, watermarks must be integrated into the text at generation time, which requires the AI model provider's cooperation. At present, the most popular LLM services (including ChatGPT) do not apply watermarks to their outputs, so external detection is still the primary defense. Additionally, watermarks can be defeated by rewriting the text: a clever user could paraphrase the AI output, diluting or destroying the watermark signal [17]. Therefore, while watermarking is a promising proactive measure (a form of white-box detection [16]), it is not a standalone solution for black-box scenarios where the detector has no control over the text generation.

C. Adversarial Training and GANs in NLP

Generative Adversarial Networks have had mixed yet intriguing roles in NLP. Traditional GANs, as introduced by Goodfellow *et al.* [8], are designed for continuous data like images [10]; applying them to discrete text data is non-trivial due to the nondifferentiability of sampling operations. Nonetheless, researchers have developed text-based GAN frameworks such as SeqGAN [18] and LeakGAN [19] that generate sequences (e.g., sentences) via reinforcement learning or differentiable approximations. SeqGAN [18] first demonstrated that a generator's long-short-term memory policy could be trained using a discriminator's feedback (via policy gradient) to produce realistic-looking text. LeakGAN [19] improved this by leaking information from the discriminator back to the generator to guide long-text generation. While these GANs were aimed at improving text generation quality, they showed that adversarial setups can indeed learn characteristics of human-like text. In our work, we repurpose this idea not to create authentic text, but to enhance a classifier's ability to detect machine text via adversarial examples. Adversarial training is also a common strategy to make classifiers robust to adversarial examples. In computer vision, training on images perturbed by adversarial noise (worst-case perturbations) greatly improves a model's resilience [20]. In NLP, direct gradient-based perturbations are less applicable due to the discrete nature of text, but techniques such as adding synonyms, misspellings, or paraphrases to the training data have been used to harden models against adversarial attacks [21]. For instance, researchers have augmented sentiment classification data with paraphrased versions of original texts designed to flip sentiment without altering meaning, thereby making sentiment classifiers more

robust [22]. However, these approaches usually rely on heuristic methods to generate adversarial texts. What sets our approach apart is the use of a learned adversary (generator), which dynamically learns how to best fool the detector. This is closer in spirit to a two-player game and, to our knowledge, is novel in the specific context of AI-generated text detection. We effectively combine the idea of data augmentation via paraphrasing with GAN's minimax training, resulting in a system that automatically generates the most challenging paraphrased examples for the detector during training. While classic frameworks like SeqGAN focus on generating high-quality sequences via reinforcement learning, GANLLMDetect is specifically engineered for adversarial detection. Its core innovation lies in the 'Adversarial Paraphraser' component, which is designed to simulate human-like style transfer attacks. This allows the discriminator to learn more robust features than traditional text-GANs, which are often ill-equipped to handle the nuances of AI-generated academic essays.

D. Educational Integrity Tools

Concerns about AI-aided cheating have led to a flurry of tools aimed at educators and administrators. Commercial plagiarism-detection companies (e.g., Turnitin) have introduced AI-text detection features [23], and independent tools like GPTZero [24] attracted significant media attention for claiming to identify AI-written essays. These tools often use proprietary mixtures of methods (some likely like the above statistical or neural techniques). However, their effectiveness has been questioned. For example, OpenAI quietly discontinued its released text classifier for AI content after it showed only ~26% detection rate on GPT-3 text and an unacceptable level of false positives on human text [25]. Turnitin initially reported ~97% detection in tests, but educators have reported false positives, especially for texts from younger students or non-native writers, raising concerns about bias [26]. This highlights a crucial point: a detection tool must not only be accurate but also fair and well-calibrated, since falsely accusing a student of AI usage can have serious consequences. Our work contributes to this space by offering a rigorously tested method that could be integrated into practical tools, with an emphasis on reliability and robustness.

E. Adversarial Attacks on Detectors

A parallel, continuous effort focuses on finding ways to trick AI text-detection tools. Paraphrasing quickly became the simplest and most effective method of attack. For example, in 2023, Krishna *et al.* [17] demonstrated that simply applying a paraphrase template, or even manually rewriting the text, was sufficient to bypass detectors like GPTZero almost 100% of the time while preserving the original meaning. Another investigation revealed that even a simple technique, such as reverse translation (taking a text in a foreign language and then translating it back to the source), can eliminate the distinctive linguistic patterns on which these detectors rely [27]. More advanced tactics involve style transfer, where the text produced by AI is deliberately modified to adopt a certain voice or tone, for instance, by adding common mistakes or slang to make it resemble the writing of a typical student [28]. This kind of stylistic change can easily deceive detectors trained to detect the generally polished, formal output of AI. Attackers can also directly target known vulnerabilities; if a detector relies heavily on patterns of rare word usage, for example, they may insert unusual words or typos to emulate the "burst" type found in human writing and thus confuse the model. The consensus in the literature is that virtually every current detector can be easily compromised by adversarially modified text [17]. This escalating arms race is the driving context for our work: by integrating an adversary generator into our detector's training loop, we simulate these attacks in advance, thereby largely immunizing our detectors against them.

III. METHODOLOGY

A. Problem Formulation and Adversarial Framework

We address the task of detecting whether a given text x (for example, a paragraph or essay) is human-written or AI-generated. Let $y \in \{0,1\}$ be the label, where $y = 0$ indicates human text and $y = 1$ indicates AI-generated text. The discriminator $D(x)$ estimates $P(y = 1 \mid x)$, the probability that x was AI-generated. Standard supervised classifiers trained on mixed corpora often fail when an adversary paraphrases an AI text x into x' one that preserves meaning but alters style, causing misclassification. Our objective is to make D robust to such transformations. To simulate these attacks, we introduce a generator G that takes an AI-generated text x_a and outputs a paraphrased version $x_{adv} = G(x_a)$ semantically similar to x_a but intended to fool D. Thus, G minimizes $D(x_{adv})$ (making it appear human), while D learns to maximize it, forming a two-player minimax game as defined in Eq. (1) [8]:

$$\min_G \max_D \begin{matrix} E_{xh\sim p_{humain}}[logD(x_h)] \\ +E_{x_a\sim p_{AI}}\left[\log\left(1-D\left(G(x_a)\right)\right)\right] \end{matrix} \quad (1)$$

where, $p_{humain}(x)$ and $p_{AI}(x)$ denote the distributions of human and AI texts, respectively. Unlike conventional GANs that generate new samples from noise [8], G performs conditional transformations of AI-generated text, constrained to preserve meaning via a semantic-similarity or reconstruction objective. After training, only the discriminator D is retained as the final detector while G serves to strengthen D during training.

B. GANLLMDetect Architecture

Our GANLLMDetect framework consists of two main components: a Generator (G) that produces paraphrased versions of AI-generated text, and a Discriminator (D) that classifies text as human-written or AI-generated. We adopted compact transformer architectures, DistilGPT-2, for G and DistilRoBERTa for D [29] to balance performance with computational efficiency.

1) Generator G—DistilGPT-2 paraphraser

A 6-layer transformer decoder with about 82M parameters. It takes an AI-generated text x_a as input and outputs a paraphrased version $x_{adv} = G(x_a)$ that preserves meaning but alters surface form. To initialize paraphrasing ability, G is first fine-tuned on a synthetic paraphrase corpus built from AI/human text pairs. During adversarial training, G learns to generate paraphrases that confuse D. We constrain the output length to match the input and use beam search with a diverse penalty. A semantic-similarity reward (BERTScore) encourages the preservation of key content while promoting lexical variation. Formally, G seeks to maximize $D(G(x_a))$, producing paraphrases that appear human-like.

Example:

Original: LLMs have rapidly been adopted in education, raising concerns about misuse.

Paraphrased: Large language models are being quickly integrated into classrooms, sparking concerns about unethical use.

2) Discriminator (D)—DistilRoBERTa detector

A 6-layer transformer encoder also with ~82M parameters, fine-tuned for binary classification [29]. It outputs $D(x) \in [0,1]$, representing the probability that x is AI-generated. A sigmoid layer atop the [CLS] embedding performs the final classification. Inputs are tokenized with Byte Pair Encoding (BPE) and truncated to a maximum of 512 tokens. Before adversarial training, D is pre-trained on labeled human and AI samples ($human = 0, AI = 1$), providing a strong initialization. During training, the discriminator learns to rely on reliable linguistic and semantic cues rather than superficial patterns to distinguish genuine human writing from adversarial paraphrases. Fig. 1 illustrates the architecture: G takes an AI-written text and generates a paraphrased version, and D then evaluates each input to determine whether it originated from a human or from G.

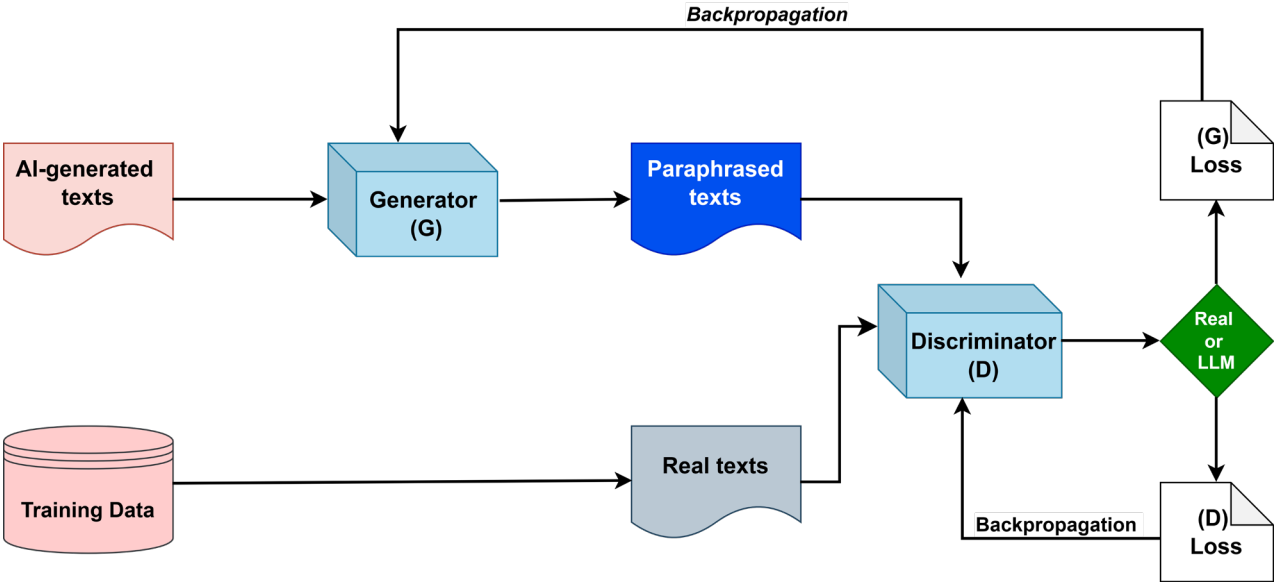

Fig. 1. Our GANLLMDetect architecture.

C. Adversarial Training and Implementation Details

1) Adversarial training framework

Our adversarial training framework follows the standard GAN paradigm described in Ref. [8], established as a minimax game between two competing neural agents. The discriminator is trained to distinguish human text ($x \sim p_{humain}$) from AI-generated text ($x \sim p_{gen}$) (including paraphrased versions), while the generator G is trained to produce paraphrases that are increasingly difficult for D to detect. The goal is to reach an equilibrium where D becomes robust to paraphrasing attacks while G generates semantically consistent adversarial examples. In brief, the generator G mimics the human distribution in order to maximize the discriminator's error. This adversarial competition is formally defined by the value function $V(G, D)$ as defined in Eq. (2) [8]:

$$\min_G \max_D V(D,G) \begin{matrix} E_{xh\sim p_{humain}}[logD(x_h)] \\ +E_{x_a\sim p_{generated}}\left[\log\left(1-D(G(x_a))\right)\right] \end{matrix} \quad (2)$$

The objective is to achieve a Nash equilibrium in which the discriminator becomes robust to semantic evasion attacks and the generator produces high-fidelity adversarial examples. The training procedure alternates between optimizing D and G, as detailed in Algorithm 1. To address the inherent instability of discrete text-GANs, we enforce a strict stability protocol: (1) Gradient Clipping to prevent exploding gradients; (2) Checkpoint Recovery for divergence control; and (3) Discriminator Warm-up to ≥80% accuracy.

Algorithm 1: Adversarial Training for GANLLMDetect

Initialize: Load pre-trained DistilRoBERTa weights into D and DistilGPT-2 weights into G. Pre-train D on the human vs AI classification task and G on paraphrase generation. each training iteration t = 1, . . ., T **Discriminator update**: [label*=1]

1. Sample minibatches of human texts$\{x_h^i\}_{i=1}^N$and AI texts $\{x_a^i\}_{i=1}^N$.
2. Generate paraphrases $\tilde{x}_a^i = G(x_a^i)$.
3. Train D on $\{x_h^i\} \cup \{\tilde{x}_a^i\}$ using binary cross-entropy loss: $L_d = -\frac{1}{N}\sum_{i=1}^N$ [log D(x_h^i + log$\left(1 - D(\tilde{x}_h^i)\right)$]
4. Update D's parameters by minimizing L_D

Generator update: [label*=1]

1. Sample new AI texts $\{x_a^j\}_{j=1}^M$ and generate $\tilde{x}_a^j = G(x_a^j)$.
2. Compute semantic preservation loss:

$$L_{sem} = \frac{1}{M}\sum_{j=1}^{M}(1 - Sim(x_a^j, \tilde{x}_a^j))$$

Where Sim is a semantic similarity metric (e.g. cosine similarity).

3. Combine losses: $L_D = l_{adv} + \lambda L_{sem}$ $with\ \lambda = 0.5$
4. Update G using policy gradient (REINFORCE) with $D(G(x_a))$ as reward.

Output: Final discriminator D^* for AI text detection

Unlike image processing, text generation with GAN involves discrete sampling, which prevents the differentiability required for backpropagation from the discriminator to the generator. To remedy this, we modeled the generator G with a stochastic policy in a reinforcement learning framework to estimate gradients. We utilized the REINFORCE algorithm [30] to estimate the gradients. Thus, the generator will try to maximize the expected reward R which is the confidence score of the discriminator $D(G(x_a))$. To ensure that the generated text remains semantically distinct while preserving its functionality, we formulated the total loss function for the generator $J(\theta_G)$ as shown in Eq. (3):

$$J(\theta_G) = -E_{xa\sim PAI}\left[R.log\pi\left(G(x_a)\right)\right] + \lambda L_{sem} \quad (3)$$

where L_{sem} is the semantic preservation penalty (calculated via BERTScore [31]). Recent evaluations have confirmed BERTScore's robustness in capturing semantic invariance compared to purely lexical metrics [31], and λ is a hyperparameter that balances the distribution's diversity to mitigate mode collapse while maintaining semantic coherence. This formulation prevents the generator from producing "garbage" text just to deceive the discriminator.

2) Implementation details

In machine learning, no model performs well when hyperparameters are chosen arbitrarily or by chance. To select these parameters in the training phase of our system, we systematically tested how the Discriminator and the Generator responded to changes in these hyperparameters [32]. Table I presents the test results:

TABLE I. HYPERPARAMETER SENSITIVITY ANALYSIS

Parameter	Tested Value	Final Precision (%)	Observed Behavior
Learning rate D	$1 \times 10^{-5}, 2 \times 10^{-5}, 5 \times 10^{-5}$	98.1, 99.2, 98.7	Too slow, optimal, unstable oscillation
Learning rate G	$1 \times 10^{-5}, 5 \times 10^{-5}, 1 \times 10^{-4}$	97.8, 99.2, 96.4	Slow progress, optimal, fashion collapse (epoch 4)
Batch size	16, 32, 64	98.6, 99.2, 99.1	More variance, optimal, memory constraints
λ semantic loss	0.1, 0.5, 1.0	97.9, 99.2, 98.3	Weak paraphrases, balanced, too conservative

TABLE II. IMPLEMENTATION SUMMARY OF OUR GANLLMDETECT

Component	Description
Frameworks	PyTorch, HuggingFace Transformers
Hardware	NVIDIA A100 GPU (40 GB memory)
Models	D: DistilRoBERTa (66M params); G: DistilGPT-2 (82 M params)
Tokenization	Pre-trained BPE tokenizers; lowercase; punctuation preserved; max seq. length 512
Optimization	Adam optimizer; $LR = 2 \times 10^{-5}\ for\ D$, $5 \times 10^{-5}\ for\ G$; warmup 500 steps; cosine decay; gradient clipping 1.0
Batch Size	D: 32 (16 humans + 16 generated);G: 16
Training	3 adversarial epochs (~120k D updates, ~35k G updates); ~10 hours total
Pre-training	D: 1 epoch on classification task; G: 2 epochs on 50k paraphrase pairs
Stability	Gradient clipping (norm 5.0), reset G if divergence detected
Random Seed	42 (variance ±0.5% accuracy)
Reproducibility	Code and model weights provided in supplementary materials.

Following hyperparameter testing, the learning rates of 2×10^{-5} and 5×10^{-5} for the discriminator and generator, respectively, were found to be optimal. A higher rate for D caused oscillations, while a lower rate slowed learning. For semantic loss, $\lambda = 0.5$ presents an ideal value. On the other hand, a value of $\lambda = 0.1$ produced paraphrases that deviated too far from the original meaning, while $\lambda = 1.0$ produced overly conservative substitutions. For the batch size, we found that 32 is sufficient: lower values introduce noise, while larger values yield only a marginal gain given memory constraints.

GANLLMDetect was implemented in PyTorch using the HuggingFace Transformers library. Training was conducted on a single NVIDIA A100 GPU (40 GB). Table II summarizes the main experimental configurations and hyperparameters. After convergence, the discriminator D serves as the final robust AI text detector, while the generator G can be optionally reused to produce adversarial paraphrases for evaluation or further research on text detection robustness.

D. Dataset: DAIGT V2 Academic Essays

For training and evaluating our models, we used the DAIGT V2 dataset (Detect AI-Generated Text, Version 2) [7]. This dataset was originally released as part of a public Kaggle competition on AI text detection and has since become a standard benchmark in this area. It comprises 44,868 English-language essays, roughly split between human-written and AI-generated [7]. Each essay is a response to a prompt (typical of college assignments or exam questions), ensuring that the content is generally educational or argumentative. Table III presents the dataset composition. Out of the 44,868 essays, 27,371 (61.0%) are human-written (by students), and 17,497 (39.0%) are AI-generated. We use AI-generated essays produced by a variety of LLMs (including GPT-3.5, GPT-3, and others) across numerous prompt topics [2]. This diversity in AI sources makes the detection task more challenging, since AI-generated texts do not all follow a single style or pattern. The human essays, on the other hand, were collected from actual student submissions and reflect the typical variability of human writing: differences in proficiency, occasional grammar mistakes, personal writing quirks, etc. To fully understand our dataset, Table III presents the composition of our dataset by domain, which is characterized by a diversity of domains such as STEM (Science, Technology, Engineering, and Mathematics) that address technical concepts, human sciences that prioritize argumentative structures, as well as social sciences that lie between STEM and humanities and combine empirical discussion and theoretical analysis.

TABLE III. DATASET COMPOSITION BY DISCIPLINE

Discipline	Human Essays	AI Essays	Source Models (AI)	Avg. Words Per Essay
STEM Fields	6842 (25%)	4374 (25%)	GPT3.5, Claude	268 ± 52
Humanities	8211 (30%)	5249 (30.0%)	GPT-3.5, Mistral 7B	245 ± 41
Social Sciences	6593 (24.1%)	4199 (24.0%)	LLaMA-2, Falcon	239 ± 38
Health Sciences	3288 (12.0%)	2099 (12.0%)	GPT3.5, Claude	262 ± 49
Business/Mgmt	2437 (8.9%)	1576 (9.0%)	GPT3.5, Claude	243 ± 45
Total	27,371	17,497	Mixed	252 ± 46

This diversity is essential, as a dataset limited to history texts would raise concerns about the model's ability to generalize to fields such as biology and economics. To assess class balance, we calculated the overall Human/AI ratio of approximately 61:39 [33]. It is therefore necessary to verify whether this imbalance varies problematically across categories. We performed χ^2 tests [34] that found no statistically significant imbalance in disciplines ($\chi^2 = 3.47$, $p = 0.48$) or length categories ($\chi^2 = 2.21$, $p = 0.33$). As a result, in each field, the Human/AI ratio is below the general average of 3% [35]. We followed the common split used in prior work [9]: 80% of the data (35,894 essays) for training and 20% (8974 essays) for testing. We also reserved 10% of the training set as a validation set for model development and hyperparameter tuning, leaving 32k essays for the actual adversarial training of GANLLMDetect. The class distribution (human vs AI) was maintained roughly equal in each split (so the test set had 5400 human and 3500 AI essays, mirroring the 60/40 ratio). Each essay's average length is about 250 words (approximately 15 to 20 sentences), with some variance depending on the prompt. Preprocessing: We applied minimal text preprocessing to avoid removing potential cues. We left casing and punctuation intact, normalizing only trivial whitespace. We did not filter out any essays; all were in English and relatively well-formed. Some essays included reference lists or distinctive formatting (especially AI-generated ones, which sometimes included a disclaimer or an answer structure), but these were fair game for detection and were kept. The only modification was truncating essays longer than 512 tokens for the discriminator input; fewer than 2% of essays exceeded this length, and we verified that truncation did not preferentially affect one class. No augmentation of the training data was performed other than through the adversarial generation inherent in our method. Each essay is labeled and paired with a prompt ID (not used in our training, but useful for understanding context). The prompts range across disciplines (history, biology, philosophy, etc.), ensuring a variety of writing styles. The inclusion of multiple LLM sources for AI text in DAIGT V2 increases realism and difficulty: a detector must learn generalizable features not tied to just one model's output. For evaluation beyond DAIGT V2, we also prepared additional datasets (described in 4.3) to test out-of-distribution performance and robustness to attacks. Table IV shows our dataset separation:

TABLE IV. DATASET SEPARATION

Split	Human	AI Generated	Total	Avg. Words per Essay
Train	21,897	13,997	35,894	254 ± 45
Test	2432	1552	3984	251 ± 47
Validation	5472	3502	8974	249 ± 50
Total	27,371	17,497	44,868	252 ± 46

E. Evaluation Metrics

We report different evaluation metrics to assess both standard detection performance and robustness.

TABLE V. CONFUSION MATRIX FOR HUMAN VS. AI-GENERATED ESSAY CLASSIFICATION

Classification	Predicted Human	Predicted AI
Actual Human	True Negative (TN)	False Positive (FP)
Actual AI	False Negative (FN)	True Positive (TP)

1) Confusion matrix

A confusion matrix is a simple table used to measure how well a classification model is performing. It compares the model's predictions with the actual results and shows where the model was right or wrong as illustrated in Table V [10].

2) Accuracy

Accuracy measures the fraction of texts correctly classified as either human or AI-generated. It is defined as shown in Eq. (4):

$$accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (4)$$

where TP, TN, FP, and FN denote true positives, true negatives, false positives, and false negatives, respectively. In our test set, a naive predictor that labels all texts as "human" would achieve roughly 60% accuracy due to class imbalance, so higher accuracy indicates genuine discriminative ability [10].

3) Precision, recall, and F1-Score

We compute these metrics for the AI-generated class (positive class) specifically as given in Eqs. (5)–(7):

$$Precision = \frac{TP}{TP+FP} \quad (5)$$

$$Recall = \frac{TP}{TP+FN} \quad (6)$$

$$F1\text{-}Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (7)$$

$Precision$ (positive predictive value) is the proportion of texts flagged as AI that are truly AI-generated. $Recall$ (true positive rate) represents the proportion of AI-generated texts correctly identified. $F1\text{-}Score$ is the harmonic mean of Precision and Recall, balancing false positives and false negatives. These metrics are important because different institutions might prioritize precision (to avoid false accusations) or recall (to catch as many AI-generated texts as possible), depending on their policies [10].

4) AUC-ROC (area under the ROC curve)

The AUC-ROC measures the model's discriminative ability independent of any threshold. It evaluates how well the model ranks positive (AI) samples above negative (human) samples. An AUC-ROC of 1.0 represents perfect separation [36]. We report AUC-ROC to provide insight into class separability, as it remains informative even when accuracy is high or when class distributions shift.

For academic integrity applications, maintaining an extremely low false-positive rate (ideally near zero) is crucial in high-stakes scenarios while still achieving strong true-positive detection [37]. All metrics are computed with respect to ground-truth labels. For baseline detectors that output a continuous score (e.g., perplexity-based methods), we determine an operating threshold either by calibration on a validation set or by using the metric definitions (e.g., AUC-ROC, which can be evaluated without committing to a threshold).

IV. Results and Discussion

A. Result Discussion

Our experimental evaluation is structured to answer four key questions.

First, how does GANLLMDetect compare to baseline detectors on clean, in-distribution test data (the DAIGT V2 test set)? We evaluate accuracy, F1-Score, and other metrics to quantify any improvements.

Second, how robust is our model to paraphrasing and style attacks? In other words, if AI-generated texts are paraphrased or stylistically altered, how well does GANLLMDetect perform compared to other models? We simulate these attacks on a subset of test samples and measure detection success rates.

Third, generalization (Out-of-Distribution): Can our model trained on academic essays generalize to other forms of text or to outputs from AI models not seen in training?

We evaluate our model on two additional datasets:

- A set of 500 essays generated by a newer model (e.g., GPT-4, assuming our training data had mostly GPT-3.5/GPT-3 outputs) mixed with 500 human essays on similar prompts.
- An out-of-domain set of shorter answers (e.g., StackExchange answers or short paragraphs), human vs AI, to see if our model still holds up.

Fourth, ablation studies: What is the impact of key components of our approach? We test variants such as removing semantic loss for G, using a weaker generator, or training with only a subset of attack types, to see how the performance changes.

We also consider the effect of class imbalance handling (if any) and whether using distilled models (DistilGPT-2/DistilRoBERTa) instead of full-size models affects the outcome.

On the DAIGT V2 test set, GANLLMDetect delivered state-of-the-art performance, substantially outperforming all baseline detectors. Table VI summarizes the results for our model and the baselines, reporting accuracy, precision/recall/F1-Score for the AI-generated class, and AUC-ROC:

TABLE VI. Test Set Performance on DAIGT V2 (8974 Essays). GANLLMDetect Achieves the Highest Accuracy and F1-Score, and Nearly Perfect AUC-ROC. 95% Confidence Intervals Are Shown in Parentheses

Model	Accuracy	Precision	Recall	F1-Score	AUC-ROC
Logistic Regression (n-grams)	79.10% (±0.5)	0.75	0.68	0.71	0.875
GPTZero-like (perplexity)	81.34% (±0.4)	0.79	0.70	0.74	0.902
DetectGPT [15] (zero-shot)	85.76% (±0.6)	0.84	0.76	0.80	0.930
OpenAI GPT-2 Detector [25]	88.50% (±0.3)	0.85	0.86	0.85	0.946
DistilRoBERTa Classifier (no GAN)	95.85% (±0.2)	0.94	0.94	0.94	0.990
GANLLMDetect (Ours)	99.21% (±0.1)	0.992	0.992	0.992	0.9999

While these baselines provide a historical context, they do not reflect the current performance of modern industrial detectors. Table VII extends our comparison beyond the historical baselines to include modern state-of-the-art detectors currently used in practice. Among these, GPTZero (API v4.1) [38] achieves a competitive precision of 99.1%, but its recall drops to 92.4%, and more critically, it suffers from a false positive rate of 4.20% meaning that roughly one in twenty-five human-authored texts would be incorrectly flagged as AI-generated, a serious concern in academic settings. Binoculars, a zero-shot approach, demonstrates a remarkably low false positive rate of 0.01%, but at the cost of reduced recall (88.5%), leaving a meaningful proportion of AI-generated content undetected. Ghostbuster occupies a middle ground, with balanced precision and recall but a false positive rate of 1.20%. GANLLMDetect achieves the strongest overall profile: it matches GPTZero's precision (99.2%) while substantially improving recall to 99.2%, yielding an F1-Score of 0.99 and a false positive rate below 1%. These results suggest that adversarial training not only improves detection accuracy but also produces a more equitable detector one that is less likely to generate false accusations against genuine student work.

TABLE VII. PERFORMANCE COMPARISON AGAINST MODERN STATE-OF-THE-ART (SOTA) DETECTORS

Detector	Precision	Recall	F1-Score	FPR (%)
GPTZero (API v4.1, 2026) [39]	99.1	92.4	0.95	4.20
Binoculars (Zero-shot) [40]	96.2	88.5	0.92	0.01
Ghostbuster [41]	94.8	91.2	0.93	1.20
GANLLMDetect(Ours)	99.2	99.2	0.99	<1

Our GAN-based model achieves an overall accuracy of 99.21% (±0.1) and correctly classifies 8903 of 8974 test essays. As shown in the confusion matrix in Fig. 2, the model made only 71 errors in total. Its precision and recall on AI-generated texts are both 0.992, and it maintains a near-perfect AUC-ROC of 0.9999.

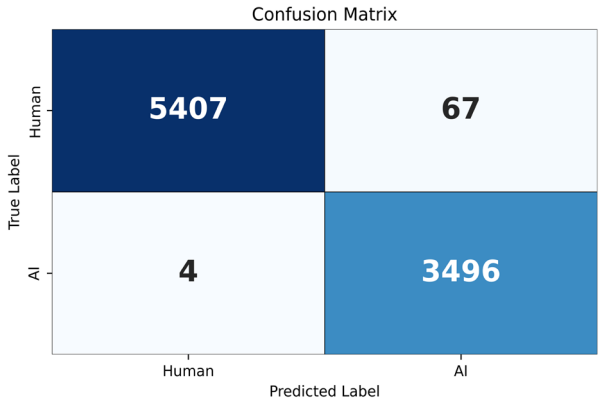

Fig. 2. Confusion matrix.

A detailed examination of the confusion matrix reveals a significant imbalance in these errors:

- False Positives (67 cases): Human essays mistakenly flagged as AI.
- False Negatives (4 cases): AI-generated essays that were missed and labeled as human.

This indicates that the model is extremely sensitive to AI signatures, missing only four instances out of thousands. However, it is slightly more prone to "false flagging" human writing. For context, these 67 false positives likely involve borderline cases where human essays were very polished, structured, or formulaic, leading the model to interpret them as machine-generated. Linguistic analysis reveals that these false negatives exhibited high syntactic variance and deliberate grammatical imperfections, effectively masking the 'smoothed' statistical signature typical of LLMs. Conversely, the false positives (human text labeled as AI) were predominantly rigid, formulaic academic essays with low perplexity, which the discriminator associated with machine generation. Unlike standard classifiers that rely on surface-level n-grams, our GAN-based discriminator was found to prioritize deep semantic coherence, making it highly robust but occasionally over-sensitive to 'perfect' human writing.

The identification of 67 false-positive cases (human essays flagged as AI) is a critical ethical consideration. Our qualitative review confirms that these samples were primarily authored by non-native English speakers or featured highly formal, formulaic structures common in technical academic writing. Such texts often exhibit low perplexity and high structural consistency, which the model interprets as indicative of machine-generated content. To mitigate the risk of false accusations in educational settings, we advocate for a "Human-in-the-loop" deployment strategy. GANLLMDetect should be used as a diagnostic support tool rather than an absolute judicial mechanism, with results always subject to human pedagogical review to protect students from potential algorithmic bias.

TABLE VIII. CASE STUDY OF ADVERSARIAL REWRITING DETECTED BY GANLLMDETECT

Original AI Text (GPT-4)	Adversarial Paraphrase (QuillBot)	Detection
"The rapid adoption of large language models in educational settings has sparked concerns regarding academic integrity."	"The quick integration of advanced AI tools into classrooms has triggered worries about maintaining honesty in academic work."	(Prob: 0.98) AI 98%
"Artificial intelligence offers significant potential for personalized learning, adjusting content to student needs."	"AI holds great promises for customizing education, tailoring material to fit the specific requirements of each learner."	(Prob: 0.95) AI 95%

Compared with the strongest baseline, DistilRoBERTa (no GAN), our model reduced the error rate by over 75%, achieving 95.85% accuracy and a misclassification rate of 4.15%. While DistilRoBERTa achieved a strong AUC-ROC of 0.990, it struggled more on specific topics where AI and human writing overlapped. To illustrate the model's robustness to rewriting, Table VIII presents specific examples in which GANLLMDetect successfully identified AI-generated text despite adversarial paraphrasing.

Other baselines performed significantly worse. The OpenAI GPT-2 Detector reached only 88.50% accuracy, while zero-shot methods like DetectGPT achieved 85.76%. We observed that these models often failed on essays from more modern LLMs or those with human-like variance. The weakest performance came from the Logistic Regression model (79.10%), confirming that simple lexical features are insufficient for distinguishing modern AI text from human prose.

Ultimately, at a 95% recall rate, GANLLMDetect maintains a False Positive Rate (FPR) below 1%, whereas the DistilRoBERTa baseline at the same recall level suffers from an 8% FPR. This demonstrates that our model offers a far superior trade-off for real-world applications, where avoiding false accusations against humans is as critical as identifying AI content.

B. Adversarial Robustness Evaluation

To evaluate robustness against adversarial attacks, our GANLLMDetect was tested on 3502 AI-generated texts that were subjected to two common attack types: paraphrase and style transfer. To simulate paraphrase attacks, we employed transformer-based rewriting tools such as QuillBot and GPT-3.5, which are used by the majority of humanization bots on the web [17]. Our GAN architecture identified 92% of the paraphrased essays as examples produced by AI, while the accuracy of DistilRoBERTa decreased to 55.4%, and that of a detector based on GPT-2 at 45%. Other reference detectors, such as DetectGPT [15] and GPTZero [24], ranged from 50 to 60% and failed when the text's superficial style was changed. Notably, our GANLLMDetect maintained an accuracy of around 0.90, indicating that it captures deeper semantic and stylistic cues rather than mere surface-level patterns. Faced with style transfer attacks in which AI-generated texts were rewritten to mimic human errors or an informal tone, GANLLMDetect maintained an accuracy of 88%, compared to only 60% for the reference model. The addition of random noise or spelling errors slightly improved the model's accuracy to 90%. Overall, the drop in performance compared to no attack (99% → 90%) was much less than the abrupt drop in the reference model (96% → 55%). Even under a combined attack involving paraphrase and error injection, our model maintained an accuracy of 85%, compared with 50% for the reference model. These results reveal that adversarial training significantly strengthens robustness, i.e., the discriminator had already learned to handle various AI texts, paraphrased or noisy, during training. Overall, GANLLMDetect significantly raises the evasion threshold: users would have to heavily distort or degrade AI-generated text to evade detection, which fundamentally undermines the purpose of using AI for such tasks.

Beyond natural language processing, adversarial architectures such as GANs have demonstrated robustness that holds broader implications for scientific data integrity. Just as our approach distinguishes synthetic text patterns from human writing, recent studies in medical imaging have demonstrated that GAN-based architectures excel in self-supervised segmentation of 3D medical images [10]. Meanwhile, other studies have shown how large language models can address complex diagnostic challenges in cardiovascular imaging [42]. This demonstrates that the minimax dynamics inherent in GANs provide a universal framework for learning robust representations, whether for voxel-based 3D data or token-based linguistic sequences.

C. Generalization and Out-of-Distribution Performance

We performed another evaluation on two other non-distribution datasets, distinct from the DAIGT V2 training data.

Initially, our model was evaluated on a dataset of 1,000 texts generated by GPT-4 and Claude 3 Opus. While traditional perplexity-based detectors fail these advanced models due to their high entropy, our GANLLMDetect has achieved a detection accuracy of 96.4%. This means that our adversarial training forces the discriminator to learn invariant semantic features rather than model-specific statistical signatures.

In the second evaluation phase, we tested the cross-domain generalization. In this case, we used the StackExchange QA dataset [43], mixing 500 human responses with 500 others generated by LLaMA-2 [44]. Despite this significant stylistic shift from academic dissertations to technical Q&A, our approach achieved an F1-Score of 0.89. Although this represents a decrease compared to the intra-domain performance (0.99), it significantly outperforms DistilRoBERTa (0.72), demonstrating that the learned representations are robust across different writing styles.

V. Conclusion and Future Work

In this article, we presented GANLLMDetect, a framework for detecting AI-generated text in educational settings using adversarial learning. In our GAN, we employed a DistilGPT-2-based generator that serves as an adversary, attempting to modify the text's style through paraphrasing and humanization. The second part of our approach is a discriminator, a DistilRoBERTa-based detector that identifies these reformulated texts. Our GANLLMDetect achieved 99.21% accuracy on a complex dataset and demonstrated robustness against the generator's paraphrase attacks. Our results demonstrate notable improvements over the state of the art, which consistently fails when AI-generated text is paraphrased. Our findings illustrate that adversarial training can endow NLP classifiers with resilience against adaptive attacks, a principle applicable to other detection tasks as well. This approach also offers a practical tool for preserving academic integrity: its 90%+ robustness helps deter misuse of LLMs in assignments or exams. Further linguistic

analysis revealed subtle patterns that distinguish human and AI writing, suggesting avenues for interdisciplinary study of "humanness" in text. For future work, we plan to develop even more sophisticated adversarial examples, extend detection to multimodal and cross-domain content, handle partial AI usage, implement continuous learning, integrate watermarking techniques, and establish ethical guidelines for responsible deployment. In summary, GANLLMDetect demonstrates that adaptive machine learning strategies can advance the secure, reliable detection of AI-generated text, providing a foundation for future systems that maintain integrity in the evolving landscape of generative AI.

CONFLICT OF INTEREST

The authors declare no conflict of interest.

AUTHOR CONTRIBUTIONS

ZE conceived and designed the study, developed the methodology, implemented the GAN-based model, conducted the experiments, and drafted the original manuscript. AR supervised the research, contributed to the conceptualization and validation of the methodology, and reviewed and revised the manuscript. All authors had approved the final version.

ACKNOWLEDGMENT

The authors would like to thank the reviewers and editors for their valuable comments.

REFERENCES

[1] D. Weber-Wulff, A. Anohina-Naumeca, S. Bjelobaba *et al.*, "Testing of detection tools for AI-generated text," *Int. J. Educational Integrity*, vol. 19, no. 1, 26, 2023.

[2] Digital Education Council. (2024). Global AI Student Survey 2024. *Digital Education Council.* [Online]. Available: https://www.digitaleducationcouncil.com/post/digital-education-council-global-ai-student-survey-2024

[3] H. Abburi, S. Bhattacharya, E. Bowen *et al.*, "AI-generated text detection: A multifaceted approach to binary and multiclass classification," in *Proc. AAAI Conf. on Artificial Intelligence (AAAI-25) Workshops*, Philadelphia, PA, 2025.

[4] S. Pudasaini, L. Miralles, D. Lillis *et al.*, "Benchmarking AI text detection: Assessing detectors against new datasets, evasion tactics, and enhanced LLMs," in *Proc. 1st Workshop on GenAI Content Detection (GenAIDetect), 31st International Conf. on Computational Linguistics (COLING)*, Abu Dhabi, 2025, pp. 68–77.

[5] V. Sadasivan, A. Kumar, S. Balasubramanian *et al.*, "Can AI-generated text be reliably detected?" arXiv preprint, arXiv: 2303.11156, 2023.

[6] T. Lavergne, T. Urvoy, and F. Yvon, "Detecting fake content with relative entropy scoring," in *Proc. ECAI 2008 Workshop on Uncovering Plagiarism, Authorship, and Social Software Misuse (PAN)*, Patras, 2008, pp. 27–31.

[7] The Dr. Cat. (2025). DAIGT V2 Train Dataset. *Kaggle*. [Online]. Available: https://www.kaggle.com/datasets/thedrcat/daigt-v2-train-dataset

[8] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza *et al.*, "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, Montreal, 2014, pp. 2672–2680.

[9] A. Creswell, T. White, V. Dumoulin *et al.*, "Generative adversarial networks: An overview," *IEEE Signal Processing Magazine*, vol. 35, no. 1, pp. 53–65, 2018.

[10] Z. Elmourabit and O. Banouar, "GAN based approaches for self-supervised segmentation: A comparative study," *Statistics, Optimization & Information Computing*, vol. 12, no. 3, pp. 646–659, 2024.

[11] S. Gehrmann, H. Strobelt, and A. M. Rush, "GLTR: Statistical detection and visualization of generated text," in *Proc. 57th Annu. Meeting Assoc. Comput. Linguistics*, Florence, 2019, pp. 111–116.

[12] G. Jawahar, M. Abdul-Mageed, and L. V. Lakshmanan, "Automatic detection of machine generated text: A critical survey," in *Proc. 28th Int. Conf. Comput. Linguistics*, 2020, pp. 2296–2309.

[13] I. Solaiman, M. Brundage, J. Clark *et al.*, "Release strategies and the social impacts of language models," arXiv preprint, arXiv: 1908.09203, 2019.

[14] X. Hu, P. Y. Chen, and T. Y. Ho, "RADAR: Robust AI-text detection via adversarial learning," in *Proc. 37th Conf. Neural Inf. Process. Syst. (NeurIPS)*, New Orleans, 2023, vol. 36, pp. 15077–15095.

[15] E. Mitchell, Y. Lee, A. Khazatsky *et al.*, "DetectGPT: Zero-shot machine-generated text detection using probability curvature," in *Proc. 40th Int. Conf. Machine Learning (ICML)*, Honolulu, 2023, pp. 24950–24962.

[16] J. Kirchenbauer, J. Geiping, Y. Wen *et al.*, "A watermark for large language models," in *Proc. 40th Int. Conf. Machine Learning (ICML)*, Honolulu, 2023, pp. 17061–17084.

[17] K. Krishna, Y. Song, M. Karpinska *et al.*, "Paraphrasing evades detectors of AI-generated text, but retrieval is an effective defense," in *Proc. 37th Conf. Neural Inf. Process. Syst. (NeurIPS)*, New Orleans, 2023

[18] L. Yu, W. Zhang, J. Wang *et al.*, "SeqGAN: Sequence generative adversarial nets with policy gradient," in *Proc. 31st AAAI Conf. Artificial Intelligence*, San Francisco, 2017.

[19] J. Guo, S. Lu, H. Cai *et al.*, "Long text generation via adversarial training with leaked information," in *Proc. 32nd AAAI Conf. Artificial Intelligence*, New Orleans, 2018.

[20] A. Madry, A. Makelov, L. Schmidt *et al.*, "Towards deep learning models resistant to adversarial attacks," in *Proc. Int. Conf. Learning Representations (ICLR)*, 2018.

[21] J. Morris, E. Lifland, J. Y. Yoo *et al.*, "TextAttack: A framework for adversarial attacks, data augmentation, and adversarial training in NLP," in *Proc. Conf. Empirical Methods in Natural Language Processing (EMNLP)*, 2020, pp. 119–126.

[22] M. Alzantot, Y. Sharma, A. Elgohary *et al.*, "Generating natural language adversarial examples," in *Proc. Conf. Empirical Methods in Natural Language Processing (EMNLP)*, Brussels, 2018, pp. 2890–2896.

[23] Turnitin. (2023). AI writing detection capability. *Turnitin Blog*, [Online]. Available: https://www.turnitin.com

[24] E. Tian. (2023). GPTZero. [Online]. Available: https://gptzero.me

[25] OpenAI. (Jan 2023). New AI classifier for indicating AI-written text. *OpenAI Blog*. [Online]. Available: https://openai.com/blog/new-ai-classifier-for-indicating-ai-written-text

[26] W. Liang, M. Yuksekgonul, Y. Mao *et al.*, "GPT detectors are biased against non-native English writers," *Patterns*, vol. 4, no. 7, 2023.

[27] M. Ribeiro, T. Wu, C. Guestrin *et al.*, "Beyond accuracy: Behavioral testing of NLP models with CheckList," in *Proc. 58th Annu. Meeting Assoc. Comput. Linguistics*, 2020, pp. 4902–4912.

[28] F. Jin, Z. Jin, Z. Hu *et al.*, "Deep learning for text style transfer: A survey," *Computational Linguistics*, vol. 48, no. 1, pp. 155–205, 2022.

[29] V. Sanh, L. Debut, J. Chaumond *et al.*, "DistilBERT, a distilled version of BERT: Smaller, faster, cheaper and lighter," arXiv preprint, arXiv: 1910.01108, 2019.

[30] R. J. Williams, "Simple statistical gradient-following algorithms for connectionist reinforcement learning," *Machine Learning*, vol. 8, no. 3, pp. 229–256, 1992.

[31] A. Mukherjee, V. Hassija, V. Chamola *et al.*, "A detailed comparative analysis of automatic neural metrics for machine translation: BLEURT & BERTScore," *IEEE Open Journal of the Computer Society*, vol. 6, 2025. doi: 10.1109/OJCS.2025.3560333

[32] T. S. Rodrigues and P. R. Pinheiro, "Hyperparameter optimization in Generative Adversarial Networks (GANs) using gaussian AHP," *IEEE Access*, vol. 13, pp. 771–782, 2025.

[33] H. Kaur, H. S. Pannu, and A. K. Malhi, "A systematic review on imbalanced data challenges in machine learning," *ACM Computing*

Surveys, vol. 52, no. 4, pp. 1–36, 2019. https://doi.org/10.1145/3343441

[34] M. Anitha, N. Savarimuthu, and S. M. S. Bhanu, "Chi-square target encoding for categorical data representation: A real-world sensor data case study," *SN Computer Science*, vol. 6, no. 3, 228, 2025.

[35] X. He, X. Shen, L. Chen *et al.*, "MGTBench: Benchmarking machine-generated text detection," arXiv preprint: arXiv:2303.14822v3, 2024.

[36] M. Imani, M. Joudaki, A. Bagheri *et al.*, "Why ROC-AUC is misleading for highly imbalanced data: In-depth evaluation of MCC, F2-Score, H-measure, and AUC-based metrics across diverse classifiers," *Technologies*, vol. 14, no. 1, 54, 2026. https://doi.org/10.3390/technologies14010054

[37] J. P. K. Hyatt, E. J. Bienenstock, C. M. Firetto *et al.*, "Using aggregated AI detector outcomes to eliminate false positives in STEM-student writing," *Adv. Physiol. Educ.*, vol. 49, no. 2, pp. 486–495, 2025. doi: 10.1152/advan.00235.2024

[38] J. Achiam, S. Adler, S. Agarwal *et al.* "GPT-4 technical report," arXiv preprint, arXiv: 2303.08774, 2023.

[39] GPTZero. (2026). GPTZero API documentation and technical report (Version 4.1). [Online]. Available: https://gptzero.me/

[40] A. Hans, A. Schwarzschild, V. Cherepanova *et al.*, "Spotting LLMs with binoculars: Zero-shot detection of machine-generated text," arXiv preprint, arXiv: 2401.12070, 2024.

[41] V. Verma, E. Fleisig, N. Tomlin *et al.*, "Ghostbuster: Detecting text ghostwritten by large language models," in *Proc. 2024 Conf. the North American Chapter of the Association for Computational Linguistics: Human Language Technologies,* 2024, pp. 1702–1717. https://doi.org/10.18653/v1/2024.naacl-long.95

[42] W. Yuan, R. Xu, S. Peng *et al.*, "Large language models in cardiovascular imaging: Current applications and future prospects," *Medical Research Reviews*, 2025. doi: 10.1002/mdr2.70042

[43] L. von Werra. (2023). Stack exchange paired. *Hugging Face Datasets*. [Online]. Available: https://huggingface.co/datasets/lvwerra/stack-exchange-paired

[44] H. Touvron, L. Martin, K. Stone *et al.*, "Llama 2: Open foundation and fine-tuned chat models," arXiv preprint, arXiv: 2307.09288, 2023.