Advanced International Journal for Research (AIJFR)
E-ISSN: 3048-7641 ● Website: www.aijfr.com ● Email: editor@aijfr.com

Detecting AI-Generated Text

Amala Teresa John [1], Sudha D [2]

[1] Student, Department of Computer Applications, SCMS School of Technology and Management, Kochi
[2] Assistant Professor, Department of Computer Applications, SCMS School of Technology and Management, Kochi

Abstract

Nowhere near perfect, today's AI text often reads just like something a person would write. Instead of relying on one method alone, researchers now test several ways to tell machine-made words apart. Some tools learn from labeled examples, others look at number patterns hidden in sentences. Curvature in probability spaces helps spot fakes without prior training data. Hidden signals baked into outputs during generation can leave traces behind. Short stretches of text get checked individually using fine-grained models. Systems are also built to work across different topics and styles. To measure progress fairly, a new testing structure evaluates how accurate, stable, fast, and flexible each system really is. Results show most do well in controlled tests but struggle when attackers try to fool them. Real gaps remain between lab results and messy reality. Stronger defenses may come from combining steady pattern recognition with resistance to manipulation attempts.

Keywords: AI-Generated Text Detection, Large Language Models, Adversarial Robustness, Domain Generalization, Natural Language Processing.

1. Introduction

One moment these models barely made sense, now they write like someone you might know. Because machines draft essays, articles, emails, even social posts, telling who - or what - wrote them feels harder every day. When fake writing slips through unnoticed, grades can be cheated, lies spread, voices mimicked without consent. Spotting the machine behind the words is not just useful - it quietly matters.

Most of the time, AI-written text is not copied word-for-word - so finding it requires different methods than usual plagiarism checks. Instead of spotting duplicates, researchers now look at subtle quirks hidden in how sentences are built - the rhythm, repetition, or odd choices only machines tend to make. Still, today's tools struggle when faced with new models they have not seen before or when someone tweaks output on purpose to avoid detection. What works well in theory often breaks down once used widely outside labs.

AI text detectors that shine in the lab do not always hold up once they leave it. Some manage to stay steady across different kinds of input, while others lose their footing the second something shifts. The real question is not how high a tool can score when everything lines up perfectly - it is whether it keeps working

when conditions get unpredictable. That kind of day-to-day dependability is what actually makes a detector worth using.

2. Literature Review

[1] Out of nowhere, Fraser and his group explain ways to identify machine-written text. Hidden clues link up in one approach; rhythm in words or sentences guides another. Systems learn contrasts by reviewing many samples over time. Results piece together a clearer image - what hides phony writing well, what gives it away fast. The design of the model plays a role. Length matters too. Edits by humans shift things. Even attempts to trick scanners reshape results. Together they tilt the odds. What comes through leans into what happens after, minus pushing solutions too hard. Key pieces show up, just not with loud promises. Quiet clarity carries it forward.

[2] Peering in, Abdali with colleagues explores how people detect computer-generated text. Still, hiccups pop up - large language tools sometimes push biased takes or wrong info. A section breaks down detection tactics, using smart algorithms alongside subtle clues buried in wording or silent markers tucked inside replies. Yet these methods tend to falter when sentences get twisted or learning data turns messy. All through, the study ponders whether tracking AI speech survives serious reasoning tests.

[3] Ahead of gut feeling, Boutadjine's group tried transfer learning on four big language systems, using a new set of Arabic data named Ara-Deep. Instead of relying on instinct, they found that fine-tuned models could distinguish machine-generated from human-written text with high accuracy - ranging between 94 and 99 percent. Yet an odd result emerged during testing. Machines showed sharp results, while humans fared poorly at the same task. The real surprise? It was less about how capable software had grown, more about how unreliable basic human sense appeared under pressure. This gap suggests one thing plainly: where people falter, automated help might soon fill the role.

[4] Oddly enough, Mohamed's group looked into telling apart human writing from ChatGPT output in scholarly texts - particularly within online learning platforms. Not only did they tweak advanced models like RoBERTa and T5, but also adjusted GPT-Neo-125M with custom-built science abstracts. Rather than using common datasets, training happened on actual samples taken straight from scientific studies. With findings rolling in, outcomes proved striking: accuracy soared above 99 percent. The key factor turned out to be meticulous fine-tuning, not just the size or type of model used.

[5] Out of many traits - like how confusing a text feels, its meaning patterns, mistakes it holds, or how easy it reads - one team dug into spotting machine-made and rewritten content in various tongues and topics. To pull this off, they built something called the Multilingual Human-AI-Generated Text Corpus. Instead of relying on today's top detectors such as GPTZero or ZeroGPT, their method leaned hard on smart trait selection - and beat those tools by a clear step.

[6] Looking closely at how well AI detectors spot machine-written text, Chaka tested tools against outputs from ChatGPT, YouChat, and Chatsonic. Using essay-like answers as samples, each detector showed wildly different outcomes - some flagged content others missed completely. One moment a response seemed human, the next it was labeled artificial, depending on which tool judged it. Even when the exact

same paragraph ran through several systems, verdicts clashed without pattern. False alarms popped up just as often as real AI passages slipped through unnoticed. This patchy behavior suggests today's software struggles to draw clear lines between human and synthetic writing. Because judgments shift so easily, trusting them in schools or research feels shaky at best. Better ways to tell the difference must come along before these tools can matter much in serious work.

[7] Chaka pulled together 17 past studies about how detectors work. Not one tool came out steady in its results - reliability was missing across the board. Even though Crossplag and Copyleaks did slightly less poorly, most couldn't tell real writing apart from machine-made words with any consistency. The pattern stood clear: these systems struggle when left to judge authorship alone.

[8] One step beyond mere testing, Sadasivan and team crafted a "recursive paraphrasing attack" to probe how well detection tools hold up under pressure. Not just stopping at practical tests, they showed this method weakens defenses across many types - including watermarking - without badly hurting readability. What stands out is their model framing detection success around how far apart human and machine-generated texts statistically drift. Instead of relying on intuition, they grounded results in measurable divergence between two kinds of written patterns.

[9] Even if fake text gets better at sounding human, spotting it still stays possible, say Chakraborty and team. Their math shows longer texts give clearer clues - more words mean more chances to catch subtle patterns machines miss. Though harder to detect when quality improves, enough material tips the balance back toward identification. Size matters here - not perfection.

[10] Wang et al. tackled the problem of sentence-level AIGT detection, which is very important for finding documents that have both human and machine authors. The paper presents SeqXGPT, an innovative approach that utilizes log probability lists from white-box LLMs as features for a classifier founded on convolutional and self-attention networks.

[11] Filling a gap that most detectors leave wide open, Bhattacharjee and team built EAGLE - a domain generalization framework designed to keep up with models it has never encountered before. Rather than retraining from scratch each time a new system appears, labeled data from older models is used to pull out features that travel well across domains. When put to the test against cutting-edge systems like GPT-4 and Claude, the framework held its ground, spotting generated text it had no prior exposure to.

[12] Weber-Wulff et al. performed one of the most thorough evaluations of detection tools, testing 14 different systems. The research found that the tools that are currently available are not accurate or reliable, and they tend to classify text as human-written. Also, it was found that obfuscation methods like machine translation and paraphrasing made performance much worse.

[13] Kumar and Mindzak looked into how well people can tell the difference between texts written by people and texts written by AI. Their study found that people were pretty good at telling the difference between texts written by humans (63% accuracy) but much worse than chance at telling the difference between texts written by AI (only 24% accuracy). This shows that people tend to think that texts are written by humans.

[14] Six commercial AI text detection tools were put head to head by Akram, using a freshly built multi-domain dataset as the testing ground. What came back was a striking spread in how well each tool performed. One stood clearly above the rest - Originality.ai hit a 97.09% accuracy rate, while the remaining tools fell noticeably short of that mark.

[15] Elkhatat and colleagues took the question into academic territory, putting five detection tools through their paces against content from both GPT-3.5 and GPT-4. A clear pattern emerged - every tool performed better against the older GPT-3.5 output than it did against the newer GPT-4. On top of that, human-written control responses threw the tools off, suggesting the gap between detecting machine and human writing is murkier than expected.

3. Discussion and Analysis

Detecting text generated by AI involves several different methods, each with its own pros and cons. The next few sections look at seven important detection methods that are often used in modern research.

3.1 Supervised Fine-Tuning

Trained on examples marked by humans - some written by people, others by machines - these classifiers learn through labels. RoBERTa or T5 form the backbone of most systems built for telling one type apart from the other. Accuracy climbs above 95 percent if test data looks like what they have seen before. Drop in effectiveness shows up fast once new kinds of machine-made text enter the picture. Their success ties closely to familiar patterns, struggling beyond known sources.

3.2 Feature-Based Statistical Detection

Some approaches use clear language clues like repetition patterns, word variety, surprise factor, flow shifts, readability levels. Instead of needing the original system, they work across different setups fairly well. Easy to follow how decisions are made helps schools or offices adopt them. Yet if someone rewrites text cleverly while keeping meaning intact, those obvious number trails get distorted easily. So when sentences change shape but keep sense, trust in these tools wobbles more than holds steady.

3.3 Zero-Shot Detection (Likelihood Curvature Methods)

Text that feels off might get flagged by checking how likely words are supposed to appear together. Instead of teaching a system from scratch, it uses an existing language model to spot odd patterns. When small changes shake up the predictions too much, something could be artificial. Skipping massive hand-labeled collections helps move faster at setup time. But running these checks again and again eats up processing power fast. If someone twists the wording enough, the method sometimes stops seeing what is wrong.

3.4 Watermarking-Based Detection

During text creation, subtle shifts are built into how tokens appear. Later on, these hidden trends can be spotted to confirm origin. By shaping output early, some tracking load moves upstream. Still, rewording, translating, or condensing tends to strip away those marks. So relying only on such markers will not hold up under manipulation.

3.5 Sentence-Level Detection (Granular Classification)

Most times, spotting AI-written parts means checking how likely each word was to appear. These tools work well when a piece mixes machine-made lines with ones people wrote. Instead of guessing whole sections, they pinpoint fake bits more precisely. Getting such detail often needs raw data from the source model - this step complicates things. Rolling them out across huge systems still pushes limits.

3.6 Domain Generalization Frameworks

Most methods try to build features that stay consistent, even when the model structure changes. Using tricks like competitive updates or comparison-based signals helps cut down reliance on any single large system. Tests show they often handle shifts between models pretty well. Still, tuning them takes more time plus effort than simpler approaches. Even with those hurdles, this path stands out for lasting performance gains.

3.7 Hybrid and Multi-Layered Detection Systems

No single method catches everything - which is exactly why combining them makes sense. Layering labeled data models, pattern spotting, and hidden signal checks means each one covers where another falls short. Results get cross-checked as they pass through each layer, so errors that slip past one technique rarely survive the next. Yes, building something like this takes more effort to set up. But once running, it holds together far better in the messy conditions of real-world use. Tools built with this kind of depth are likely where things are headed - a harder build today in exchange for something that actually lasts.

4. Standardized Comparative Evaluation

Table 1: Standardized Detection Performance Comparison

Method	Accuracy	Generalization	Adv. Robustness	Scalability
Supervised	High	Low	Low	Medium
Feature-Based	Medium	Medium	Low	High
Zero-Shot	High	Medium	Low	Medium
Watermarking	Medium	Medium	Very Low	High
Domain Generalization	High	High	Moderate	High

Table 1 compares different ways of detecting across several evaluation dimensions that are important for real-world use. Supervised systems are very accurate when the conditions are right, but they do not work well with new models. Feature-based systems are easier to understand, but they can still be changed in ways that keep the meaning the same. Zero-shot detection lessens the need to retrain while keeping a good balance between robustness and computational cost. Domain generalization gets fairly balanced results, which shows how flexible its architecture is. The table shows that it is necessary to look at more than one metric, like accuracy, to get a full picture.

5. Robustness Analysis

Getting these tools to hold up under pressure remains the hardest problem to crack in real-world use. Paraphrasing attacks chip away at detection models by swapping words and reshuffling sentence structure - models trained on fixed datasets simply were not built to handle that kind of moving target. Running text through translation and back muddies the language patterns further, leaving models even less sure of what they are looking at. Supervised detectors tend to feel this the hardest, since they lean heavily on surface-level patterns that fall apart the moment someone puts in a little effort to disguise the writing. Domain generalization takes a different angle - rather than chasing surface cues, it anchors itself to deeper meaning patterns that hold steady even when the words around them shift.

Table 2: Performance Degradation Under Adversarial Attacks

Method	Baseline	After Attack	Performance Drop
Supervised	98%	62%	-36%
Feature-Based	85%	55%	-30%
Zero-Shot	91%	60%	-31%
Domain Generalization	95%	80%	-15%

Though small, the dip in performance appears across every detection setup when attacks are introduced. Supervised approaches suffer most sharply - this points to reliance on shallow text patterns. Instead of collapsing completely, feature-driven and zero-shot techniques falter less, yet remain far from robust. What holds up better is domain generalization, thanks to its grasp on deeper structure. Because of these patterns, testing frameworks might need built-in stress tests using adversarial examples plus checks across varied data sources.

6. A Scalable Detection Method

The EAGLE framework shows that Domain Generalization is the best scalable way to find AI-generated text. This is because it directly addresses the main problem with current detectors: they cannot generalize to new generative models. Conventional supervised fine-tuning techniques attain elevated accuracy yet experience considerable performance decline when faced with outputs from recently launched LLMs. EAGLE, on the other hand, learns generator-invariant representations that stay the same across different model architectures. This lets it find text from advanced systems like GPT-4 and Claude without having to be trained on their data. This means that it is not necessary to keep making large labeled datasets for each new model, which makes the method more useful and long-lasting for use in the real world. Empirical results further show that EAGLE performs within 4.7% of a fully supervised detector while still being very robust across different models. This shows that domain generalization is a forward-looking and scalable solution for the growing challenges of AI text detection.

7. Future Directions

If there is one thing future research cannot afford to ignore, it is adversarial robustness. Training needs to account for the kinds of attacks that actually show up in the wild - paraphrasing, translation, style-transfer - not just clean, well-behaved inputs. A detector that crumbles the moment meaning is preserved but wording is shuffled around is not ready for anything beyond a controlled setting. Cross-model generalization deserves just as much attention. The goal should be systems that can read the fingerprints of a newly released model without needing to be retrained from scratch each time - built on representations deep enough to travel across architectures rather than ones tied to the surface patterns of models already known.

Combining supervised classifiers, statistical feature analysis, and watermark verification into multi-layered detection frameworks may make them more resilient than single-method approaches.

Also, making large, standardized benchmark datasets that cover a wide range of domains, languages, and adversarial conditions will make it possible to fairly and consistently test detection models.

Lastly, future research should look into the theoretical limits of detectability to see if very advanced AI-generated text can always be different from human writing or if detection performance will eventually level off to random levels.

8. Conclusion

Large Language Models have moved fast, and detection research has been scrambling to keep up ever since. Supervised fine-tuning, feature-based analysis, zero-shot detection, watermarking, domain generalization - each holds its own in a controlled setting, but cracks start to show the moment text gets deliberately reworked or a different model enters the picture. No single approach comes out on top across every situation. Holding up in the real world demands more than just accuracy - it takes a system that can generalise, scale, and withstand attempts to throw it off. As generative models keep evolving, detection systems that rely on a single method or inconsistent evaluation standards will struggle to stay relevant. Multi-layered architectures and shared benchmarks are not optional extras - they are what lasting reliability actually looks like.

References

1. K.C. Fraser, H. Dawkins, S. Kiritchenko, "Detecting AI Generated Text: Factors Influencing Detectability with Current Methods", Journal of Artificial Intelligence Research, 2025, 82, 2233-2278.
2. S. Abdali, R. Anarfi, C.J. Barberan, J. He, "Decoding the AI Pen: Techniques and Challenges in Detecting AI-Generated Text", Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '24), 2024.

3. A. Boutadjine, F. Harrag, K. Shaalan, "Human vs. Machine: A Comparative Study on the Detection of AI-Generated Content", ACM Transactions on Asian and Low-Resource Language Information Processing, 2025, 24 (2), Article 12.
4. T.A. Mohamed, M.H. Khafagy, A.B. Elsedawy, A.S. Ismail, "A Proposed Model for Distinguishing Between Human Based and ChatGPT Content in Scientific Articles", IEEE Access, 2024, 12, 121251-121260.
5. K. Schaaff, T. Schlippe, L. Mindner, "Classification of Human- and AI-Generated Texts for Different Languages and Domains", International Journal of Speech Technology, 2024, 27, 935-956.
6. C. Chaka, "Detecting AI Content in Responses Generated by ChatGPT, YouChat, and Chatsonic: The Case of Five AI Content Detection Tools", Journal of Applied Learning & Teaching, 2023, 6 (2).
7. C. Chaka, "Reviewing the Performance of AI Detection Tools in Differentiating Between AI-Generated and Human-Written Texts: A Literature and Integrative Hybrid Review", Journal of Applied Learning & Teaching, 2024, 7 (1).
8. V.S. Sadasivan, A. Kumar, S. Balasubramanian, W. Wang, S. Feizi, "Can AI-Generated Text be Reliably Detected?", arXiv preprint arXiv:2303.11156v4, 2025.
9. S. Chakraborty, A.S. Bedi, S. Zhu, B. An, D. Manocha, F. Huang, "On the Possibilities of AI-Generated Text Detection", arXiv preprint arXiv:2304.04736v3, 2023.
10. P. Wang, L. Li, K. Ren, B. Jiang, D. Zhang, X. Qiu, "SeqXGPT: Sentence-Level AI-Generated Text Detection", arXiv preprint arXiv:2310.08903v2, 2023.
11. A. Bhattacharjee, R. Moraffah, J. Garland, H. Liu, "EAGLE: A Domain Generalization Framework for AI Generated Text Detection", arXiv preprint arXiv:2403.15690v1, 2024.
12. D. Weber-Wulff, A. Anohina-Naumeca, S. Bjelobaba, T. Foltynek, J. Guerrero-Dib, O. Popoola, P. Sigut, L. Waddington, "Testing of Detection Tools for AI-Generated Text", International Journal for Educational Integrity, 2023, 19 (26).
13. R. Kumar, M. Mindzak, "Who Wrote This? Detecting Artificial Intelligence-Generated Text from Human-Written Text", Canadian Perspectives on Academic Integrity, 2024, 7 (1), 1-9.
14. A. Akram, "An Empirical Study of AI Generated Text Detection Tools", Advances in Machine Learning & Artificial Intelligence, 2023, 4 (2), 44-55.
15. A.M. Elkhatat, K. Elsaid, S. Almeer, "Evaluating the Efficacy of AI Content Detection Tools in Differentiating Between Human and AI-Generated Text", International Journal for Educational Integrity, 2023, 19 (17).