UDC: 004.8 https://doi.org/10.15407/jai2024.04.233

A. Nykonenko
Cherkasy State Technological University
460, Shevchenka Blvd, Cherkasy, Ukraine, 18000
andrey.nikonenko@gmail.com
https://orcid.org/0000-0002-9442-1601

HOW TEXT TRANSFORMATIONS AFFECT AI DETECTION

Abstract. This study addresses the critical issue of AI writing detection, which currently plays a key role in deterring technology misuse and proposes a foundation for the controllable and conscious use of AI. The ability to differentiate between human-written and AI-generated text is crucial for the practical application of any policies or guidelines. Current detection tools are unable to interpret their decisions in a way that is understandable to humans or provide any human-readable evidence or proof for their decisions. We assume that there should be a traceable footprint in LLM-generated texts that is invisible to the human eye but can be detected by AI detection tools-referred to as the AI footprint. Understanding its nature will help bring more light into the guiding principles lying at the core of AI detection technology and help build more trust in the technology in general. The main goal of this paper is to examine the AI footprint in text data generated by large language models (LLMs). To achieve this, we propose a new method for text transformation that should measurably decrease the AI footprint in the text data, impacting AI writing scores. We applied a set of stage-by-stage text transformations focused on decreasing meaningfulness by masking or removing words. Using a set of AI detectors, we measured the AI writing score as a proxy metric for assessing the impact of the proposed method. The results demonstrate a significant correlation between the severity of changes and the resulting impact on AI writing scores, highlighting the need for developing more reliable AI writing identification methods that are immune to attempts to hide the AI footprint through subtle changes.

Keywords: AI Writing Detection, LLMs, AI-Generated Text, Text Transformations, AI Footprint, AI Detectors, Ethical Use of AI.

Introduction

During recent years the progress achieved by neural network-based technologies is very significant, this includes rapid improvements in image recognition, text-to-speech, speech-to-text technologies, natural language understanding, reasoning, machine reading comprehension, and many other areas. However, the most essential results have probably been received in the text generation area, with the increasing growth of LLMs' popularity. This technology seems to be able to address multiple tasks that were previously considered as separate, distinct domains that should be addressed by different scientific approaches and techniques. No one expects such a serious breakthrough from a generative approach that already allows us to solve a multitude of tasks and boundaries of this technology are only to be defined later. We can say that LLM's rise is probably one of the biggest findings of recent years and its full potential is to be discovered.

As with any new technology not all of its applications are meant for a great good and there are a lot of potential risks associated with the widespread use of the LLMs technology. Generative AI is capable of improving our day-to-day life significantly, giving a more natural and effective way of receiving information, reducing the amount of effort needed for daily routines, and automating some simple tasks and it's only the beginning. On the other hand, bad actors found their own ways of using the technology, this leads us to questions related to the regulations of AI to ensure ethics and to set up legal ways of using it. While different countries are making significant efforts towards creating AI regulations to keep the exponential growth of the technology under control, some companies and individuals are already using new opportunities to get some profit in a not-very-ethical way. The use of AI for social manipulation, fake information spreading, academic misconduct, increasing bias, privacy violations, security risks, and many others are among the biggest threats related to the rise of AI.

This study is related to the topic of AI writing detection which currently plays a key role in the deterrence of technology misuse and proposes a foundation for the controllable and conscious use of AI. The ability to being able to differentiate between human-written and AI-generated text is crucial for the practical application of any policies or guidelines. In research [1] authors state that, without training, humans can distinguish between LLM-generated (GPT3) and human-written text at a random chance level. Regarding the ability of humans to be trained on this task authors say: "We explore three approaches for quickly training evaluators to better identify GPT3-authored text (detailed instructions, annotated examples, and paired examples) and find that while evaluators' accuracy improved up to 55%, it did not significantly improve across the three domains". Nowadays, GPT3 is deprecated and newer models are significantly more capable of generating grammatically correct, well-structured, meaningful, and engaging text, which makes it even harder to identify by the human eye. That's why AI Writing detection is the only reliable option that could help us control the spread of AI and ensure its ethical and responsible use.

Related works

Despite its novelty and because of its importance the AI writing detection topic is an active research area whose foundation is discovered quite well. The guiding principles of building AI detection tools, technological core, and main approaches are widely discussed. Evaluation of existing detectors, their performance in different scenarios and corner cases, AI detection avoidance techniques, and many more topics are covered in the literature as well. Here we are giving a brief summary of the current AI Detection landscape with a focus on the existing gap we are addressing in the current study.

There is a set of well-defined approaches that could be used as a foundation for building the AI writing detector. Authors of [2] and [3] propose the following classification:

1. Feature-Based Approaches. These methods are based on a feature engineering approach from the classical NLP to convert a text to a set of feature vectors and then use them for training a model for classification tasks. Any classical or Neural network-based model can be used as an ML backend for these methods.
2. Neural Language Model Approaches. This set of methods is based on the usage of existing LLMs as detectors without additional training (zero-shot learning case) or with some additional training (fine-tuning case). The typical choice of the basic model could be any state-of-the-art LLM or Transformer, but the most common choice is to use a Transformer architecture due to its smaller size and (because of this) lower cost per prediction.
3. Feature-Based + Neural Language Model. This approach represents a mixture of two previous methods, where we use LLM as a feature generator and Neural Network or Transformer as an ML backend.
4. Watermarking. The whole idea behind this method is about having a dictionary of specifically marked tokens and obligating the generative model to use them. In this case, the whole detection is reduced to counting the amount of marked tokens in the text.
5. Human-aided. This is a set of methods based on using human expertise supported by a set of additional tools (e.g. GLTR [4]) capable of analyzing textual data and expressing insights in a human-readable form.
6. Information retrieval-based. This approach is proposed in [5] and is based on the idea of storing all LLM-generated texts in a centralized DB. Later on when one needs to identify whether text AI generated or not, all they need to do is to check the existence of the text in question in the DB.

On the other side, in the evaluation of existing commercial and open-source detectors, there is a lot of scientific research focused on different use cases, domains, and specific techniques. Work [6] discusses issues related to the reliability of AI detectors in practical scenarios. In [7] authors analyze the topic of the effectiveness of software designed to detect AI-generated writing using a massive evaluation of 16 AI Detectors. Among their main findings, three of the tested detectors have very high accuracy, while most of the other detectors can distinguish between GPT-3.5 papers and human-generated papers with reasonably high accuracy. The author of [8] conducted an analysis to evaluate the amount of AI-generated papers on Arxiv using physics, mathematics, and computer science articles. They state that the increase of AI-labeled papers after the ChatGPT release is approximately 3% - it rose from 4.38% up to 7.37% in one year.

Relevance of the Research

Despite good progress in research in both AI Detection tools creation and their detailed and independent evaluation, there is still a gap in this field we aim to target with the current research. This gap is connected to the fact that the current generation of detection tools is unable to interpret their decisions in a way that is understandable to humans or to provide any human-readable evidence or proof they have based their decisions. As we mentioned earlier, based on available research, humans aren’t able to identify AI-generated text without specific tools, which means that they can’t double-check on their own the decision made by an AI Detector. This leads to eroded trust and doubts about AI Detection technology in general. We assume that there should be a traceable footprint in LLM-generated texts that isn’t visible to the human eye but can be seen by AI Detection tools. We call that trace AI footprint and we believe that understanding its nature will help us bring more light into guiding principles lying at the core of AI detection technology, produce visible evidence, propose improvements to the way of building a new generation of AI detectors, and help to build more trust in the technology in general. The main goal of this paper is to examine the AI footprint in text data generated by large language models (LLMs). To achieve this, we aim to develop new methods, test them on real data, and analyze their effectiveness using existing AI detection tools.

Scientific Novelty

Most of the existing research in the analyzed area is focused on topics of possibility of the detection in general, some specific use cases, like adversarial prompting and impact of detection avoidance techniques, and accuracy assessment of individual detectors. We propose a new method for text transformation that presumably should allow us to measurably decrease the AI footprint in the text data impacting AI writing scores for analyzed data. Based on our previous study we found some key principles related to text manipulations aimed to raise the AI writing score, here we propose another study focused on reducing AI writing score. We propose a set of stage-by-stage text transformations that presumably should decrease the AI footprint. We use a set of AI detectors to measure the AI writing score as a proxy metric for measuring the impact of the proposed method.

Method

In this study, we propose a new method for AI-generated text transformations focused on decreasing the AI footprint by decreasing the meaningfulness of the text. Our previous research [9] showed that there is a reasonable connection between generated text meaningfulness and its AI score. In that research, we used the noise-to-text transformation method to measure the impact of changes on the AI score. This time we propose a different approach: we start with meaningful AI-generated text and apply different transformation techniques intended to decrease the meaningfulness. We propose a few techniques, ranging from minor changes to relatively significant adjustments.

The method consists of 6 stages, where the first stage “Stage6” reuses the same data we have generated in the final stage of the previous research. For research consistency and easier understanding we preserve the naming convention for the data transformation stages we have used in the previous study. Other 5 stages, based on transformation type, could be divided into two major stages: Stage6A and Stage6B. Stage6A describes different text manipulations with a limit of two changes per sentence, while Stage6B proposes up to 4 changes per sentence, making changes more severe.

Stage6A contains three substages: Stage6A1, Stage6A2, and Stage6A3. Where Stage6A1 is responsible for masking up to two nouns in each sentence. In this research, we use “#” character for masking, but probably any other character could be used instead without a significant change in the results. Each word selected for masking was programmatically replaced by a sequence of “#” characters, where the length of the sequence exactly corresponds to the number of characters in the original masked word. We decided to follow this rule to preserve the original sentence and text lengths and exclude their impact on the evaluation results. Stage6A2 makes the same changes as Stage6A1, but instead of masking words, it removes them completely. Stage6A3 makes the same changes as Stage6A1, but replaces nouns with random words with the same length. Stage6B contains two substages: Stage6B1, Stage6B2. The first of them, Stage6B1 is responsible for masking up to 4 nouns in each sentence. Masking happens in the same way as it was done in A stages. Stage6B2 is about masking up to 2 nouns and 2 verbs in each sentence.

All the changes were done using Python, NLTK package, and WordNet as a word dictionary. This is a significant difference between this research and the previous one [9] where all changes were done using LLM only. For nouns, we refer to anything with POS tags NN and NNS, and for verbs, we refer to anything with POS tags in this list: (VB, VBD, VBG, VBN, VBP, VBZ). We applied all the discussed transformations to the same 10 texts from the final stage (Stage6) of the previous research, to have consistency in results. Final versions of the data created in each proposed stage are available here [10]. When data is ready we send it through a set of AI Detectors to get an AI score.

Detectors

Using the results of the previous research [9] we decided to update the list of AI detectors we will be using in our experiments. Although it would be better to stay with a consistent list of detectors for easier research reproducibility, there are a set of reasons that make it reasonable to update the list.

As the previous research showed, the AI detectors have a different sensitivity threshold, meaning that some of them are more susceptible to AI-generated text, while others tend to mark it as human till there is no doubt it was AI-generated. We can’t state that detectors with a high sensitivity threshold have a pure quality versus detectors with a lower threshold, this is far away from the truth. The real reason is a business case for a detector, as depending on usage there is a different price for a False Positive (FP) and False Negative (FN) prediction. For example, for detectors used in education like Turnitin’s solution [11] FP means false accusation of a student which could have very bad consequences for the student, institution, and Turnitin. On the other hand, FN means just missed cases of AI usage which isn’t great from the academic integrity standpoint and the tool’s overall performance, but significantly less important if recall in general stands pretty high. So we can expect that in general for academic integrity-focused solutions sensitivity threshold should stay high, forcing them to make a decision in the student’s favor in all the cases except obvious, very highly probable AI writing. Which is good for their business use case, but makes those detectors less interesting from our research perspective.

Talking about other tools that are focused on masking AI footprint, the main goal of their AI detection is as low FN rate as possible. AI detection serves as an additional capability for them with the primary aim to ensure customers that produced output is undetectable. That’s why FPs are less important for them, as FP means just that the customer needs to use their main tool one more time. While FN could put a customer at risk of being caught by an integrity solution, with significant consequences to his educational career. That’s why there is a general expectation that this type of tool should be more sensitive. But it’s not always the case, for example, the previous research showed that the most sensitive tool among all discovered was Originality [12], which belongs to the integrity-focused group. The possible explanation could be that for masking AI footprint solution AI detection is just an additional capability that doesn’t bring them money, so the amount of effort spent on it’s development could be relatively low, compared to edtech companies where it’s part of the main business.

Another reason why we decided to review the detectors list is the emergence of new detection techniques like [13], so it would be worth it to include them in consideration. The same as previously for research purposes we are leaning towards detectors with a better granularity, as it is vital to see the impact of even very small changes. To follow this principle we removed from consideration all the detectors with not a numerical output. Also, we excluded from the consideration some of the detectors from the previous research based on their performance on Stages 5 and 6.

Trying to add new detectors to our list we were trying to evaluate their sensitivity using some of the examples of data from Stages 5 and 6 as a test.

– We tested Hive [14], but it tends to assign scores close to 0% AI for data from Stage 5 and scores close to 100% AI for data from Stage 6, meaning that its sensitivity threshold is too high for the current research.

– We were trying to use Plagiarism Check detector [15], but they offer only 5 pages for free, so we tested them using just 3 texts from Stage 5, two were detected as 0% AI and one as 100% AI. Not very consistent, but the sample was too small to tell with certainty. We decided not to include it on the list.

– We tested Grammarly AI Detector [16] on some of our texts, and it seems its sensitivity is good enough, so we are going to include it in the list.

– Also, we decided to add [17] due to their interesting mass-scale approach to training the detector. We tested it on some samples from our data and it seems to be overconfident on some examples from Stage 5, producing scores close to 100%. At the same time in some other examples from stage 5 it’s underconfident, producing scores close to 0%, the same behavior was observed in a few examples from Stage 4. We decided to proceed with this detector, but with the possibility to exclude it from the research later in case if in other experiments it continues to perform more like a binary classifier (producing only 0% or 100% scores).

Authors of [18] proposed RAID - Shared Benchmark for AI Detectors evaluation, so we took the top 3 detectors from their leaderboard for our testing. There is no guarantee that they will be able to perform well on our task, but it is worth trying. For preliminary evaluation, we used the same strategy as for the detectors mentioned earlier.

– Recognized some of the texts from Stage 5 as “likely AI written” and some as “human written”, so it's not able to produce a numerical output [19].

– Binoculars [20] was able to detect some of the texts from stage 5 as “Most likely human-generated” and “Most likely AI-generated”, so the system could be sensitive, but doesn’t satisfy our numerical output criteria.

– Radar Tester [21] proposes very interesting detection capabilities by using up to 4 models as detectors, where each model produces its own numerical output. Most of the models produced pretty high scores for data from Stage 5 and even higher scores for data from Stage 6. We also tested them on data from Stage 4 and received scores were significantly lower compared to scores from latter stages. The system seems to be a good candidate.

The final list of the detectors we will be using in the current research is shown in Tab. 1.

Table 1. List of the AI detectors used in the research

Scribbr
GPTZero
Quillbot
Originality AI
Pangram
Grammarly AI Detection
Radar Tester

Results

Here is a summary of stats representing the average AI writing score per stage. Our main intent was to try different text manipulation techniques to reduce the AI score. In table 2 we see final stats collected across all the stages.

– At the very beginning before applying any modifications at Stage6 we have an average AI score of 85% which is pretty high. The whole content at that stage is AI generated with meaningful and cohesive sentences inside paragraphs but without connection between paragraphs.

– At Stage6A1 we masked up to two nouns in each sentence, which led to a slight decrease in AI score to 80%. It is worth mentioning that the applied masking technique significantly decreased text meaningfulness and cohesion, but only slightly changed the AI score.

– Stage6A2 represents the same change, as it was done at Stage6A1 but with the complete removal of masked words. Definitely, the impact on the text was more severe than in the previous stage and the AI score was significantly lower - 64%. This is the biggest decrease in the score among all the stages.

– Stage6A3 deals with replacing selected words with random words instead of masking. It is interesting because it could impact the meaning of the text more significantly compared to masking, but on the other hand impact on the text structure is less severe. This stage received an 82% AI score, making it even higher than on original Stage6A1. We can restate that by saying that replacing some nouns in the text with random words has the least impact on the AI score among all other proposed techniques.

– Stage6B1 has the second biggest impact on the AI score, it actually does the same thing as Stage6A1, but on a bigger scale masking instead of 2 up to 4 nouns per sentence. The effect is more severe.

– Stage6B2 has the third biggest (or average among all 5 transformations) impact on the AI score. It works with the same transformation as Stage6A1, but applies it to both nouns and verbs. In general, we expect that the number of words replaced on Stage6B1 and Stage6B2 should be in the same ballpark, which could explain very close AI scores.

Regarding the performance of the detectors, there are a few interesting observations to mention. Detailed information is available in Table 3:

– Grammarly seems to be the most sensitive one to any transformation, even the smallest transformation proposed in stage Stage6A1 dropped the score from 100% to 50%, and all other transformations made it even lower.

– Pangram and Radar.Vacuna seem to be two the least sensitive, their scores slightly deviating between 80+% and 100%, typically staying in the 90%-100% range.

Table 2. Summary of received AI writing scores after applied transformations

	Stage6	**Stage6A1**	**Stage6A2**	**Stage6A3**	**Stage6B1**	**Stage6B2**
Stage Result	**85%**	**80%**	**64%**	**82%**	**68%**	**70%**

Table 3. Average scores per stage per AI detector

Stage	Scribbr	GPTZero	Quillbot	Grammarly	Radar				Originality	Pangram	Average
					Dolly V2	Camel	Dolly V1	Vicuna			
Stage6	87%	87%	81%	100%	100%	49%	58%	97%	97%	97%	**85%**
Stage6A1	80%	90%	80%	56%	52%	72%	87%	95%	91%	98%	**80%**
Stage6A2	60%	39%	60%	18%	51%	71%	95%	96%	57%	88%	**64%**
Stage6A3	86%	75%	86%	37%	85%	71%	92%	98%	88%	98%	**82%**
Stage6B1	67%	85%	67%	30%	46%	42%	70%	81%	94%	93%	**68%**
Stage6B2	66%	88%	66%	18%	48%	53%	83%	94%	86%	97%	**70%**
Average	**75%**	**77%**	**73%**	**43%**	**64%**	**60%**	**81%**	**93%**	**86%**	**95%**	**75%**

– Originality seems to be the third least sensitive with very stable scores across all the stages except for Stage6A2. This is an interesting observation, as in our previous study it has shown the highest sensitivity.

– In general among all the detectors except the Radar family Stage6A2 received the lowest scores among all the stages. For the Radar family, Stage6B1 received the lowest scores.

– Radar.Camel performs in a very unexpected way, its score for Stage6 is the lowest and the score for the Stage6A2 is the highest. Which is exactly the opposite of the agreement between all 10 AI detectors.

Discussion

The main goal of this study was to analyze how different text transformations affect modern AI detectors' ability to identify LLM-generated data. With the help of systematic changes of AI-generated data aimed to reduce its meaningness and structural consistency, we were trying to discover the nature of the AI footprint. Where AI footprint is the subtle set of characteristics that is visible to AI detectors and used by them for distinguishing between AI-generated and human-written text, but invisible to a human eye.

Received results demonstrate a significant correlation between the severity of changes and resulting impact on AI writing score produced by a set of analyzed detectors. Particularly, more significant changes lead to a bigger decrease in AI scores. For example, Stage6A2 where up to 2 nouns were removed from each sentence shows a significant (more than 20%) decrease of the average AI score from 85% to 64%. This could suggest that removing key grammar elements of a sentence significantly decreases AI detectors' ability to recognize typical textual patterns added by an LLM.

Another interesting observation is that Stage6A3 responsible for the replacement of some of the nouns with random words showed a significantly lower impact on AI score resulting in 82%. This shows that despite changes in text meaning, structural patterns and character/token statistical distributions remain mostly unchanged. This supports the results presented in [9] regarding detectors' ability to rely not only on semantic coherence but also on hidden statistical patterns.

Detectors’ different sensitivity levels and in some cases, contrary behavior emphasizes the difficulty of AI detection tasks where it could be relatively hard to demonstrate consistent behavior in different scenarios.

Detectors like Grammarly showed a significant sensitivity even for small changes in a text, with a sharp drop in AI scores as a reaction to minor changes. Some other detectors, like Pangram and Radar.Vicuna stay with consistently high scores for most of the stages, which could suggest reliance on other types of features that are less affected by proposed transformations. In general, different sensitivity demonstrated by detectors could serve as evidence that they rely on different features, but despite that, they are all able to demonstrate a pretty high accuracy. This means that AI footprint as a phenomenon could be a pretty complex entity existing in a multitude of dimensions, but being able to identify and catch even a subset of its characteristics could be enough for a detector to perform relatively well on a standard set of circumstances. This suggests that future research should focus on exploring more edge cases to identify gaps in individual detectors. If a transformation reveals a gap in one detector but doesn't affect the score of another, it indicates that the targeted change is part of the AI footprint captured by the detector that remains unaffected. This could be a method to implicitly reproduce key elements of the AI footprint and gain a deeper understanding of its nature.

Conclusion

In this study, we have explored the impact of text transformations related to the decrease of meaningfulness on the AI writing score identified by different AI detectors. Our findings show that significant changes in grammar structure and text coherence could significantly reduce the AI score and potentially make text undetectable. This finding can serve as evidence that the AI footprint used by detectors is sensitive to changes in both structural patterns (like syntax) and semantics.

Received results highlight the need for development of the more reliable methods of AI writing identification, that should be immune to attempts to hide the AI footprint through subtle changes. Transparency increase for detection tools and techniques is also a crucial step on a way to building more trust with customers and making it possible for them to understand key elements and the logic behind a detection process.

In conclusion, as AI models become bigger the quality of generated text is rapidly increasing, but together with its quality diversity and amount of obfuscation techniques are rising as well. AI researchers and developers of AI detectors must incorporate recent findings in the field to make the detectors' decisions explainable and interpretable while staying immune to text manipulations and obfuscations. A better understanding of the nature of AI footprint could bring significant progress towards the ethical use of AI and better safeguard positive applications.

References

1. Clark, E., August, T., Serrano, S., Haduong, N., Gururangan, S., & Smith, N. A. (2021). All that's' human'is not gold: Evaluating human evaluation of generated text. *arXiv preprint arXiv:2107.00061*.
2. Homolak, J. (2023). Exploring the adoption of ChatGPT in academic publishing: insights and lessons for scientific writing. *Croatian Medical Journal*, *64*(3), 205.
3. Pan, W. H., Chok, M. J., Wong, J. L. S., Shin, Y. X., Poon, Y. S., Yang, Z., ... & Lim, M. K. (2024, April). Assessing AI Detectors in Identifying AI-Generated Code: Implications for Education. In *Proceedings of the 46th International Conference on Software Engineering: Software Engineering Education and Training* (pp. 1-11).
4. Gehrmann, S., Strobelt, H., & Rush, A. M. (2019). Gltr: Statistical detection and visualization of generated text. *arXiv preprint arXiv:1906.04043*.
5. Krishna, K., Song, Y., Karpinska, M., Wieting, J., & Iyyer, M. (2024). Paraphrasing evades detectors of ai-generated text, but retrieval is an effective defense. *Advances in Neural Information Processing Systems*, *36*.
6. Sadasivan, V. S., Kumar, A., Balasubramanian, S., Wang, W., & Feizi, S. (2023). Can AI-generated text be reliably detected? *arXiv preprint arXiv:2303.11156*.
7. Walters, W. H. (2023). The effectiveness of software designed to detect AI-generated writing: A comparison of 16 AI text detectors. *Open Information Science*, *7*(1), 20220158.
8. Akram, A. (2024). Quantitative Analysis of AI-Generated Texts in Academic Research: A Study of AI Presence in Arxiv Submissions using AI Detection Tool. *arXiv preprint arXiv:2403.13812*.

9. Faure, E., Nykonenko, A. (2024). Noise-to-text method in evaluation of AI-generated texts analysis. *Proceedings of the 1st International Scientific and Practical Conference on Computational Intelligence and Smart Systems*. Lviv, Ukraine.

10. A. Nykonenko, How Text Transformations Affect AI Detection. Data and experiments. URL:https://docs.google.com/spreadsheets/d/1G6kVXCiKa_9aVdgVcD5wxb0hwDFndgp5quudxuUJgc/edit?usp=sharing

11. Turnitin AI Technical Staff. (2023). *Turnitin's AI writing detection model architecture and testing protocol*. Turnitin. https://www.turnitin.com/

12. "AI Content Checker and Plagiarism Check|GPT-4 | ChatGPT." Accessed: Sep. 15, 2024. [Online]. Available: https://originality.ai/

13. Emi, B., & Spero, M. (2024). Technical Report on the Checkfor. ai AI-Generated Text Classifier. *arXiv preprint arXiv:2402.14873*.

14. Hive https://hivemoderation.com/ai-generated-content-detection

15. TraceGPT https://plagiarismcheck.org/

16. Grammarly AI Detector https://www.grammarly.com/ai-detector

17. Emi, B., & Spero, M. (2024). Technical Report on the Checkfor. ai AI-Generated Text Classifier. *arXiv preprint arXiv:2402.14873*.

18. Dugan, L., Hwang, A., Trhlik, F., Ludan, J. M., Zhu, A., Xu, H., ... & Callison-Burch, C. (2024). RAID: A Shared Benchmark for Robust Evaluation of Machine-Generated Text Detectors. *arXiv preprint arXiv:2405.07940*.

19. It's AI https://its-ai.org/

20. Hans, A., Schwarzschild, A., Cherepanova, V., Kazemi, H., Saha, A., Goldblum, M., ... & Goldstein, T. (2024). Spotting llms with binoculars: Zero-shot detection of machine-generated text. *arXiv preprint arXiv:2401. 12070*. Hu, X., Chen, P. Y., & Ho, T. Y. (2023). Radar: Robust ai-text detection via adversarial learning. *Advances in Neural Information Processing Systems*, *36*, 15077-15095.

The article has been sent to the editors 18.10.24.
After processing 27.10.24.
Submitted for printing 30.12.24.