Proceedings Paper DOI: 10.58190/icat.2023.28

PROCEEDINGS OF

INTERNATIONAL CONFERENCE ON ADVANCED TECHNOLOGIES

https://proceedings.icatsconf.org/

11th International Conference on Advanced Technologies (ICAT'23), Istanbul-Turkiye, August 17-19, 2023.

A Framework for Detecting AI-Generated Text in Research Publications

Paria Sarzaeim, Aarya Mayurpalsingh Doshi, Qusay H. Mahmoud

Faculty of Engineering and Applied Science

Ontario Tech University

Oshawa, ON, L1G 0C5, Canada

{paria.sarzaeim, aaryamayurpalsingh.doshi,qusay.mahmoud}@.ontariotechu.net

***Abstract*— The use of generative artificial intelligence is becoming increasingly prevalent in creating content in various formats such as text, video, and image. However, there is a need to distinguish between content that has been generated by humans and content that has been generated by AI as misuse of these technologies can raise scientific and social challenges. Moreover, there are concerns about the reliability and comprehensiveness of the content generated by AI without human validation. This paper presents a framework for AI-generated text. The prototype implementation of the proposed approach is to train a model using predefined datasets and deploy this model on a cloud-based service to predict whether a text was created by a human or AI. This approach is specifically focused on assessing the accuracy of scientific writings and research papers rather than general text. The proposed framework is compared with recently developed tools such as OpenAI Text Classifier, ZeroGPT, and Turnitin. The results show that training a text classifier can be highly useful in detecting whether a text is written by a human or AI. The source code and dataset are made open source so others can experiment with the prototype implementation and use it for future research.**

***Keywords*— Generative artificial intelligence, research papers, machine learning, AI-generated text**

I. INTRODUCTION

Artificial intelligence (AI) has recently made significant strides, particularly in the development of generative models for computer vision and natural language processing (NLP) [1]. These advancements have resulted in the creation of highly advanced models capable of generating hyper-realistic content in various forms, such as text, images, and video. These generative models have shown great potential in applications such as content creation, chatbots, and virtual assistants, and are increasingly being utilized in industries ranging from healthcare to finance [2].

Large language models (LLMs) are now capable of producing high-quality texts with numerous potential applications, including writing codes, completing documents, and answering questions. They also have shown that they can improve over time. While these technologies can be useful as AI writing assistance and autocomplete tools [3], using them also poses significant challenges, including issues such as plagiarism, generating fake news, and manipulating web content which can have negative societal impacts [1]. For example, this content may be used to manipulate public opinions [4]. Therefore, people start to be concerned about the texts being written by humans or AI and question the reliability of the content as shown in Fig. 1 adapted from [4].

ChatGPT is one of the widely used LLMs and has shown a lot of improvements in a considerably short time. It has shown traits of innovation by answering questions and generating new content like stories and poems. It has also passed the United States Medical License Examination theory part and has been reported to be listed as an author in some manuscripts received by Nature publications [5]. However, in the scientific realm, generating accurate and insightful scientific text through LLMs presents a unique set of challenges as scientific text must provide novel and original insights to readers. The precision and reliability of employing such models in scientific writing remain unclear and controversial [6]. Additionally, there are concerns regarding the completeness of information generated by AI. The information they provide may not be sufficient enough to use them in practical applications [5].

Moreover, the ability of these models to generate human-like content presents a range of technical and social challenges. Misuse of AI technology can result in significant issues such as the spread of disinformation and information fraud [7], and there is a risk of passing off AI-generated content as the user's work and submitting it to conferences or journals [6]. The problem of significant concern here is the plagiarism of original content, which has been prevalent in AI journalism even though ChatGPT itself is not involved [7].

This issue raises ethical concerns, such as whether there should be a defined limit to the amount of AI-generated content

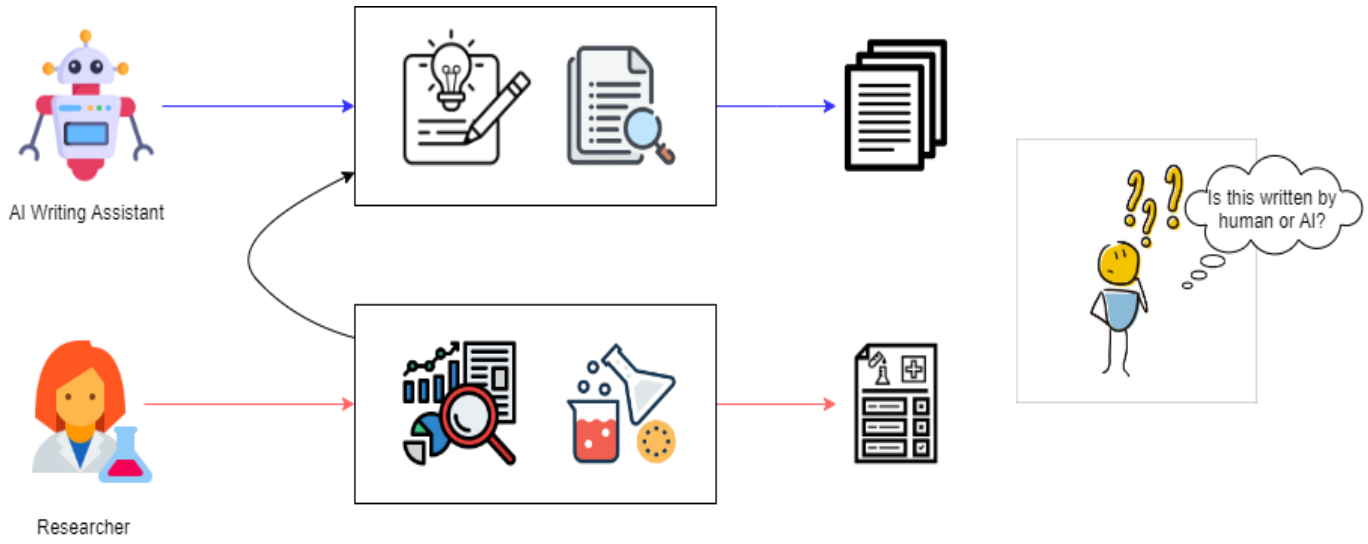

Fig. 1. With the spread of using generative AI technologies in generating text, people start to question the text they read wondering if it written by humans or AI; however, detecting AI-generated text can be challenging.

to be considered appropriate. Furthermore, the personification of AI models like ChatGPT can be a controversial topic among a wider audience [5].

Equity concerns arise when considering the potential future cost of using ChatGPT and similar AI tools. Currently, the early version of ChatGPT is free to use, but ChatGPT-4 is a paid service and there is a possibility that new tools may become prohibitively expensive subscription-based services due to their popularity. This could result in an unequal distribution of resources among researchers [5].

Several efforts have already been made to detect AI-generated content, including ZeroGPT [8], OpenAI Text Classifier [9], and Turnitin [10]. Some of these tools such as OpenAI Text Classifier do not provide definitive answers. Challenge with the reliability of detecting AI-generated text have been raised in several papers that showed that paraphrasing the output text generated by an AI can evade these detectors. They also performed experiments on the available methods in the literature and found that watermarking detectors can also fail due to spoofing attacks in which an adversary may generate a non-AI text that is detected to be AI-generated, so the rate of false positive of the detection model will increase. They argue that the cost of a false positive detection could be huge as humans could be wrongly accused of plagiarism or the reputation of NLG models' developers could be damaged [11] [12] . They also emphasize the difficulty in distinguishing AI-generated text from human-generated text when the total variation (TV) norm shows only a slight difference between the distributions of the two [11].

However, recent research showed that while paraphrasing attacks can reduce the detection performance, there is still always a hidden possibility to detect AI-generated text by collecting more samples or sentences [13].

Therefore, it is essential for publishing companies to be aware of this issue and address these challenges by using novel protection tools for ensuring the ethical use of AI technology. This paper presents the design and development of a cloud-based solution to detect AI-generated content. A proof of concept prototype has been constructed and the machine learning model was trained on a dataset containing human-generated content from the research literature as well as AI-generated text.

The rest of this paper is organized as follows. Section II discusses the related work. The proposed framework, solution architecture and prototype implementation are presented in Section III. Evaluation results are discussed in Section IV. Finally, conclusions and ideas for future work are presented in Section V.

II. Related Work

As AI technology continues to advance, it is crucial to address the potential impact on scientific writing and authorship. The possibility of AI-generated text being mistaken for human-generated text poses a challenge for publishers and researchers. Some publishers have already implemented bans on AI text-generation tools, while others are considering the use of paywalls or login credentials to prevent AI from scraping articles.

Currently, authorship guidelines for many journals and publishers do not explicitly mention AI-generated text, but this may change in the future. Springer Nature has already implemented a ban on using AI text generation tools and other publishers and editorial boards may follow suit [5]. Another approach to consider is to put articles behind a free paywall or require login credentials to prevent AI from scraping the articles. Although this may seem contrary to open science principles, some publishers and academic organizations are already considering this approach [5]. Therefore, there is a need to address this solution by developing software and training AI

models to be capable of distinguishing AI and human-generated text.

When it comes to the topic of artificial intelligence and scientific writing, several related works have explored various aspects, but most of the current works in AI-generated texts focus on news text or online text [6]. It is common to use classifier-based models in natural language processing to detect fake news and misinformation. However, in related research, an approach named DetectGPT was proposed based on the idea that language models like GPT generate text that falls into specific areas of the model's probability distribution where the function has negative curvature or local maxima of the log probability. Their proposed model is used for detecting language model-generated text in scientific writing [14]. Other models like DetectGPT are known as traditional approaches to detect AI-generated text as they are based on statistical metrics such as entropy, perplexity, and n-gram frequency [13]. Another research aimed to develop a tool based on text mining techniques that could identify fake scientific papers generated by SCIgen. The tool used a combination of statistical and linguistic analysis to identify patterns in the language and structure of the papers that are characteristic of machine-generated text [15]. There is another related work in which they focused on the features and writing style of the text, and trained their framework on the abstract of the papers [6]. Such methods that work based on extracting specific patterns are called watermarking. A watermark is a hidden pattern in a text invisible to humans, but algorithmically detectable [16]. In pioneering studies, the possibilities of watermarks in language were conducted by manipulating syntax trees [17] [16]. In addition, using soft watermarking, a text is partitioned into tokens and a statistical test will be performed that can detect watermarks with a corresponding p-value that indicates the confidence level of detection [18].

There are also many online tools for AI-content detectors such as Giant Language Model Test Room [19], and Writer.com's AI Content Detector [20], but we will focus on scientific papers in our project as we will train our model on scientific text.

OpenAI classifier is another famous tool in this context as it is particularly designed to detect text generated by ChatGPT, but as they mentioned on their website, it still has some limitations. For example, it has the limitation of entering a minimum number of 1,000 characters and making wrong predictions [9].

Tools like Turnitin also exist that can identify potential plagiarism, which is already in use by some scientific journals. Turnitin provides many useful tools for educators for detecting plagiarism, submitting assignments, and providing online grades and feedback. Recently, it added an AI check tool as well.

However, the challenge of distinguishing AI-generated text from the human-generated text in scientific writing remains a significant concern, and further research is necessary to develop effective tools and strategies for addressing this issue

Therefore, it is essential to strike a balance between the benefits of AI technology in scientific research and the need to maintain transparency and credibility in the scientific publication process. As AI continues to advance, it is crucial to stay vigilant and adapt to new challenges to ensure the integrity of scientific writing and authorship.

In this paper, we present a cloud-based tool that can classify the text as AI-generated or human-generated using a multilayer neural network that can be helpful to address the above concerns.

III. AI-Generated Text Classifier

The proposed model in this paper is capable of computing the probability percentage of an input text being written by AI by taking advantage of text feature extraction tools and a machine learning classifier. A schema of the web app is shown in Fig. 2 The app consists of a user interface, the AI text detector tool, a feedback survey, and a database connected to it.

We developed the model using the scikit-learn library and trained it on a dataset that we created based on papers available on Google Scholar. We added key paragraphs and sections from various research papers of different areas to a CSV file and classified them manually as human-written. Then, we gathered similar text using ChatGPT and marked it as AI-generate. With this dataset, we trained our model to learn the differences between AI-generated text and human-written text.

A. *Data Collection*

We collected data manually from abstracts, discussions, conclusions, or future works of more than 300 scientific papers on Google Scholar in various areas such as computer science, mechanical engineering, environmental sciences, biology, medicine, chemistry, etc. We tried to select paragraphs and sections which are more informative about the whole paper and its key findings of or the paragraphs that include the preliminary definitions. Then we asked ChatGPT to write similar content using the same keywords and the title of the papers and added them to the dataset. Therefore, each instance on our dataset is either a text from a scientific paper or ChatGPT. Eventually, our dataset had 1200 instances labelled as human for the text extracted from the papers and AI for the text given by ChatGPT in a CSV file. We considered "0" for human-written text and "1" for AI-generated text.

B. *Architecture*

The input to the AI detector tool is text. This input will be processed by a count vectorizer which converts the input text to vectors or tokens. Then, the AI detector will take these tokens as input and uses a multilayer neural network to classify them as AI-generated or human-generated. The model calculates the probability of the text being AI-generated and returns an approximate percentage, which is displayed on the user interface. All these queries will be saved in the database.

Furthermore, the app includes a survey form for the users to provide feedback about their experience with the app and the results they get from the AI detector. The app keeps the history

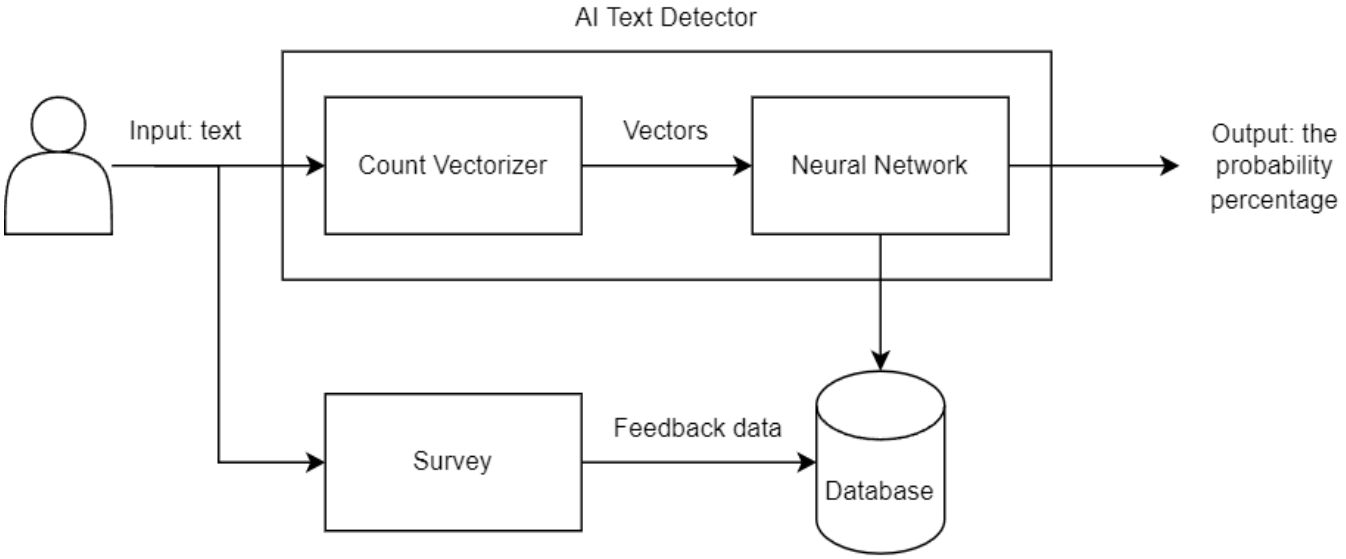

Fig. 2. Architecture of the proposed framework

of each user including their queries and their feedback in a PostgreSQL database.

C. Implementation

A proof of concept prototype has been constructed to classify text in case of being written by AI. According to previous works, the structure of a text written by AI is different from a text written by humans; also, to enable a machine learning algorithm to process text, it needs to be converted to fixed-length numerical vectors. Therefore, a count vectorizer was used that converts text to vectors. This tokenization approach generates an encoded vector consisting of the length of the entire vocabulary with the frequency of each word within the text [21]. In this paper, the count vectorizer was trained on 50 vectors or features for each instance in the dataset. The code and the dataset are available at https://github.com/Pariasrz/A-framework-for-detecting-AI-generated-text.

Then, an artificial neural network was trained on the dataset that we collected. An artificial neural network is a machine learning algorithm that attempts to simulate information processing in the human brain. This algorithm can be represented as a connected graph, where each node in the network has an activation function that processes the input data and sends a signal to other nodes. During training, the neural network learns to assign classes to unlabelled samples based on their features. This is achieved by adjusting the weights between the nodes in the network, which represent the strength of the connections between the nodes. The neural network is initially trained on a set of random weights, and after each iteration, the weights are updated to improve the network's ability to solve the problem [22]. This update is performed by an activation function.

The advantage of using an artificial neural network is that it can learn complex patterns and relationships in the data that may be difficult to identify using traditional methods. Additionally, neural networks can handle large datasets with high dimensionality, making them well-suited for tasks like text classification [23]. In this paper, a multilayer perceptron neural network was used that consists of 3 layers with 100 neurons in each layer, and the output of each layer is given to the next layer as input. Multilayer perceptron neural networks are nonlinear models with one or multiple inputs, along with hidden layers that connect these inputs to one or more outputs in a nonlinear manner [24]. As mentioned earlier, this model is designed to classify text into two categories, whether it is generated by AI or written by humans.

After training the model, it was integrated into our code, and a user interface (UI) was developed for it. Furthermore, a survey form was added in which the users are asked to answer some basic questions related to their experience with using our app. Once the user submits the survey form, they will receive an email confirming that their feedback was received and stored in our database. This data will be used to retrain the model to improve the accuracy and user experience of the app. The app has been deployed on AWS EC2, to take advantage of the scalability and reliability of Amazon's cloud infrastructure. This allowed us to ensure that the app could handle a large number of users and remain available even in the face of unexpected spikes in traffic. Additionally, by hosting the app on EC2, we were able to provide users with a fast and responsive experience, as well as the peace of mind that their data was being stored securely and reliably.

Overall, the proposed detection tool enables users to check if text is written by human or an AI app, and helps them avoid plagiarism in scientific publishing.

D. Challenges

Developing an accurate machine learning model for detecting AI-generated text versus human-written text was one of the most complex tasks, and it required a significant amount of data cleaning and pre-processing to ensure that the model is properly trained and performs well. To tackle this we manually created a dataset with 1200 instances in it.

Integrating the machine learning model into the app's backend and ensuring that it runs smoothly and efficiently also posed some challenges, especially if there was a large amount of data to process. Developing a user-friendly and intuitive UI that allows users to easily input text and view the results of the AI detector also required careful design and testing.

We successfully addressed the challenge of maintaining and securing the database by implementing proper security measures and regular backups to prevent data loss or unauthorized access. As the app grew and more users were added to the database, we ensured efficient database performance and scalability by regularly monitoring and optimizing the database. These measures helped ensure a seamless user experience and protected user data from potential security threats.

With all the challenges resolved and a robust system in place, the app is now able to provide users with a secure and efficient platform for detecting AI-generated text and checking for typos. The integration of the survey form has also allowed for valuable user feedback, which has been used to continuously improve the app's functionality and user experience. The database is now well-maintained and secured, ensuring the safety of user information and survey responses. Overall, the app is now an effective tool for detecting AI-generated text and improving the accuracy and readability of the content generated by the user.

IV. EVALUATION

One of the most challenging aspects was evaluating the accuracy of our model against existing market standards, so we compared it with the OpenAI classifier and ZeroGPT. We aimed to achieve higher accuracy levels than the benchmarks set by these tools. Despite the limitation of having a smaller dataset to train our model compared to industrial-level training, we firmly believe that our prototype has the potential to be further developed and scaled up for industrial usage.

As mentioned in previous sections, Turnitin has recently added a new tool to detect AI-generated content. They provide this tool for educators due to the concerns raised by using AI technologies in writing. However, they imply that their model's false positive rate is not zero, so professors and educators should also apply their knowledge when judging a written text. Turnitin provides the percentage of the text which is AI-generated; however, as regular users, we did not have access to this tool. Additionally, the writer.com tool has also limited the number of characters of the input text to 1500 characters for regular users.

Finally, to evaluate the performance of our text classification model, we conducted a comparison between our model and other existing solutions including OpenAI text classifier and ZeroGPT according to the classification accuracy score. As we used 20% of the dataset to test our model, we used the same portion of the dataset for the comparison as well. The results are displayed in Table I.

We used our dataset as input text for the OpenAI classifier, which returns results in the form of five different phrases indicating the likelihood of the text being AI-generated. We considered the phrases "very unlikely" and "unlikely" to indicate human-generated text, while "possibly," "very likely," and were considered AI-generated. OpenAI classifier shows the result for some texts as "unclear", and they did not make it clear what exactly leads to this result. It also has the limitation of entering at least 1000 characters otherwise it will not show any result, and it will ask the user to enter more text.

The accuracy of OpenAI text classifier was calculated to be 42.08%, with 40% of the results being unclear. In some cases it answers with "unclear", and in some cases, it did not give a result because of the number of characters. While adding more sentences can be suitable for the detectors to improve their detection performance, it is still important to consider that sections such as abstracts in many papers are not more than 1000 characters. By omitting this limitation, our model with an accuracy of 89.09% demonstrated significantly better performance as compared to the OpenAI classifier.

This evaluation process provides valuable insights into the effectiveness of our model in identifying AI-generated text. The results indicate that our model is highly accurate and can differentiate between human-generated and AI-generated text with greater accuracy than the OpenAI classifier.

To evaluate our model, we followed a similar pattern and compared its performance to the ZeroGPT AI detector tool, which uses perplexity score as a metric for detecting AI-generated text. The higher the perplexity score, the less likely it is that the text be detected as human-generated by ZeroGPT. This tool considers any perplexity score above a certain threshold as AI-generated and below the threshold as human-generated. ZeroGPT tool achieved an accuracy of 87.5%.

In addition, our model showed similar false positive and false negative rates compared to the ZeroGPT tool. Our model demonstrated competitive performance in accurately detecting AI-generated text, with similar accuracy. This suggests that our model still needs a lot of improvement and also needs to get trained over a huge amount of dataset. These findings are significant for the development of more reliable and accurate text classification models, which can be used for various applications such as detecting fake news, identifying spam messages, evaluating the reliability of scientific writing, and improving the overall quality of natural language processing.

TABLE I.
COMPARISON BETWEEN AVAILABLE TOOLS

Model	Type of Result	Accuracy
OpenAI Classifier	Nominal with 40% unclear results	42.08%
ZeroGPT	Nominal with the accuracy reported	87.5%
AI Text Detector	Numerical with the probability reported	89.95%

V. CONCLUSION AND FUTURE WORK

The proposed framework and cloud-based AI text detector prototype are designed to tackle the increasing problem of AI-generated content being used for malicious purposes. With a focus on scientific writing, our model is unique in the market and provides a much-needed solution for detecting plagiarism and manipulation in the scientific community.

The limitations of the proposed model can be summarized as (1) having not enough amount of data to train the model, (2) using a simple text pre-processing method and (3) not support for detecting AI-generated figures and tables. To address these limitations, for future work, we plan to expand the dataset and explore different pre-processing techniques, and continue monitoring the performance of our model to make possible and necessary updates. This will help us to ensure that the app stays up to date with the evolving landscape of AI-generated content. This also includes keeping up with advancements in GPT models and other AI technologies especially AI assistive tools for writing that may affect the model's ability to accurately detect AI-generated text.

We also plan to explore the possibility of integrating our model with other existing tools and platforms to provide a more comprehensive solution for detecting and combating AI-generated text. For example, our model could be integrated into plagiarism detection software to identify instances of AI-generated text being used to produce academic papers or articles.

Another potential avenue for future work is to expand the scope of our model beyond scientific writing to other genres, such as news articles or social media posts. This would require collecting and labelling a diverse dataset of texts and training the model to recognize the unique features of each genre.

REFERENCES

[1] R. Rombach, A. Blattmann, D. Lorenz, P. Esser and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

[2] "Much to discuss in AI ethics," *Nature Machine Intelligence*, vol. 4, no. 12, pp. 1055-1056, 2022.

[3] S. Sun, W. Zhao, V. Manjunatha, R. Jain, V. Morariu, F. Dernoncourt, B. V. Srinivasan and M. Iyyer, "IGA: An intent-guided authoring assistant," *arXiv preprint arXiv:2104.07000*, 2021.

[4] Y. Ma, J. Liu, F. Yi, Q. Cheng, Y. Huang, W. Lu and X. Liu, "AI vs. Human - Differentiation Analysis of Scientific Content Generation," 2023.

[5] N. Anderson, D. Belavy, S. Perle, S. Hendricks, L. Hespanhol, E. Verhagen and A. Memon, "AI did not write this manuscript, or did it? Can we trick the AI text detector into generated texts? The potential future of ChatGPT and AI in Sports & Exercise Medicine manuscript generation.," *BMJ Open Sport & Exercise Medicine*, vol. 9, no. 1, 2023.

[6] Y. Ma, J. Liu and F. Yi, "Is this abstract generated by ai? a research for the gap between ai-generated scientific text and human-written scientific text," *arXiv preprint*, 2023.

[7] R. Zellers, A. Holtzman, H. Rashkin, Y. Bisk, A. Farhadi, F. Roesner and Y. Choi, "Defending against neural fake news," *Advances in neural information processing systems*, vol. 32, 2019.

[8] "GPT-4, ChatGPT & AI Detector by ZeroGPT: detect OpenAI text," [Online]. Available: https://www.zerogpt.com/.

[9] "AI classifier for indicating AI-written text," OpenAI, [Online]. Available: https://openai.com/blog/new-ai-classifier-for-indicating-ai-written-text/. [Accessed 14 February 2023].

[10] "Turnitin," [Online]. Available: https://www.turnitin.com/.

[11] V. S. Sadasivan, A. Kumar, S. Balasubramanian, W. Wang and S. Feizi, "Can AI-Generated Text be Reliably Detected?," *arXiv preprint arXiv:2303.11156*, 2023.

[12] K. Krishna, Y. Song, M. Karpinska, J. Wieting and M. Iyyer, "Paraphrasing evades detectors of ai-generated text, but retrieval is an effective defense," *arXiv preprint arXiv:2303.13408*, 2023.

[13] S. Chakraborty, A. S. Bedi, S. Zhu, B. An, D. Manocha and F. Huang, "On the possibilities of ai-generated text detection," *arXiv preprint arXiv:2304.04736*, 2023.

[14] E. Mitchell, Y. Lee, A. Khazatsky, C. D. Manning and C. Finn, "DetectGPT: Zero-Shot Machine-Generated Text Detection using Probability Curvature," *arXiv preprint arXiv:2301.11305*, 2023.

[15] C. Labbé and D. Labbé, "Duplicate and fake publications in the scientific literature: how many SCIgen papers in computer science?," *Scientometrics*, vol. 94, pp. 379-396, 2013.

[16] M. J. Atallah, V. Raskin, M. Crogan, C. Hempelmann, F. Kerschbaum, D. Mohamed and S. Naik, "Natural language watermarking: Design, analysis, and a proof-of-concept implementation," in *Information Hiding: 4th International Workshop, IH 2001 , April 25--27, 2001 Proceedings 4*, 2001.

[17] H. M. S. B. Meral, A. S. Özsoy, T. Güngör and E. Sevinç, "Natural language watermarking via morphosyntactic alterations," *Computer Speech & Language*, vol. 23, no. 1, pp. 107-125, 2009.

[18] J. Kirchenbauer, J. Geiping, Y. Wen, J. Katz, I. Miers and T. Goldstein, "A watermark for large language models," *arXiv preprint arXiv:2301.10226*, 2023.

[19] "Catching Unicorns with GLTR," [Online]. Available: http://gltr.io/. [Accessed 14 February 2023].

[20] "AI content detector," [Online]. Available: https://writer.com/ai-content-detector/. [Accessed 14 February 2023].

[21] K. Poddar and K. S. Umadevi, "Comparison of various machine learning models for accurate detection of fake news,"

Innovations in Power and Advanced Computing Technologies (i-PACT), vol. 1, pp. 1-5, 2019.

[22] S. Uddin, A. Khan, M. Hossain and M. A. Moni, "Comparing different supervised machine learning algorithms for disease prediction," *BMC Medical Informatics and Decision Making,* vol. 19, p. 281, 2019.

[23] M. Khashei and M. Bijari, "An artificial neural network (p, d, q) model for timeseries forecasting," *Expert Systems with applications,* vol. 1, pp. 479-489, 2010.

[24] A. N. Kia, M. Fathian and M. R. Gholamian, "Using MLP and RBF neural networks to improve the prediction of exchange rate time series with ARIMA," *International Journal of Information and Electronics Engineering,* vol. 2, no. 4, pp. 543-546, 2012.