AI Meets AI: Artificial Intelligence and Academic Integrity

A Survey on Mitigating AI-Assisted Cheating in Computing Education

Ying Xie
Department of Information Technology, Kennesaw State University
yxie2@kennesaw.edu

Shaoen Wu
Department of Information Technology, Kennesaw State University
swu10@kennesaw.edu

Sumit Chakravarty
Department of Electrical and Computer Engineering, Kennesaw State University
schakra2@kennesaw.edu

ABSTRACT

This paper discusses pressing issues in the area where Artificial Intelligence (AI) meets Academic Integrity (AI). It starts by outlining the potential consequences of AI-assisted cheating, including the risks posed to education quality, fairness, and the credibility of academic institutions. After reviewing an array of strategies reported in the literature to counteract such cheating, this paper calls for rigorous research to assess the effectiveness of those strategies. It further suggests a range of research topics in detecting AI-generated content and highlights a promising research direction focusing on motivating students' interest in learning through innovative AI applications that divert their efforts away from misuse of technology. Lastly, the paper suggests that addressing AI cheating requires ethical education, academia-industry collaboration, integration into AI ethics, and an international consortium. One of the unique contributions of this paper is outlining a range of potential research directions, both technical and non-technical, in this area where AI meets AI.

CCS CONCEPTS

• **Applied computing** → Education; Interactive learning environments; Education; E-learning.

KEYWORDS

Artificial Intelligence, Academic Integrity, Cheating, Computing Education

ACM Reference Format:
Ying Xie, Shaoen Wu, and Sumit Chakravarty. 2023. AI Meets AI: Artificial Intelligence and Academic Integrity: A Survey on Mitigating AI-Assisted Cheating in Computing Education. In *The 24th Annual Conference on Information Technology Education (SIGITE '23), October 11–14, 2023, Marietta, GA, USA.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3585059.3611449

1 INTRODUCTION

One of the most significant advancements in AI that impact computer education directly is large language models that can produce coherent texts about virtually anything. Technologies like ChatGPT [1], powered by GPT-3.5/GPT-4, as well as tools like Google's Bard [2], and Meta's LLaMA [3], can generate text outputs based on a few words or sentences as input. These technologies, typically called generative AI, have versatile applications, such as generating realistic essays and code snippets. It is tempting for students to exploit generative AI models to cheat, especially on programming assignments. Therefore, it is crucial to develop methods to detect and prevent AI-generated work [5–7].

AI-assisted cheating can have detrimental consequences, including:

1). Degrading the quality of education: When students rely on AI to generate full code solutions by simply inputting prompts, rather than understanding programming concepts, syntax, and debugging themselves, they miss out on valuable learning opportunities. Lacking critical thinking and problem-solving practice will cause students to fail to develop a genuine understanding of the subject matter. Consequently, they may struggle to apply learned concepts to new problems [8, 9].

2). Creating an unfair advantage: Students who employ AI-assisted cheating gain an unfair advantage over their peers who do not use such tools. By using AI, they can complete assignments quickly and nearly effortlessly. This unfair advantage creates an imbalanced learning environment, which may demoralize students who adhere to academic integrity [10–12]. Such an imbalanced learning environment could erode the core principles of equity and honesty that education strives to nurture.

3). Damaging the integrity of academic institutions: Cheating with AI tools undermines the honesty and authenticity of the grades that students earn. If grades do not accurately reflect students' skills and knowledge, concerns could be raised about the credibility of the assessment process within academic institutions [10]. Such erosion of integrity not only damages the reputation of the institution but also hinders students' prospects of securing employment or gaining admission to graduate schools [13].

As AI generative models rapidly evolve, their abilities expand quickly. New tools like OpenAI's Code Interpreter [4] are capable of not only generating code but also automatically debugging and executing code in a cloud environment and directly presenting the final output. Students who are completely reliant on this type of tool for programming assignments will completely miss out on every stage of programming practice while still earning high grades. Therefore, cheating with AI tools is a serious trending issue that needs to be addressed promptly and proactively.

SIGITE '23, October 11–14, 2023, Marietta, GA, USA

ACM ISBN 979-8-4007-0130-6/23/10.
https://doi.org/10.1145/3585059.3611449

2 SOLUTIONS SUGGESTED BY LITERATURE

Many people have created strategies to tackle the problem of AI-assisted cheating. These solutions include:

1). Explanations and Justifications: Students should be able to explain and justify their code. This strategy assumes that students who can provide comprehensive explanations and justifications are more likely to have a true understanding of their code [14, 15]. On the contrary, it raises concerns about potential cheating.

2). Complex, Open-Ended Problems: Assignments should be designed to be complex and open-ended, allowing for multiple possible solutions. As of right now, AI tools are largely unable to generate working solutions to complex and ambiguous problems. This strategy also engages students to exercise cognitive heavy lifting by employing critical thinking skills and developing creative approaches to problem-solving [16, 17, 21, 23].

3). Oral Examinations: Oral exams can be employed to assess students' understanding of code since oral exams are more difficult to cheat on. During an oral exam, students are required to explain and defend their code without the aid of notes or other materials. This makes it harder for students to rely on AI-generated solutions, as they must demonstrate a genuine understanding of the code [18, 19, 23].

4). Group Projects: Collaborative group projects can act as a deterrent to AI-assisted cheating. By assigning group projects, students are encouraged to work together and share their code with one another. This collaborative environment makes it more challenging for individual students to rely solely on AI-generated solutions [20–22].

5). Adapting Assignment Formats: Assignments can be modified to make them more resistant to AI-assisted cheating by emphasizing skills that AI cannot replicate. For instance, an assignment that asks students to first understand a given code and then optimize it as much as possible requires higher-level thinking beyond basic code generation. [23].

6). Promoting Conceptual Understanding: Students should be encouraged to develop a conceptual understanding of computing. If students are taught to deeply comprehend coding theory and principles, rather than primarily focusing on code proficiency, they gain the capability to solve novel problems beyond an AI tool's capabilities. Therefore, prioritizing conceptual understanding helps counter students' propensity to utilize AI shortcuts, while preparing them for complex coding challenges where AI assistance falls short [24].

7). Promoting Originality and Creativity: Students should be original and creative in their coding. For example, students could be required to code to solve a problem that they have a passion for. If a student is working on a project they genuinely want to do, they are less likely to cheat. Also, by emphasizing originality over prefabricated solutions, educators can empower students to reach their creative potential while minimizing the effectiveness of AI plagiarism tools [25].

8). Educating Students about AI Tools: Students should be educated about the capabilities and limitations of AI tools. By providing information on how AI tools work and discussing the ethical implications of using them for cheating, educators can raise awareness and discourage misuse [26–28, 36].

9). Citing AI Assistance: Students should be required to cite AI assistance when they use it. This strategy promotes transparency regarding the use of AI technologies. Students who utilize AI tools to generate code should be obliged to cite the specific tools they used and explain how they employed them. This not only helps prevent code plagiarism but also enables tracking of AI tool usage within the classroom [29].

10). Use of Lockdown Browsers: The use of lockdown browsers that prevent students from accessing other websites or tools during an online exam can be an effective way to reduce AI-assisted cheating during exams [22, 30].

11). Education on Consequences: Schools can teach students about the negative consequences of AI-assisted cheating, including the impact it can have on their academic records, future career prospects, and personal integrity [22].

13). Clear Academic Integrity Policies: Schools should establish clear academic integrity policies and consequences for violating those policies. These policies should be communicated to students and enforced consistently [31].

Even though these strategies have been suggested and discussed in writing, it is important to conduct research to assess how well they work and if they are practical in real academic environments.

3 UTILIZING AI TO DETECT AI-GENERATED TEXT

OpenAI has launched a tool, the OpenAI AI Text Classifier [32], that attempts to distinguish between human-written and AI-generated text. The classifier isn't particularly accurate, with a success rate of around 26%. However, when used together with other methods, it could be helpful in preventing AI text generators from being abused. The OpenAI AI Text Classifier was trained on text from 34 text-generating systems from five different organizations, including OpenAI itself. The training data was paired with similar human-written text from sources such as Wikipedia, websites extracted from links shared on Reddit, and a set of "human demonstrations" collected for a previous OpenAI text-generating system. However, this tool is far from mature. For instance, it requires a minimum of 1,000 characters, or about 150 to 250 words, to work on any text, and it doesn't detect plagiarism. It also may misclassify texts written in a language other than English, due to its English-centric dataset.

Besides OpenAI's efforts, other tools have emerged to detect AI-generated text. For example, GPTZero, developed by a Princeton University student, uses criteria including "perplexity" (the complexity of text) and "burstiness" (the variations of sentences) to detect whether text might be AI-written [33]. Plagiarism detector Turnitin is also developing its own AI-generated text detector [34].

Most current tools for detecting AI-generated text employ features based on the vocabulary, grammar, and style<strike>s</strike> of sentences. Our research team is currently working on an interesting project that performs the detection by looking at logic flows within full paragraphs or even entire papers. A recent paper [46] proposed a novel method to embed watermarks in text generated by large language models with negligible impact on the quality of generated text. However, as the detecting techniques improve, so

will the text-generating AI, thus creating a cat-and-mouse game like the one between cyber criminals and security researchers.

4 RESEARCH OPPORTUNITIES ON DETECTING AI-GENERATED CODE

Because there are currently no tools that exist specifically for identifying AI-generated code, creating such tools is an important ongoing endeavor.

One potential research topic is to identify and extract effective indicators of AI-generated code. Those indicators could include unusual conciseness, efficiency, or repetitiveness compared to human-written code. Code with significantly fewer lines or more optimized logic than expected could signal AI generation. Another sign might be if the code strongly resembles previously published code, indicating plagiarism from an AI model trained on existing codebases. However, code obfuscation techniques could be used by students to partially hide AI-written code's fingerprints; therefore, research is needed to develop more sophisticated and accurate techniques for indicator extraction.

Certain existing AI/Machine Learning applications may be repurposed for this purpose. For instance, [41] uses temporal analysis to model time-dependent changes in university students' online assignment submission behavior. By using a similar strategy, an AI/Machine Learning system trained on sequences of students' code submissions could flag substantial deviations in code style as potential indicators of AI authorship. Repeated similarities in code across a cohort could also suggest the use of AI.

Reliably watermarking AI-generated code is another interesting research topic to pursue. Embedding subtle fingerprints to text parts of programming code using similar strategies described in [46] could be a starting point to explore. These text components may include variables, comments, logging, invisible outputs, indents, and non-functional codes. Alternative strategies could leverage code structures, control flows, and or compilation/execution artifacts for watermarking.

AI-based approaches for detecting AI-assisted coding could be ingeniously integrated with blockchain technology, whose secure and tamper-proof attributes could facilitate an unalterable record of student code. This would prevent any post-submission alterations. However, integrating blockchain with existing Learning Management Systems (LMS) [35] requires extensive development.

Another interesting research idea is to use quantum computing for code verification. Quantum computers can process exponentially greater amounts of data compared to traditional computers by leveraging quantum mechanical phenomena. This massive processing capability could be employed to analyze extensive codebases and detect patterns indicating AI generation. For example, quantum algorithms could rapidly analyze code syntax, structure, efficiency, comments, and other markers on a large scale to flag probable AI-generated submissions. Quantum machine learning models could also be developed to uncover subtle indicators of AI authorship. Furthermore, quantum cryptography could be applied to authenticate code submissions. However, quantum computing is still in its early stages, with quantum volumes insufficient for the applications mentioned above. It also requires significant, and currently unrealistic, hardware and software expansions in education settings. While promising in theory, utilizing quantum advances for combating AI-developed code is still years from practical feasibility.

5 MAKING AI PART OF THE SOLUTION, NOT THE PROBLEM

As a matter of fact, AI itself has the potential to enhance, rather than hinder, academic integrity. One positive way in which AI can contribute is by creating assignments tailored specifically to the skill levels of each student. By using course goals and individual capabilities as prompts, AI may generate assignments that are suited to meet each student's learning needs. Personalized learning material could make students feel more comfortable and confident in learning thus less motivated to cheat. Furthermore, AI could be trained to produce engaging assignments that spark students' curiosity. Such assignments encourage students to put in their best work and effort. Therefore, when utilized strategically, AI has the capacity to strengthen integrity instead of compromising it.

Research topics along this line include, but are not limited to, 1) using AI to automatically identify students' learning styles and track students' learning progress; 2) automatically accessing students' learning effectiveness and dynamically adjusting learning materials; 3) training AI to producing engaging assignments; and 4) evaluating the impact of personalized learning assisted by AI in different learning environments.

6 FURTHER CONSIDERATION ON MITIGATING AI-ASSISTED CHEATING

The issue of AI-assisted cheating extends beyond a technical challenge - it is also a question of ethics and education. Simply developing better detection tools is not enough. In this section, we outline several promising areas for educators and researchers to consider.

1). Curriculum Design on Ethics Training. As mentioned in section 2, to address AI-assisted cheating at its root, curricula on ethics training need to be developed for students at all levels. There are multiple options to incorporate ethics training in computing education, including but not limited to, a) developing independent courses; b) integrating ethics training into existing computing courses in the forms of lectures, projects, and/or case studies; c) offering seminars with guest speakers from both academia and industry; and d) hosting hackathons or competitions that provide students opportunities to detect AI-generated content. The effectiveness and feasibility of each of these strategies need to be studied with the outcome being disseminated.

2). Collaboration Between Academia and Industry. Academia needs to partner with industry to address AI-assisted cheating together. Below are some possible collaboration scenarios.

Tech companies, who have already worked to combat harmful AI-generated content [42, 43], could partner with academic researchers to repurpose their advancements for academic usage.

AI industry leaders, such as OpenAI and Google, are developing tools to watermark AI-generated text [42, 47]. Once these technologies become mature, watermark detection services could be offered to schools for reliable detection of AI-generated content.

Developing platforms offered by some tech companies, such as GitHub [44]. and IBM [45], provide real-time feedback and guidance

on coding to software developers. These techniques can be adapted to provide guidance to students on their coding exercises.

3). Multiple journals and workshops are dedicated to research on AI ethics [36–39]. In [48], the author called for a new sub-discipline for AI ethics and proposed an integral framework for studying this discipline. When considering AI-related academic integrity, it is beneficial to tie it with the entirety of AI ethics. In this way, we can use frameworks, principles, and best practices proposed in AI ethics to address issues in AI-related academic integrity. When we place academic integrity on the whole landscape of AI ethics, it becomes clear that not just students and educators, but all stakeholders in the AI ecosystem take responsibility for academic integrity. These include all those developing, distributing, and using AI academically. Therefore, research on AI-related academic integrity should take a systematic method to study how to shape appropriate development, deployment, and usage of AI technology in education, instead of simply taking fragmented approaches to prevent AI cheating.

4). To facilitate collaboration among all stakeholders in the AI ecosystem to combat issues of AI-related academic integrity, an international consortium is needed to provide a global venue for exchanging ideas, sharing R&D outcomes, and disseminating best practices. Given its long-term efforts and commitments to computing education, the ACM SIGITE [40] could take the initiative in creating such a consortium and take the leadership role in promoting academic integrity in the age of AI.

7 CONCLUSION

As artificial intelligence continues to evolve, the issue of AI-assisted cheating in the educational realm has come to the forefront. In this paper, we intend to provide a comprehensive picture of this emerging area where academic integrity (AI) meets artificial intelligence (AI). We started by outlining the harmful consequences of AI-assisted cheating and emphasized the imperativeness of addressing these issues. Then we surveyed mitigation strategies proposed in the literature and called for rigorous research to evaluate the effectiveness of those strategies. We further proposed multiple research directions for detecting both AI-generated text and AI-generated code. Besides using AI for detection, a more positive application is to use AI for creating personalized, curiosity-sparking learning environments that guide students away from misusing AI. This type of application presents a valuable research prospect for both technological and educational researchers.

Given that the challenge of AI-assisted cheating goes beyond technical solutions, we also presented research and collaboration opportunities in multiple non-technique areas, including 1) ethical education, 2) collaboration between academia and industry, and 3) the formation of an international consortium to tackle this challenge. We especially emphasized the necessity for a systematic approach to studying academic integrity with AI by integrating it into AI ethics. By working collaboratively, we believe a learning environment can be established in the age of AI, which will prioritize academic integrity, foster original thinking, and facilitate true subject mastery.

REFERENCES

[1] ChatGPT. https://chat.openai.com.
[2] Bard. https://bard.google.com.
[3] Introducing Llama 2. https://ai.meta.com/llama.
[4] How to Use ChatGPT Code Interpreter. https://www.datacamp.com/tutorial/how-to-use-chat-gpt-code-interpreter.
[5] Ben T. 2023. ChatGPT. Artificial Intelligence, and Academic Integrity. https://oai.missouri.edu/chatgpt-artificial-intelligence-and-academic-integrity.
[6] The impact of ChatGPT on Academic Integrity. 2023. https://www.enago.com/thesis-editing/blog/the-impact-of-chatgpt-on-academic-integrity.
[7] Debby C., Peter C. & J. Reuben S. 2023. Chatting and cheating: Ensuring academic integrity in the era of ChatGPT. (2023). Innovations in Education and Teaching International. DOI: 10.1080/14703297.2023.2190148
[8] Justin G. 2023. How AI Will Permanently Disrupt the Education Industry. Gold Penguin, goldpenguin.org/blog/how-ai-will-permanently-disrupt-the-education-industry/.
[9] Manjeet R. & Dan Y. 2023. The Impact of Artificial Intelligence and ChatGPT on Education. Newsroom, St. Thomas University, news.stthomas.edu/the-impact-of-artificial-intelligence-and-chatgpt-on-education/.
[10] AI-Powered Cheating: A New Challenge for Schools and Educators 2023. AI Time Journal (https://www.aitimejournal.com/ai-powered-cheating-a-new-challenge-for-schools-and-educators/43575/)
[11] AKI P. 2022. AI is making it easier than ever for students to cheat by Slate. https://slate.com/technology/2022/09/ai-students-writing-cheating-sudowrite.html
[12] Christine L. What is the potential of AI writing? Is cheating its greatest purpose? https://www.turnitin.com/blog/what-is-the-potential-of-ai-writing-is-cheating-its-greatest-purpose
[13] The reputational effects of academic dishonesty in higher education 2023. https://www.meazurelearning.com/resources/the-reputational-effects-of-academic-dishonesty-in-higher-education
[14] The importance of justification in student learning 2022. https://www.gogreenva.org/the-importance-of-justification-in-student-learning/
[15] Why students need to explain their reasoning 2018. https://www.carnegielearning.com/blog/why-students-need-to-explain-their-reasoning/
[16] Scott B. & Greg. S. 2018. Factors influencing student success on open-ended design problems, International Journal of Technology and Design Education, 28(4), DOI:10.1007/s10798-017-9415-2
[17] Michael B. 2022. What Is Creative Problem-Solving & Why Is It Important? https://online.hbs.edu/blog/post/what-is-creative-problem-solving
[18] Allison T. 2021. Oral Exams: A More Meaningful Assessment of Students' Understanding, Journal of Statistics and Data Science Education, DOI:10.1080/26939169.2021.1914527
[19] Della D. 2020. Revitalizing Classes Through Oral Exams, https://www.insidehighered.com/advice/2020/09/09/how-use-oral-examinations-revitalize-online-classes-opinion
[20] What are the benefits of group work? - Eberly Center - Carnegie Mellon University (cmu.edu)
[21] Avoiding AI-based cheating. https://ecas.engin.umich.edu/avoiding-ai-based-cheating/
[22] Jdmughal 2023. 10 effective ways schools can prevent AI-assisted cheating. https://technologyia.com/10-effective-ways-schools-can-prevent-ai-assisted-cheating/
[23] Beatrice N. 2023. College professors are considering creative ways to stop students from using AI to cheat. https://www.businessinsider.com/ai-chatgpt-college-professors-students-cheating-2023-1
[24] Jewoong M., Daeyeoul L. & Gi C. 2020. A conceptual framework for teaching computational thinking in personalized OERs, Smart Learning Environments, 7(6), DOI:10.1186/s40561-019-0108-z
[25] Jonathan B. 2018. Originality in Coding. https://www.turnitin.com/blog/originality-in-coding
[26] Selin A. & Christine G. 2022. Artificial intelligence in education: Addressing ethical challenges in K-12 settings, AI and Ethics, 2, DOI:10.1007/s43681-021-00096-7
[27] Glenn K. 2023. Educators need to understand and embrace artificial intelligence writing tools. https://edsource.org/2023/educators-need-to-understand-and-embrace-artificial-intelligence-writing-tools/685299
[28] Olufemi S. 2023. AI in the classroom: pros, cons and the role of EdTech companies. https://www.forbes.com/sites/theyec/2023/02/21/ai-in-the-classroom-pros-cons-and-the-role-of-edtech-companies/?sh$=$23d2283efeb4
[29] How to Cite Artificial Intelligence https://libguides.ccsu.edu/c.php?g$=$736245&p$=$9555042
[30] LockDown Browser – Prevent cheating during online exams. https://web.respondus.com/k12/lockdownbrowser/
[31] Classroom Policies for AI Generative Tools, https://docs.google.com/document/d/1RMVwzjc1o0Mi8Blw_-JUTcXv02b2WRH86vw7mi16W3U/edit?pli$=$1
[32] New AI classifier for indicating AI-written text. https://openai.com/blog/new-ai-classifier-for-indicating-ai-written-text
[33] Emma B. 2023. A college student made an app to detect AI-written text, https://www.npr.org/2023/01/09/1147549845/gptzero-ai-chatgpt-edward-tian-plagiarism
[34] AI Writing Detection https://www.turnitin.com/solutions/ai-writing

[35] David E. & Ramesh N. 2020. Blockchain and LMS: a proof of concept. https://elearningindustry.com/blockchain-and-lms-a-proof-of-concept
[36] AI and Ethics https://www.springer.com/journal/43681/
[37] The AI Ethics Journal https://www.aiethicsjournal.org/
[38] The 3rd Workshop on Artificial Intelligence and Ethics (AI & Ethics) https://ifipaiai.org/2023/workshops/aiethics/
[39] The 1st International Workshop on Responsible AI and Data Ethics (RAIDE 2022) https://sites.google.com/view/raide2022
[40] ACM SIGITE https://www.sigite.org/
[41] Mehmet K., Gökhan A. & Mohammad N. H. 2021. Unfolding students' online assignment submission behavioral patterns using temporal learning analytics. Educational Technology & Society, 24 (1), 223-2354.
[42] Ashley B., OpenAI, Google will watermark AI-generated content to hinder deepfakes, misinfo. https://arstechnica.com/ai/2023/07/openai-google-will-watermark-ai-generated-content-to-hinder-deepfakes-misinfo
[43] Discover 5 Top Startups Tackling Deepfakes, https://www.startus-insights.com/innovators-guide/5-top-startups-tackling-deepfakes
[44] Your AI Pair Programmer, https://github.com/features/copilot
[45] IBM Watsonx Code Assistant, https://www.ibm.com/products/watsonx-code-assistant
[46] John K., Jonas, G., Yuxin W., Jonathan K., Ian M., and Tom G. 2023. A Watermark for Large Language Models. arXiv:2301.10226 cs.LG. https://doi.org/10.48550/arXiv.2301.10226
[47] Kyle W. 2022. OpenAI's attempts to watermark AI text hit limits. https://techcrunch.com/2022/12/10/openais-attempts-to-watermark-ai-text-hit-limits
[48] Jonathan B. 2022. Ethical Analytics: A framework for a practically-oriented sub-discipline of AI Ethics. Dissertation. https://digitalcommons.kennesaw.edu/dataphd_etd/15