WIP: Beyond Code: Evaluating ChatGPT, Gemini, Claude, and Meta AI as AI tutors in Computer Science and Engineering Education

Sagnik Nath
Dept. of Computer Science and Engineering
University of California, Santa Cruz
Santa Cruz, U.S.A.
https://orcid.org/0000-0002-5225-952X

So Yoon Yoon
Dept. of Engineering and Computing Education
University of Cincinnati
Cincinnati, U.S.A.
https://orcid.org/0000-0003-1868-1054

***Abstract*—This Work-in-Progress research paper evaluates the validity of Large Language Models (LLMs) as conversational AI tutors for computer science learning. While current engineering education literature has predominantly emphasized the rapid evolution of LLMs as conversational AI tutors for programming languages, the exploration into their effectiveness within general STEM topics remains comparatively scarce. This WIP study thus centers on evaluating the potential of LLMs to facilitate understanding of core hardware design concepts critical to computer science and engineering (CSE) education. By cross-checking the responses from generative AI chatbots to an open-ended CSE-based question, we aimed to uncover how LLMs, such as ChatGPT-3.5, Claude, Gemini, and Meta AI, can contribute to teaching and learning of general CSE courses instead of a specifically coding-based one. Our method involved simulating a student query on the popular debate between CISC vs. RISC related to computer architecture and analyzing the chatbots' responses. This initial collection of data served as the foundation for a continual comparative analysis aimed at determining the inherent instructional value of each LLM and its validity and reliability. To systematically assess the responses, we introduced an evaluation framework focusing on metrics, such as response accuracy, persuasiveness, and depth of explanation. The current work anticipates not only enriching our understanding of how these advanced LLMs can support general CSE education but also identifying areas where further development is needed for a more holistic integration of LLM-based chatbots in assisting student comprehension in the overarching engineering education.**

Keywords—Large Language Models (LLMs), computer engineering education, AI tutors, CISC, RISC, computer architecture, Instruction set architecture

I. INTRODUCTION

Since its launch in November 2022, OpenAI's flagship product ChatGPT [1] has marked the beginning of a new era in general information queries. It proved sufficiently disruptive to prompt immediate media speculation [2] upon release regarding its potential to surpass established platforms like Google in terms of user preference for obtaining information. Soon after, the utility of this chatbot, leveraging the generative pre-trained transformer (GPT) based large language model (LLM) [3] for natural language processing to produce human-like responses, expanded into the educational sector, finding a large variety of potential use cases in creating customized lesson plans, providing AI-based tutoring and mentorship, and developing AI-powered educational platforms, among many others [4], [5], [6]. The most immediately applicable use case of a conversational chatbot like ChatGPT in the context of computer science and engineering (CSE) education became assisting students in the context of coding, both in code writing based on problem definition and code explanation and debugging of faulty student code [7]. Concurrent with the writing of this paper, a comprehensive textbook [8] has already been published detailing the use of ChatGPT and GitHub CoPilot [9] as tools for code generation and debugging.

However, while LLM-based chatbots have proven very effective in carrying out human-like conversations, the accuracy of their responses, especially in an educational context, cannot be empirically guaranteed given the fundamental nature of their model architecture [5]. A general approach for determining the overall reliability of an LLM has been to use standardized benchmarks, such as MMLU [10] or HUMAN EVAL [11], which contain manually constructed test cases from a range of topics across STEM, the humanities, and the social sciences. Nonetheless, these tests frequently fail to capture all possible scenarios because crafting high-quality tests is laborious. As a result, these benchmarks do not adequately assess the true correctness of LLM-generated responses, which can lead to unwarranted confidence in the results [12].

From a student's perspective of using LLM chatbots to learn coding, there at least exists the straightforward approach of running the chatbot-generated code in the appropriate High-Level Language (HLL) runtime environment to directly observe inaccuracies or failures in handling various test cases. However, verifying the correctness of LLM responses to open-ended questions from STEM fields, such as computer architecture, proves more challenging. Additionally, as students debug these potentially misleading answers, the task is

complicated by the chatbots' highly human-like conversation style, which can misleadingly affirm the accuracy of their responses, based on the faulty premise that a more persuasive dialogue necessarily indicates more correctness [13]. Moreover, the quality of responses from various publicly available chatbots can vary significantly, affecting students' understanding of concepts based on key parameters: accuracy, persuasiveness, and depth of explanation.

This WIP study seeks to bridge the gap between broad assessments of LLM efficacy based on standardized benchmarks and the nuanced experiences of students who rely on chatbots as AI tutors in computer science and engineering education. Subsequent sections outline the relevant background and prior research, detail the research questions addressed, describe the evaluation method employed, and present the findings. Additionally, we discussed potential avenues for future research, aiming to further elucidate the role of LLMs in enhancing students' grasp of core engineering concepts.

II. Theoretical Frameworks

Although extensive literature exists on optimizing coding education with ChatGPT and other code-centric AI tools like GitHub Copilot, there is less research on the comparative use of freely available chatbots such as Claude [14], Gemini [15], Meta AI [16], alongside ChatGPT-3.5. Most studies have compared these chatbots' responses for accuracy using standardized benchmarks [17]. On the other hand, the available literature directed toward a more personalized evaluation of an AI tutor has mainly centered around solely ChatGPT's reliability, and the research has been more towards fine-tuning the prompts to deliver reliable responses from ChatGPT [13]. However, there is a notable absence of research aimed at developing an evaluation framework to assess various LLM chatbots, aside from ChatGPT, from the students' perspective of using them as AI tutors to address *open-ended* STEM questions, that is, questions that do not have a straightforward right or wrong answer but *may need more context*.

To address the above divide, we propose to create a set of metrics for evaluating specific open-ended questions to be posed to various chatbots and see their respective performance in terms of accuracy, persuasiveness, and depth of explanation. We based these three metrics on the three theoretical frameworks of Constructivist Learning Theory [18], the Elaboration Likelihood Model (ELM) of persuasion [19], and Cognitive load theory [20].

We accordingly posited and aimed to address the following research question:

- *How do LLMs, such as ChatGPT, Claude, Gemini, and Meta AI, differ from each other in their ability to effectively explain specific complex engineering concepts in terms of accuracy, persuasiveness, and depth of explanation?*

III. Method

In the preliminary phase of our study, we presented an open-ended question related to computer architecture to the LLM chatbots, such as ChatGPT-3.5, Claude, Gemini, and Meta AI. This subject area was selected based on the first authors' expertise in teaching a computer architecture course within the Computer Science and Engineering Department at a research university.

These particular LLM-powered chatbots for our study were chosen for their ease of access as each is freely available online and does not require local installation or complex setup procedures. This easy accessibility is essential for replicating typical student interactions and ensuring that our findings are relevant to a broader educational context and generalizable across diverse educational settings.

A. The Open-Ended Question

All four LLM chatbots were given the same question: "*how is cisc better than risc."* Here, "cisc" refers to Complex Instruction Set Computer (CISC) and "risc" refers to Reduced Instruction Set Computer (RISC) [21]. These terms represent two distinct philosophies in processor architecture design, primarily based on the diversity of unique instructions the machine code can handle. Historically, most commercial architectures, like Intel's x86, were CISC. This approach was adopted to manage limited and slow memory and to compensate for less sophisticated compilers, despite necessitating complex hardware design and multiple clock cycles per instruction. In contrast, RISC gained traction in the 1980s and 1990s, favored for its streamlined hardware design that reduced the number of clock cycles per instruction and benefited from advancements in compiler technology. In undergraduate computer architecture courses, RISC is often preferred over CISC due to its more straightforward and student-friendly hardware design [22].

Misunderstandings can arise among students about which design philosophy, CISC or RISC, is "better." For instance, the continued prevalence of the x86 architecture, a common CISC design in PCs and laptops might suggest to some that CISC is superior. Conversely, the increasing adoption of the RISC V architecture [23] could imply the superiority of RISC principles. However, labeling an architecture simply as CISC or RISC doesn't fully address its superiority in performance or power efficiency. Factors, such as instruction count, clock speed, and cycles per instruction, must be considered as well to accurately compare specific CISC and RISC implementations. Thus, the debate provides an excellent opportunity to evaluate how LLM chatbots explain and contextualize these fundamental concepts in computer architecture, aiding in students' comprehensive understanding.

B. Evaluation Metrics

We employed three primary metrics to evaluate the chatbot responses to our research questions: accuracy, persuasiveness, and depth of explanation. Accuracy and depth of explanation were chosen based on Constructivist Learning Theory and Cognitive Load Theory. These theories highlight the importance of accurate and detailed interactions between students and their AI tutors, which significantly influence the student learning process.

Persuasiveness, based on the Elaboration Likelihood Model (ELM) and particularly its Peripheral Route, is specifically assessed for factually inaccurate responses. As

discussed in [13], LLMs like ChatGPT can be particularly eloquent, using persuasive language that may erroneously convince students of the accuracy of their responses. Therefore, it is crucial to measure how persuasive inaccurate responses can be to better understand and mitigate their potential to mislead students.

Only the initial responses to the open-ended question from four chatbots were archived to apply the evaluation matrix. The responses, saved for Gemini's, have been archived online through A.I. Archives [25]. A.I. Archives does not support archiving of Gemini responses; therefore, we used Google's provided option to save the chat.

C. Evaluation Criteria

Accuracy is quantified by the ratio of incorrect facts to total factual statements, with each response rigorously cross-verified against credible sources [22] [24]. Persuasiveness is assessed *only for inaccurate responses* using a Likert scale from 1 (not persuasive) to 5 (highly persuasive), focusing on the effectiveness of the language, logical structure, and any rhetorical devices used in erroneously convincing the evaluator of the response's validity. Depth of explanation is rated on a scale from 1 (superficial) to 5 (comprehensive), based on a rubric that evaluates the complexity and clarity of the information provided.

IV. Preliminary Results

A. ChatGPT-3.5 from OpenAI

The conversation has been archived in [26].

1) Accuracy

Out of a total of ten statements evaluated, only one was found to be incorrect. The statement in question was: "*Hardware Resources Utilization: CISC instructions may utilize hardware more efficiently by incorporating microcoded sequences to execute complex operations, which can reduce the amount of hardware required for certain tasks compared to RISC architectures.*" This assertion is somewhat misleading. While it is true that many CISC *implementations* utilize microcode to simplify the execution of complex instructions, this does not necessarily mean they require less hardware compared to RISC architectures. On the contrary, the control logic in CISC systems is often more complex due to the need to handle these intricate instructions, typically making the hardware more elaborate than in RISC implementations. Therefore, the accuracy rate was 90%

2) Persuasiveness

For the incorrect response, the persuasiveness was deemed as 1 (not persuasive). Given that students are usually introduced to the concept of microcoding in their coursework, they are likely to recognize the inconsistency between the response provided and the standard descriptions found in textbooks. [24].

3) Depth of Explanation

This was deemed as 3. Barring the single erroneous response, the overall response gave sufficient context as to the nuances in comparing CISC with RISC.

B. Claude from Anthropic

The conversation has been archived in [27].

1) Accuracy

Out of the nine statements reviewed, two were identified as incorrect. The first erroneous statement pertained to Claude's views on "Hardware Simplicity", which erroneously stated "*the hardware needed to decode and execute them can be simpler than the hardware required for RISC architectures*". This was similar to those previously noted in ChatGPT-3.5. The second inaccurate statement was: "*Single-cycle execution: Some CISC instructions can be executed in a single cycle, potentially improving performance for certain tasks compared to RISC processors, which may require multiple cycles to accomplish the same task*." Contrary to this statement, it is typically RISC architectures that can execute instructions in a single cycle, whereas CISC systems generally require at least two to three clock cycles per instruction. As a result, the overall accuracy rate for the evaluated statements stands at 77.78%.

2) Persuasiveness

For the incorrect responses, a rating of 2 (Slightly persuasive) was assigned. Although Claude's statement about 'Hardware Simplicity' may have been relatively easy to identify as misleading, akin to the response from ChatGPT-3.5, the assertion regarding single-cycle CISC instructions was delivered in an articulate manner that appeared to convey some sense of accuracy and confidence.

3) Depth of Explanation

This was deemed as 4. The overall response gave sufficient context as to the nuances in comparing CISC with RISC. Besides listing the specific potential advantages of a CISC implementation over RISC, it offers greater context on the advantages of RISC as well.

C. Gemini from Google

The conversation has been archived in [28].

1) Accuracy

Out of the seven statements reviewed, two were identified as incorrect. The first error concerned the generalization that *"RISC tends to dominate the modern landscape*." While it is true that RISC V has been receiving significant media attention lately due to its open-source nature [23], CISC architectures continue to maintain a substantial presence. It is possible that Claude's assertion of RISC's dominance was influenced by the recent surge in media coverage about RISC V. The second misleading statement claimed that CISC is "*Potentially faster for legacy code*." The explanation provided did not support this claim. Since CISC-specific code must run on CISC architectures and cannot be executed on RISC architectures, the statement lacks meaningful context. As a

result, the overall accuracy rate for the evaluated statements stands at 71.42%.

2) Persuasiveness

For the incorrect responses, a rating of 1 (not persuasive) was assigned. Particularly, Gemini's rationale for CISC being "*Potentially faster for legacy code*" would immediately inform the student of the lack of reliability from Gemini on the overall context of the response.

3) Depth of Explanation

Gemini's response provided only a short summary, lacking substantial depth. From a student's perspective, this limited explanation fails to capture the complexities involved in the debate between CISC and RISC architectures. Consequently, a rating of 1 was assigned.

D. Meta AI from Meta

1) Accuracy

Out of a total of 7 statements evaluated, one was found to be incorrect: *"Faster execution: Complex instructions can be executed in a single clock cycle, reducing the number of cycles needed to complete a task."* In general, it is possible for a specific CISC implementation to be faster than a RISC implementation for running a specific benchmark based on various factors, but the reasoning provided by Meta AI here is false, as has been indicated in Claude's incorrect response as well. As a result, the overall accuracy rate for the evaluated statements stands at 85.71%.

2) Persuasiveness

For the incorrect responses, a rating of 2 (Slightly persuasive) was assigned. The response presentation was deemed structured as opposed to Gemini which could lead to the misleading assumption from students that the response was correct.

3) Depth of Explanation

Overall, Meta AI provided a more nuanced explanation of the open-ended question compared to Gemini. However, even the correct responses were presented with abrupt rationales that might increase the cognitive load for students. Consequently, a rating of 2 was assigned.

V. Discussion

The results using the evaluation metrics of the accuracy persuasiveness, and depth of explanation of the responses to the open-ended question in the context of CSE were summarized in Table I below.

TABLE I. Evaluation of Four LLM Chatbots

AI Tutor	Accuracy	Persuasiveness for Incorrect Responses	Overall Depth of Explanation
Chat GPT-3.5	90.00%	1	3
Claude	77.78%	2	4
Gemini	71.42%	1	1
Meta AI	85.71%	2	2

It is critical to recognize that the outcomes of this study depend heavily on the specific open-ended question posed to the AI tutors. While ChatGPT-3.5 demonstrated superior performance compared to other freely available chatbots, it should not be regarded as the default AI tutor for STEM courses. The subjective nature of the Likert scale ratings for persuasiveness and depth of explanation by the evaluator means that these metrics would benefit from further validation in our future work. We aim to include a more extensive cataloging of multiple student interactions with AI tutors and ensure adequate sample sizes for robust statistical analyses.

Our planned future research would also aim to develop rigorous evaluation metrics and the application of statistical methods using multiple open-ended questions. These approaches will help determine validity and reliability as well as the generalizability of the findings, such as their applicability across different educational contexts.

Ultimately, our goal is to establish a system of rubrics and evaluation criteria that STEM instructors can use to assess various chatbots. This system would differ from the standardized benchmark approach or prompt-tuning approaches by allowing instructors to use more generalized, student-provided prompts. This approach ensures that our rubric system remains flexible, applicable to any chatbot, and adaptable to updates or newer versions of the same models. By evaluating AI tutors based on a standardized rubric, instructors can more effectively determine which chatbot is most suitable as a supplementary tutor for their specific curriculum, thereby enhancing the educational value provided to students.

We anticipate that the framework developed from this study could be generalized to address any open-ended STEM questions besides Computer Science and Engineering ones posed to these AI models in their capacity as AI tutors.

References

[1] "Introducing ChatGPT," [Online]. Available: https://openai.com/index/chatgpt/. [Accessed 12 May 2024].

[2] J. Carpenter, "OpenAI's ChatGPT could start a search engine revolution. Should Google be worried?," [Online]. Available: https://fortune.com/2022/12/05/openai-chatgpt-chatbot-ai-artificial-intelligence-google-alphabet. [Accessed 12 May 2024].

[3] L. Floridi and M. Chiriatti, "GPT-3: Its Nature, Scope, Limits, and Consequences," *Minds & Machines,* vol. 30, p. 681–694, 2020.

[4] X. Zhai, "ChatGPT User Experience: Implications for Education," [Online]. Available: https://dx.doi.org/10.2139/ssrn.4312418. [Accessed 12 May 2024].

[5] E. Kasneci, K. Sessler, S. Küchemann, M. Bannert, D. Dementieva, F. Fischer and U. Gasser, "ChatGPT for good? On opportunities and challenges of large language models for education," *Learning and Individual Differences,* vol. 103, no. 102274, 2023.

[6] L. CK, "What Is the Impact of ChatGPT on Education? A Rapid Review of the Literature," *Education Sciences,* vol. 13, no. 4, p. 410, 2023.

[7] O. Eng Lieh , B. K. S. Gan and S. Jin , "ChatGPT, Can You Generate Solutions for my Coding Exercises? An Evaluation on its Effectiveness in an undergraduate Java Programming Course.," in *ITiCSE 2023*, 2023.

[8] L. Porter, Learn AI-assisted Python Programming: With GitHub Copilot and ChatGPT., Simon and Schuster, 2024.

[9] "GitHub CoPilot," [Online]. Available: https://github.com/features/copilot. [Accessed 13 May 2024].

[10] "MMLU (Massive Multitask Language Understanding)," [Online]. Available: https://paperswithcode.com/dataset/mmlu. [Accessed 13 May 2024].

[11] "HumanEval," 13 May 2024. [Online]. Available: https://paperswithcode.com/dataset/humaneval.

[12] J. Liu, . C. S. Xia, Y. Wang and L. Zhang, "Is Your Code Generated by ChatGPT Really Correct? Rigorous Evaluation of Large Language Models," [Online]. Available: arXiv:2305.01210v3.

[13] G. Polverini and B. Gregorcic, "How understanding large language models can inform the use of ChatGPT in physics education," *European Journal of Physics,* vol. 45, no. 2, 2024.

[14] "Claude AI," [Online]. Available: https://claude.ai/chats. [Accessed 14 May 2024].

[15] "Gemini AI," [Online]. Available: https://gemini.google.com/app. [Accessed 14 May 2024].

[16] "MetaAI," [Online]. Available: https://ai.meta.com/meta-ai/. [Accessed 14 May 2024].

[17] A. Borji and M. Mohammadian, "Battle of the Wordsmiths: Comparing ChatGPT, GPT-4, Claude, and Bard," 2024. [Online]. Available: https://ssrn.com/abstract=4476855.

[18] B. Steve Olusegun and S. Olusegun, "Constructivism learning theory: A paradigm for teaching and learning," *Journal of Research & Method in Education,* vol. 5, no. 6, pp. 66-70, 2015.

[19] P. Richard E. and J. T. Cacioppo, The elaboration likelihood model of persuasion, New York: Springer , 1986.

[20] J. Sweller, "Cognitive load theory," *Psychology of learning and motivation,* vol. 55, pp. 37-76, 2011.

[21] A. D. George, "An overview of RISC vs. CISC," in *Proceedings The Twenty-Second Southeastern Symposium on System Theory*, 1990.

[22] D. A. Patterson and J. L. Hennessy, Computer Organization and Design RISC-V Edition: The Hardware Software Interface, Morgan Kaufmann Publishers Inc., 2017.

[23] S. Chen, "These simple design rules could turn the chip industry on its head," MIT Technology Review. [Online]. [Accessed 15 May 2024].

[24] J. L. Hennessy and D. A. Patterson, Computer architecture: a quantitative approach, Elsevier, 2011.

[25] "AI Archives," [Online]. Available: https://aiarchives.org/. [Accessed 10 May 2024].

[26] "ChatGPT-3.5 response," [Online]. Available: https://aiarchives.org/id/Yvd24tqdXb1wgV5VjPQN. [Accessed May 15 2024].

[27] "Claude reponse," [Online]. Available: https://aiarchives.org/id/yoRaT1D8JusvIhReiPtR. [Accessed 15 May 2024].

[28] "Gemini response," [Online]. Available: https://g.co/gemini/share/f7cc04b889b8. [Accessed 15 May 2024].