Can AI models generate free response questions for AI courses?

Yutong Ai
ellenai@umich.edu

Mingye Chen
mingyech@umich.edu

Melina O'Dell
melodell@umich.edu

Fanghao Zhang
fanghaoz@umich.edu

1 INTRODUCTION

Free response questions (FRQs) are the most common question type for assessments in computer science courses [10]. Presently, course staff manually creates and reviews these types of questions before presenting them to students to ensure sufficient testing of course content, as well as clarity, conciseness, and correctness of the question. However, this thorough process is very time-consuming and tedious for course staff, making it a candidate for semi- or complete automation through software tools.

With the explosion in popularity of using large language models (LLMs) for text generation, using generative AI for automated question writing is being explored. In a recent study, researchers have found success using AI to generate reading comprehension quizzes [8]. Other studies have explored using AI to generate multiple-choice questions (MCQs) and coding exercises for programming courses [3][9], but writing FRQs has some unique challenges. First, the complexity and scope of learning objectives covered by one FRQ is much larger than for multiple-choice questions. FRQs are typically longer, open-ended, and written to broadly cover course content. Second, writing engineering questions is different from writing reading comprehension questions, as they need to test student knowledge of more complex and obscure content in meaningful ways, sometimes involving logic. Instructors of engineering courses may be able to take advantage of the power of LLMs to write free response questions for assessments.

This study investigates if generative AI models can be used to generate free response questions for quizzes and exams in an graduate-level Artificial Intelligence (AI) course. We will compare several different text generation models across different course topics. To evaluate performance, we will survey student participants to rate the questions, as well as evaluate the responses ourselves using several metrics for quantifying question quality. Our results can provide insights for engineering instructors interested in using AI to automate FRQ writing for course quizzes and exams, without sacrificing the quality that comes from traditional question writing practices.

Specifically, our research questions are:

- **RQ1**: Can AI write high-quality free response questions for AI courses?
- **RQ2**: Can generative AI produce correct answer keys for free response questions that it wrote?
- **RQ3**: Do certain generative AI models produce "better" free response questions than others?

We found that generative AI can generate high-quality FRQs for an AI course and produce correct answers to most of its own questions. Because different models behaved differently for various question topics and prompts, course staff should choose models based on their strengths and review the questions before deploying. We found that the models were less proficient with creating and answering logical and ethical questions, corroborating existing observations [4][1].

2 RELATED WORK

A recent study proposed to generate exams using AI instead of human writers [7]. It reviews automatic question generation (AQG) literature from 2015 to early 2019, emphasizing the need for continuous question supply, cost reduction, and advancements like adaptive testing. Although it found an increased interest and great potential in AQG, it focuses less on generating questions of controlled difficulty, enriching question forms and structures, automating template construction, improving presentation, and generating feedback. It states that existing AQG generative models are trained to operate on the syntax-tree and semantic relations of text, or populates templates of fixed text. It also suggests potential weaknesses, such as limited question forms, content, and structures, and lacks discussion on question difficulty. The use of LLMs were not mentioned in this study, so employing them may improve on these weaknesses, as they are known for creative text generation. Inspired by this research, we plan to explore generating diverse questions using LLMs with varying types and amounts of provided context. We will also survey students to get a measure of difficulty for our questions.

While previous work suggests exploring various question forms, we discovered that there are many existing studies that focus only on MCQs. A 2023 study discusses the usability and quality of the AI-generated MCQs for medical graduate exams [2]. Given the substantial workload of university instructors, this study assesses the quality of MCQs produced by ChatGPT for use in graduate medical examinations, compared to those written by instructors based on standard medical textbooks, using a rubric across five categories, rating on a scale of 1 to 10 by the authors. While it concludes that the AI tool has great potential to efficiently generate comparable-quality MCQs for medical exams, it lacks student testing and comparison with human-written questions. We plan to design a rubric of evaluation metrics to rate our questions and recruit current students to provide valuable feedback.

Another 2024 study looks at the computer science field, discovering the potential of AI for generating programming questions [3]. It compares the AI-generated MCQs with human-written MCQs for beginner courses at the undergraduate level and suggests that LLMs can automatically generate high-quality MCQs for high-level programming courses. The questions were generated based the learning objectives and course information, and evaluates the students on multiple areas, such as code output and error analysis. It highlights the feasibility of automated MCQ generation, indicating that it has the potential to reduce the time educators spend

on developing assessments. However, this research is limited to undergraduate Python courses only, and the question type in their research is limited to one type of question (MCQs). These limitations are similarly present on another 2023 and 2024 study on the same CS question generation topic [11] [6]. Our research acknowledges these potential weaknesses. We aim to experiment with different models to generate FRQs instead of MCQs, and for a graduate-level AI course with more open-ended question types.

While existing research focuses on generating the questions themselves, there is a lack of focus on generating the answer keys alongside the exam questions [3] [2] [7] [11] [6]. Answer keys play a critical role in the grading process, especially for FRQs, where student responses can greatly vary in both length and detail. Developing comprehensive answer keys could significantly enhance the efficiency and consistency of grading, ensuring a fair and standardized assessment process for all students. Automating the creation of detailed answer keys could streamline the grading workflow and reduce the potential for bias, increasing the speed of grading while maintaining high assessment standards.

Similarly, existing research does not investigate whether AI-generated questions can be answered using readily-available AI tools [3] [2] [7] [11] [6]. As students have easy access to generative AI tools, concerns of academic integrity may arise if questions are not resistant to these tools. Studies have shown that generative AI tools can perform reasonably well at answering exam and quiz questions [5]. We will research ways to use LLMs to generate robust questions, and as part of our evaluation, check if the questions can be correctly answered by a popular AI tool. This will help in ensuring that the questions are designed to truly assess students' understanding and skills, minimizing the risk of AI-assisted cheating.

3 METHODOLOGY AND EVALUATION

To address our research questions, we studied 2 state-of-the-art LLMs, GPT-3.5 and Llama 2.0, and 2 open source LLMs, Zephyr-7b-beta[1] and Xwin-LM-7B-V0.1[2]. We chose GPT-3.5 and Llama 2.0 because of their reputation for excellence in natural language processing tasks across various domains. These models are large (i.e. trained on many parameters) and are commonly used in industry. We chose Zephyr-7b-beta and Xwin-LM-7B-V0.1 because of their accessibility and benchmark scores[3] (MT-Bench and AlpacaEval). These models are smaller (7B parameters each), which may affect their ability to produce diverse and detailed FRQs. Overall, LLMs are proficient with text generation and replication, so we believe that these models can be used to generate FRQs.

The primary objective of our study was to generate FRQs tailored for AI courses, with the intention of their potential use in quizzes or exams. The course covers a broad spectrum of topics, ranging from fundamental search algorithms to discussions of ethics in AI. Typically, this course enrolls about 80 students.

We constructed two distinct prompts within the scope of topics covered in the AI curriculum. The first prompt served as a concise, context-free sentence, while the second prompt incorporated additional context and complexity. Supplementary materials (authentic quiz questions and relevant content extracted from lectures), were provided with the complex prompt. We chose to focus on 6 varying course topics: "AI Environments", "Ethics", "Search Algorithms", "Logic," "Constraint Satisfaction Problems (CSP)," and "Reinforcement Learning (RL)." In the course in our study, students struggled the most with CSP problems, so automated generation of questions could be useful for course staff to produce extra practice problems. To study the randomness of LLM text generation, each prompt was subjected to 5 iterations per question topic for each model. We found that using more than 5 iterations did not yield significantly more unique results. To answer RQ2, we also used the model to generate answer keys in the same session where the question was created. This approach was adopted to enhance the reliability of our results and minimizing variance across trials.

Across 4 models, 2 prompts, 6 question topics, and 5 iterations, we generated 240 questions. To analyze each of them, we developed a rubric of evaluation metrics:

- Did the question have any grammatical errors?
- Did the question test learning objectives from the course?
- Did the model generate a correct answer key?
- Is the answer easily found on Google?
- Is it easily answered using a state-of-the-art generative AI chatbot?
- Did the model listen to our prompt and generate a question?
- Did the model generate an FRQ (not another question type)?
- Did the question have a subjective or objective answer?
- How many subquestions did it generate?

We defined "testing learning objectives" as if the question referenced and tested content that actually appeared in the course materials or on previous assignments.

Notably, we combined and focused on two pairs of these metrics to quantify question "correctness" and "quality." "Correctness" entails if a state-of-the-art chatbot (ChatGPT) can produce a correct answer, and if the answer key originally generated by the model is correct. These metrics are important for instructors to automate generating high-quality, robust questions on short notice. "Quality" combines if a question tests a learning objective for the course, and if the answer was found using traditional search engines within one page of results. These metrics are important in a learning environment, as instructors want to test relevant material in a way that requires students to utilize knowledge from the course, rather than copying the answer. We believe a high percentage of learning objectives implies that the course staff will be more likely to use the question, and a low percentage of searchability implies a question is more creative.

We also analyzed the randomness of the 5 repeated outputs of the models in our study by calculating a similarity score between the FRQs they generated. We calculated TF-IDF scores for each generated question, as it is a common metric for measuring the importance of words relative to each other in a dataset (in our case, the collection of responses), and used it to calculate the cosine similarity between each response. We averaged the similarity scores for the responses for each model, prompt, and each question topic. This allowed us to compare the creativity and reproduciblity of

[1]https://huggingface.co/HuggingFaceH4/zephyr-7b-beta
[2]https://huggingface.co/Xwin-LM/Xwin-LM-7B-V0.2
[3]https://huggingface.co/HuggingFaceH4/zephyr-7b-beta#performance

questions from each model given the same prompt. Models that produced similar repeated responses have higher similarity scores.

To gather student opinions of quality, we designed two surveys comparing question pairs generated by different models and varying levels of prompt complexity. We paired questions generated with the same prompt but across different models, and questions generated within the same model but with prompts of different complexity. We asked participants to rate each pair:

- Which question is clearer (easier) to understand?
- Which one feels more like a fair question that you would expect on a quiz or exam in this course?
- Do either of the questions seem too difficult (unanswerable), based on what you've learned in class?
- Overall, which question feels higher quality?
- Do they provide enough details for you to answer the question, if you had to?

The survey was distributed to students enrolled in the course, as their familiarity with the course expectations and learning objectives ensured relevant feedback. By leveraging their insights, we aimed to gauge the effectiveness and appropriateness of the generated questions for the intended educational context.

4 RESULTS AND ANALYSIS

For each of our research questions, we will provide results[4], explain how they were produced, and analyze them in the context of the research question.

We reviewed all 240 generated questions, using each model, prompt complexity, and question type, and evaluated them using 9 different metrics. We also selected representative pairs of questions to compare and measure the quality of using a survey and analyzed the similarity of model responses using TF-IDF scores.

4.1 RQ1: Can AI write high-quality questions?

Across both surveys, we collected 53 (24, then 29) student survey responses and analyzed each of the FRQ comparisons. 60.9% of respondents to the second survey also filled out the first survey. To draw conclusions, we looked for the majority vote per comparison[5].

First, we found that all of the questions were considered a fair difficulty based on what the students have experienced in class. Second, students reported that the Llama 2.0 "AI Environments" and "CSP" questions were clearest and highest quality, but for "Search Algorithm" questions, GPT 3.5 was the best, closely followed by Llama 2.0. Both state-of-the-art models performed better than the 2 open source models for all question topics. Overall, questions generated with complex contextual prompts produced questions that students felt were similar to actual quiz and exam questions from the course.

These results suggest that generative AI models are capable of producing free response quiz questions for an AI course, with a similar quality and feel to what course staff traditionally delivers. Students familiar with typical quiz questions rated the AI-generated questions highly. They preferred questions generated using complex prompts, as they were clear and concise with few ambiguities.

[4]See code used for prompting here (link), and spreadsheet with data analysis here (link).
[5]See survey response data here (link) and here (link).

4.2 RQ2: Can AI answer its own questions?

Using our definition of "correctness", we calculated averages for the ChatGPT-ability and correct answer key generation across models and prompts, and gathered the results in Table 1.

We found that all models were good at answering "AI Environments", "CSP", and "RL" questions and worst at "Logic" questions. Compared with those generated by Llama 2.0 and Xwin-LM-7B-V 0.2, questions generated by GPT 3.5 and Zephyr-7b-beta are the easiest to answer using AI tools.

These observations suggest that generative AI tools tend to work better for answering open-ended, subjective questions ("AI Environments") and well-known algorithm questions ("CSP", "RL"). Instructors should consider these differences in strengths before using LLMs for FRQ generation and when deciding between models.

Topic	GPT 3.5		Llama 2.0	
	No Context	**Context**	**No Context**	**Context**
AI Env.	1, 1	1, 1	1, 1	1, 1
Ethics	1, 1	1, 1	0.6, 0.8	0.8, 0.8
Search	1, 1	1, 1	0.8, 0.8	0.2, 0.2
Logic	0.2, 0	0.6, 0.2	0, 0	0, 0.4
CSP	1, 1	0.8, 0.8	1, 1	1, 1
RL	0.6, 1	1, 1	1, 1	1, 1
Avgs.	**0.8, 0.83**	**0.9, 0.83**	**0.73, 0.77**	**0.67, 0.73**
	0.85, 0.83		**0.7, 0.75**	
	Zephyr-7b-beta		**Xwin-LM-7B-V0.1**	
	No Context	**Context**	**No Context**	**Context**
AI Env.	1, 1	0.4, 0.4	1, 1	0, 0
Ethics	0, 0.8	0, 0.6	0.6, 0.8	0.6, 1
Search	0, 1	0, 0.6	1, 1	0.6, 0.6
Logic	0.4, 1	0.4, 1	0.2, 1	0, 0.4
CSP	0.6, 1	1, 1	1, 1	1, 1
RL	1, 1	1, 1	1, 1	1, 1
Avgs.	**0.5, 0.97**	**0.47, 0.77**	**0.8, 0.97**	**0.53, 0.67**
	0.48, 0.87		**0.88, 0.6**	

Table 1: AI correctness. The first number is the percent of correctly generated answer keys, and the second number is the percent of questions that a chatbot could correctly answer in a new chat. (Higher, Lower) is better.

4.3 RQ3: Model comparison

Using our definition of "quality", we calculated averages for the searchability and learning objective coverage across models and prompts, and gathered the results in Table 2.

Overall, Llama 2.0 generated more questions that covered relevant learning objectives (78%). However, the GPT 3.5 questions improved significantly when provided with question examples, jumping from 67% to 77% of questions covering learning objectives. Xwin-LM-7B-V 0.2 generated the fewest questions that covered learning objectives. Llama 2.0 and Zephyr-7b-beta were best at generating questions that were not searchable, demonstrating high levels of creativity. GPT 3.5 generated the most searchable questions. Most "Logic" questions produced by every model were very basic and easy to answer online. "RL" questions had the highest likelihood of covering course learning objectives, and "Ethics" questions were the hardest to find answers to online, all including unique ethical scenarios.

These results suggest that there is variation in FRQ quality between LLMs; however, there are common weaknesses. LLMs are not

proficient with logic and struggle to produce unique logic-related questions and correct answers. The differences in creativity and learning objective representation between the models questions are likely due to differing training data in the models. For example, the smaller open source models produced less creative and relevant questions than the larger models. Therefore, choosing an model for FRQ generation would require tradeoffs based on the course's context and pedagogical priorities.

Topic	GPT 3.5		Llama 2.0	
	No Context	Context	No Context	Context
AI Env.	0, 1	1, 0.6	0.4, 0.2	1, 0
Ethics	1, 0.8	0, 0	0.6, 0.6	0.2, 0
Search	1, 1	1, 0.8	0.2, 0	1, 1
Logic	1, 1	1, 1	1, 1	1, 0.8
CSP	0, 1	0.6, 0.8	1, 0	1, 1
RL	1, 1	1, 1	1, 1	1, 1
Avgs.	**0.67, 0.97**	**0.77, 0.7**	**0.7, 0.47**	**0.87, 0.63**
	0.72, 0.83		**0.78, 0.55**	
	Zephyr-7b-beta		**Xwin-LM-7B-V0.1**	
	No Context	**Context**	**No Context**	**Context**
AI Env.	0.4, 0.8	0.2, 0	0, 1	0, 0
Ethics	0, 0	0, 0.4	0.2, 0.8	0, 1
Search	1, 1	0.6, 0.6	0.8, 1	0.4, 0.6
Logic	1, 1	1, 1	1, 1	0.4, 0.4
CSP	0.6, 0	1, 0	0.8, 1	1, 0
RL	1, 0.6	1, 1	1, 0.6	1, 1
Avgs.	**0.67, 0.57**	**0.63, 0.5**	**0.63, 0.9**	**0.47, 0.5**
	0.65, 0.53		**0.55, 0.7**	

Table 2: FRQ quality metrics. The first number is the percent of questions that tested learning objectives from the course in our study, and the second number is the percent of questions that a search engine could give exact answers to within one page of search results. (Higher, Lower) is better.

After generating all 240 questions, we performed a similarity score calculation on questions from each model. These results are shown in Table 3.

Llama 2.0 had the highest similarity score overall, where Xwin-LM-7B-V0.1 had the lowest. Llama 2.0 also had the highest similarity score when using prompts that included additional context. Compared to the two state-of-the-art models, which had a similarity score increase when providing context (from 0.11 to 0.15 and 0.11 to 0.18), the open source models' similarity scores drop once context is provided (from 0.10 to 0.07 and 0.07 to 0.06). Overall, the open source models (0.09, 0.07) produced questions that were random and unique, compared to the state-of-the-art models (0.13, 0.14).

This suggests that the state-of-the-art models are better at reproducing similar FRQs than the smaller open source models. This is likely due to their larger training dataset. If course staff is prioritizing consistency in their question styles, they may want to utilize the state-of-the-art models over the others.

Model	Similarity Score		
	No Context	Context	Total
GPT 3.5	0.11	0.15	0.13
Llama 2.0	0.11	0.18	0.14
Zephyr-7b-beta	0.10	0.07	0.09
Xwin-LM-7B-V0.1	0.07	0.06	0.07

Table 3: Cosine similarity scores for each model.

5 CONCLUSION

We compared 4 different generative AI models, 2 state-of-the-art and 2 open source, for generating FRQs for an AI course. We used 2 different prompts, one including additional context (such as example questions and lecture content) and one without, and generated multiple iterations of questions for 6 different course topics. We evaluated the questions using a rubric of question quality metrics, a similarity score comparison, and surveys of students in the course.

We have found that generative AI is capable of generating high-quality FRQs for an AI course, based on student opinions. After our analysis of the questions and answer keys, it can produce correct answers to most of its own questions, but struggling with logic and ethics. Llama 2.0 performed the best overall for FRQ generation. The state-of-the-art models produced higher quality questions than the open source models. While all the models can produce questions that have a fair difficulty, each model behaved differently for different question topics and different prompts, sometimes producing highly creative questions that cover learning objectives, but sometimes producing unusable content. Course staff looking to use generative AI to automate question writing should choose models based on their strengths and review the questions before deploying them to students.

There are some ethical concerns to consider before using generative AI for FRQ generation. We found that most of the questions generated by AI could also be easily answered by the AI tools. This could lead to potential violations of academic integrity. We suggest to use generative AI with caution, and that any questions automatically generated should be reviewed by course staff before using on assignments.

Among the responses that were produced by the open source models, there were a notable number of empty, duplicate, or nonsense responses. For example, when asked to generate an "AI Environments" question, the response produced was "Create a step-by-step guide for making homemade dehydrated fruit chips." which is clearly irrelevant to the prompt. This made our response evaluation harder and more time consuming than we estimated, and we spent 2 days just evaluating the responses. We also had to repeat experiments more times than expected in order to produce enough valid responses. Because the models were slow to create responses, we spent another 2 days just generating questions.

However, we were pleased to discover that most (98%) of questions were grammatically correct, as this was an initial concern we had when using AI models to generate natural language. We also found that our student survey respondents were conscientious and detailed in their responses, enhancing our results. This helped us achieve our final research goals.

Future work in this area could include expanding our analysis to more generative AI models, such as Google Gemini or Claude 3, or other open source models. If automatically generating questions becomes more appealing to course staff, a software application could be developed to interface with the AI models and assist instructors with writing course assignments. If the logical reasoning and ethics of AI models improve, there is potential to use them for automatically grading assignments that they wrote, or generating rubrics with point values to further assist course staff.

REFERENCES

[1] Emily M. Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. 2021. On the Dangers of Stochastic Parrots: Can Language Models Be Too Big? . In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency* (Virtual Event, Canada) *(FAccT '21)*. Association for Computing Machinery, New York, NY, USA, 610–623. https://doi.org/10.1145/3442188.3445922

[2] Billy Ho Hung Cheung, Gary Kui Kai Lau, Gordon Tin Chun Wong, Elaine Yuen Phin Lee, Dhananjay Kulkarni, Choon Sheong Seow, Ruby Wong, and Michael Tiong-Hong Co. 2023. ChatGPT versus human in generating medical graduate exam multiple choice questions—A multinational prospective study (Hong Kong S.A.R., Singapore, Ireland, and the United Kingdom). *PLOS ONE* 18, 8 (08 2023), 1–12. https://doi.org/10.1371/journal.pone.0290691

[3] Jacob Doughty, Zipiao Wan, Anishka Bompelli, Jubahed Qayum, Taozhi Wang, Juran Zhang, Yujia Zheng, Aidan Doyle, Pragnya Sridhar, Arav Agarwal, Christopher Bogart, Eric Keylor, Can Kultur, Jaromir Savelka, and Majd Sakr. 2024. A Comparative Study of AI-Generated (GPT-4) and Human-crafted MCQs in Programming Education. In *Proceedings of the 26th Australasian Computing Education Conference (ACE 2024)*. ACM. https://doi.org/10.1145/3636243.3636256

[4] Jessica López Espejel, El Hassane Ettifouri, Mahaman Sanoussi Yahaya Alassan, El Mehdi Chouham, and Walid Dahhane. 2023. GPT-3.5, GPT-4, or BARD? Evaluating LLMs Reasoning Ability in Zero-Shot Setting and Performance Boosting Through Prompts. arXiv:2305.12477 [cs.CL]

[5] Natalie Kiesler, Dominic Lohr, and Hieke Keuning. 2023. Exploring the Potential of Large Language Models to Generate Formative Programming Feedback. arXiv:2309.00029 [cs.AI]

[6] Nischal Ashok Kumar and Andrew Lan. 2024. Using Large Language Models for Student-Code Guided Test Case Generation in Computer Science Education. arXiv:2402.07081 [cs.CL]

[7] Ghader Kurdi, Jared Leo, Bijan Parsia, Uli Sattler, and Salam Al-Emari. 2019. A Systematic Review of Automatic Question Generation for Educational Purposes. *International Journal of Artificial Intelligence in Education* 30 (11 2019). https://doi.org/10.1007/s40593-019-00186-y

[8] Xinyi Lu, Simin Fan, Jessica Houghton, Lu Wang, and Xu Wang. 2023. ReadingQuizMaker: A Human-NLP Collaborative System that Supports Instructors to Design High-Quality Reading Quiz Questions. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems* (<conf-loc>, <city>Hamburg</city>, <country>Germany</country>, </conf-loc>) *(CHI '23)*. Association for Computing Machinery, New York, NY, USA, Article 454, 18 pages. https://doi.org/10.1145/3544548.3580957

[9] Sami Sarsa, Paul Denny, Arto Hellas, and Juho Leinonen. 2022. Automatic Generation of Programming Exercises and Code Explanations Using Large Language Models. In *Proceedings of the 2022 ACM Conference on International Computing Education Research - Volume 1* (Lugano and Virtual Event, Switzerland) *(ICER '22)*. Association for Computing Machinery, New York, NY, USA, 27–43. https://doi.org/10.1145/3501385.3543957

[10] Judy Sheard, Simon, Angela Carbone, Donald Chinn, Mikko-Jussi Laakso, Tony Clear, Michael de Raadt, Daryl D'Souza, James Harland, Raymond Lister, Anne Philpott, and Geoff Warburton. 2011. Exploring programming assessment instruments: a classification scheme for examination questions. In *Proceedings of the Seventh International Workshop on Computing Education Research* (Providence, Rhode Island, USA) *(ICER '11)*. Association for Computing Machinery, New York, NY, USA, 33–38. https://doi.org/10.1145/2016911.2016920

[11] Nguyen Binh Duong TA, Hua Gia Phuc NGUYEN, and GOTTIPATI Swapna. 2023. ExGen: Ready-to-use exercise generation in introductory programming courses. (2023).