Building an Instructional Design–Backed, GPT-Driven AI Tutor for Math Homework Support

Josh Fisher, Husni Almoubayyed, Stephen E. Fancsali, Steve Ritter, Logan De Ley, Zack Lee

Abstract

We describe the latest iteration of a digital chatbot to support students using a widely-deployed print and digital work-text for mathematics, building on a rule-based, static chatbot implementation with 200k+ users to develop a version driven by large language models that strives to be safe, mathematically accurate, and instructionally clear.

Overview and Background

While LLM-driven educational assistance has enormous potential in general education settings, there are significant challenges to developing safe, accurate, and reliable models for K-12 students, especially for mathematics students. General LLMs perform more poorly in mathematics relative to other domains, they often confabulate information, and they are in general insensitive to local instructional contexts in which most K-12 students find themselves.

Building off of LiveHint, which has been embedded within a particular curriculum in order to provide problem-specific assistance, scaffolding, and other high-quality instructional design principles, LiveHint AI extends the capabilities of LiveHint to provide dialog-based AI assistance while taking advantage of the safety, accuracy, and reliability afforded by a technology sensitive to students' local instructional contexts.

LiveHint supports students as they work on assignments by, first, reiterating the directions and then providing three to five hints on each practice problem in the homework assignment via a chatbot. Students can access these hints on a smartphone, desktop or laptop computer, or tablet.

When they are free to do so, students often do not ask for assistance (e.g., after making an error (Wood and Wood 1999)), and students often do not realize what kind of help they should seek (e.g., hints vs. answers (Nelson-Le Gall 1985). In their interactions with the LiveHint chatbot, students do not enter a natural-language query, but simply reference the question they are working on, and hints developed for that question are provided. Students are then given the opportunity to rate each hint, according to how useful they found it.

General Design of LiveHint AI

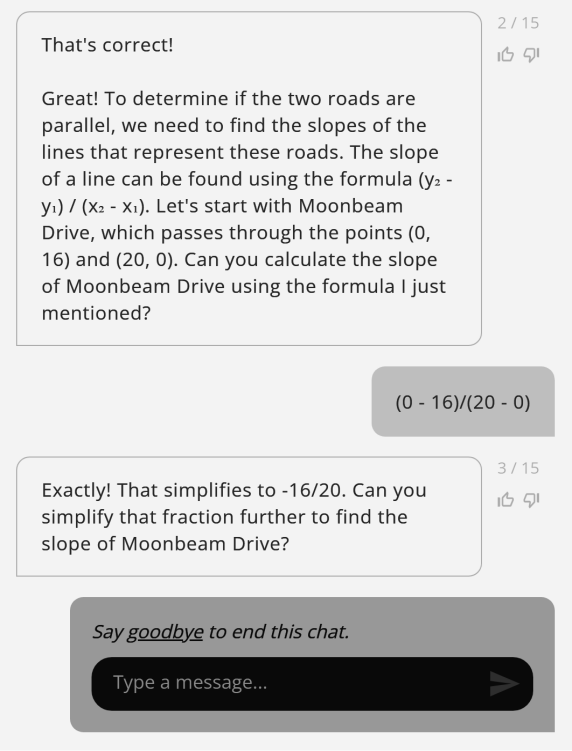

Figure 1: Screenshot from LiveHint AI, showing an example of a possible conversation with LiveHint AI. Students are able to see turn limits and can rate any AI tutor message.

LiveHint AI builds and improves on existing LiveHint technologies in two important ways. First, using the same chat structure as LiveHint, LiveHint AI now provides students with access to real, free-response, dialog-driven tutoring powered by carefully controlled and intentionally engineered LLM implementations. Instead of relying on processing a limited number of static hints, students can work through a problem while conversing with a virtual tutor engineered to helpfully orient students to problems, respond to student questions and confusions, and help explain the mathematics in appropriate ways without

directly giving students answers. Second, LiveHint AI combines student usage, preference, and performance data from LiveHint, skill data from Carnegie Learning's MATHia software, human instructional design, and safety measures driven by LLMs (e.g., toxicity detection) and other constraints (e.g., turn and session limits) to carefully constrain and monitor students' interactions with LiveHint AI and to maximize the accuracy, clarity, and impact of those interactions.

Safety and Testing

LLMs are prone to hallucinate, have well documented deficiencies in mathematics and can be encouraged to go off topic in ways that may expose bias or otherwise be particularly inappropriate for middle schoolers. We have been cautious in rolling out LiveHint AI and have focused on guardrails and testing.

LiveHint AI is instructed to remain on topic and detect and reject attempts of prompt injections (which aim to change the purpose of the LLM instance), leakage (which aims to leak the system prompts), and language inappropriate to middle school-age students. LLM instructions are often not enough to completely deter these attempts, and thus, an intermediary BERT-based toxicity detector (based on Hanu and Unitary team, 2020) is implemented to check each student and AI tutor message. We implement different thresholds of what triggers a toxicity detection based on the length of the message (shorter messages are more likely to trigger false positives) and on the existence of numbers and symbols (which are also more likely to trigger false positives). When toxic language is detected, LiveHint AI immediately ends the conversation. We also found that prompt injections and adversarial prompts are more effective in longer conversations with many turns, therefore, we limit the number of turns. Finally, transcripts are made available to teachers and parents.

Our testing has included using a “redteam” LLM focused on trying to jailbreak the main model as well as internal testing focused on mathematical correctness and instructional appropriateness. We test the model with external users (through Prolific) to gain a wider variety of responses. Finally, we are testing with students from a single district (with parental opt-out), in order to gain experience with realistic student inputs. In all cases, users can provide feedback on a turn-by-turn level and for the whole session. We have designed the system to “failback” to standard LiveHint for particular users and/or problems, so inappropriate usage or problems supporting particular problems in the Assignments] revert to LiveHint.

Accuracy and Clarity

Of central importance in tutoring situations is that the agent responsible for the instruction–virtual or human–be deeply knowledgeable of the instructional content such that it (or they) can support students' learning with consistently clear and accurate information. This ideal can be difficult to achieve regardless of the agent directing the instruction, but there are unique challenges to producing high-quality, accurate, and clear math instruction and tutoring with LLMs, which are, primarily, text-completion systems. Large Language Models often confabulate information, perform somewhat poorly in mathematics relative to other domains, and of course lack important contextual information that would facilitate clear communication with students from a variety of backgrounds and at all different levels of understanding.

LiveHint AI addresses these issues using a vast store of LiveHint and MATHia skill data, which connects knowledge of student strategies and historical domain-specific performance with expert-level solution strategies and instructional design review and testing. The outputs of this design are simple, problem-specific solution strategies that students can follow and which have been vetted for mathematical accuracy using both human and LLM-powered review. The LiveHint AI virtual agent is instructed to follow these strategies in delivering its tutoring, which dramatically reduces or eliminates confabulations (or "hallucinations"), sharply increases mathematical accuracy, and provides the agent with generalized student-level and curricular context to enhance clarity.

In addition, the data underlying LiveHint AI also provide for step-by-step mathematical reasoning and problem-specific emphases which together and separately enhance the clarity of the virtual agent's responses, guide students' attention to important elements of the session, and provide for a virtual tutoring experience that more closely resembles high-quality human tutoring.

Future Ideas

Several challenges remain to improve LiveHint AI. We aim to improve upon the personalization of LiveHint AI by remembering critical information from the students’ past sessions or their performance in MATHbook]; identify more helpful prompting techniques, as driven by student data; and build more cost-efficient implementations that will allow us to scale LiveHint AI to hundreds of thousands of students.

References

Hanu, L., Unitary team: Detoxify. Github: https://github.com/unitaryai/detoxify, (2020).

Nelson-Le Gall, S.: Help-seeking behavior in learning. Review of Research in Education 12, 55–90 (1985).

Wood, H., Wood, D.: Help seeking, learning and contingent tutoring. Computers & Education 33(2–3), 153–169 (1999).