MathyAI: An LLM-based Educational Virtual Assistant Application for Elementary Math Education

CS 224V: Conversational Virtual Assistants with Deep Learning

Priscilla Zhao[1] **Alejandrina Gonzalez**[2]
School of Education[1] School of Engineering[2]
Stanford University
{zp111611, alegre}@stanford.edu

December 23, 2022

Abstract

In this work, we explore the potential of Large Language Models (LLMs) in supporting elementary math education. First, we experiment with various ways of prompting an LLM, in this case GPT-3, to provide a multi-step conceptual explanation to various math topics in the elementary math curriculum from Khan Academy. We benchmark the various prompting methods explored. Second, with those findings we build an original system: an elementary math conversational virtual assistant we call MathyAI. Finally, we conduct preliminary user studies of the system to analyze how it could be improved. While our project touches on related research of using LLMs to solve math problems, our focus was primarily on the potential educational application of it.

All **code** can be found here for the data processing repository and here for the user-facing system repository. A **demo** of the system can be watched here.

1 Introduction & Motivation

Education chatbots have potential in supporting children's learning [8]. It not only makes the learning process more engaged, short and interesting, it also offers students lacking educational resources additional academic support [1]. The beauty of high-quality math tutor is their capability to give step-by-step feedback and hints that are specific to the particular solution path of the student [5]. In addition, the Covid-19 pandemic has caused tremendous disruption in education, with an estimated 1.5 billion students affected in at least 190 countries [10]. This has led to a tremendous learning loss among students, particularly in mathematics [9].

Motivated by this, for our project, we built an elementary educational math chatbot (conversational virtual assistant tutor), called MathyAI, with the purpose and aim to help students learn math more efficiently, by providing step-by-step guidance and instructions to help them solve problems and learn the concepts behind those problems. We do this by leveraging LLMs to provide multi-step conceptual explanations to the student. To do so, we first explore and experiement how to best prompt an LLM, in our case GPT-3, to give a conceptual explanation and, if possible, also an answer (at the end of the conceptual explanation) to various math modules one would expect in a typical elementary math school curriculum. Then, with those findings, we incorporate them into a conversational schema that underlies our original conversational chatbot system, MathyAI, for it to be able to provide conceptual hints in a multi-step conversation about a particular math module and question.

2 Related Work

Research on the applications of AI to help education has been an active area of research for at least the last decade with the boom of AI. Piech et al. have used various deep learning techniques to better understand students, model student learning as students interact with coursework, and provide feedback to students [7]. MathBot, an automated text-based tutor, could provide explanations of math concepts, practice questions, and feedback and the study found that conversational agents have been validated as promising in supporting online math education [4]. Ruan et al.'s study showed that a narrative-based tutoring system with chatbot-mediated help supports effective learning and improves engagement [8]. The feedback support systems they designed include a hint system and a chatbot. Students can type or speak to the chatbot, and it responds with personalized clues. The study is perfectly aligned with our interests of providing personalized hints that could be congruent with students' needs.

More recent work on applied AI for education has focused on the use of large language models (foundation models) in educational research. Foundation models have already started to improve the performance of some specific tasks in education, such as using MathBERT to power "knowledge tracing", the tracking a student's understanding over time based on their past responses, and the "feedback challenge", where an algorithm has to interpret a student's answer to a structured open-ended task, such as a coding question [2]. One limitation of language models is they are limited in questions that require complex quantitative reasoning. Recently, Lewkowycz et al. introduced Minerva (a Google-owned non-public model) which achieved incredible results in solving problems that require complex quantitative reasoning [6].

Though there has been extensive work in leveraging Learning and Language Models (LLMs) to solve math questions, there has been a lack of attention placed on using them to develop students' conceptual understanding. We aim to address this issue with this project.

3 Approach – What We Did

3.1 Datasets

Our first dataset is the Khan Academy dataset subset of AMPS, which has 693 exercise types with over 100,000 problems and full solutions. Problem types range from elementary mathematics (e.g. addition) to multivariable calculus (e.g. Stokes' theorem), and are used to teach actual K-12 students. Our project is focused within the following modules from the dataset: (1) Writing expressions; (2) divide mixed numbers; (3) common fractions to decimals; (4) arithmetic sequences; (5) age word problems; (6) adding and subtracting fractions with unlike denominators word problems; (7) adding and subtracting decimals word problems. Further, we augment our dataset for each module by generating synthetic math problems with GPT-3, then validating that each generated problem is solvable (that the problem generated makes sense), and labeling (the correct answer) manually since synthetically generated math problems with GPT-3 do not come with an answer. For each module, we create a train/test set split since the original dataset does not come with a test set.

Our second dataset is the GSMK8 dataset [3]. This dataset contains 8.5K high-quality and linguistically diverse grade school math word problems consisting of basic arithmetic operations that should all be solvable by a skilled middle school student. Since we mostly use this dataset for benchmarking our prompting method, we only use the test set provided by the dataset. Since we validate the answers manually when benchmarking, we take a significant subset from the test set to be our test set.

While we use both datasets for benchmarking, we only use the Khan Academy dataset for our user-facing system since our system is designed around mathematical modules by grade-level modeled after the Khan Academy elementary math curriculum.

3.2 Hypotheses

Our project is driven by three general educational and GPT-3 hypotheses, which are as follows:

1. GPT-3 will have a higher accuracy when tackling lower complexity math topics, particularly those of a lower grade level.

2. By prompting concepts underlying the question, students should be able to not only understand, but also solve the question independently. Similarly, prompting GPT-3 for the concept will help it generate the correct answer.

3. Students will gain a greater understanding of certain topics and questions by utilizing our virtual assistant, when compared to those using a simpler prompting system.

While the scope of the class does not allow us to fully test our three hypoethesis, we were only able to test for (1) and the latter half of (2). We will leave the first half of (2) and all of (3) for future work.

3.3 Prompting Methods

We delved further into the datasets and the potential of the Generative Pre-trained Transformer 3 (GPT-3) to provide hints and solutions for the questions. We examined numerous techniques for prompting GPT-3 to provide conceptual clues and answers for the math questions present in various Khan Academy K-8 modules that we could integrate into our chatbot system. To do this, we studied the Common Core Math standards, which outline the criteria for math instruction and practice, to determine the best way to guide students through the modules.

As such, we evaluated four distinct ways to prompt GPT-3. First, we explored standard prompting, as seen in Figure 1 (a), which is essentially zero-shot. We use this standard prompting method a baseline prompting method. For our second method, following out hypothesis (1) in 3.2, we develop the ConceptChain method, which consists of prompting GPT with the math question and a conceptual question, then the output is appended to the prompt to generate the next conceptual prompt, and so on, until we prompt for the answer – see Figure 1 (b). The third prompting method we explore, is a variation of our ConceptChain method and the idea introduced by Zhang et al. of appending the phrase "think step by step" to the prompt, see Figure 1 (c). Finally, we also explore the use of Chain of Thought prompting, as presented by [11], see Figure 1 (d).

3.3.1 Evaluation

We benchmark our two methods by comparing the results to the results of the standard (baseline) prompting and Chain of Thought (CoT) prompting following [11]. We use both the Khan Academy and the GSMK8 datasets for evaluation. Additionally, we were interested in comparing the accuracy of GPT-3 in providing conceptual hints versus answers. For qualitative examples of the prompting we evaluated, see Figure 1. Subsequently, this allowed us to pinpointed the most effective prompt methods to extract multi-step explanations for the various modules of the Khan Academy dataset and integrate them to our original chatbot system which is described in 3.4. As noted in the abstract, the **code** for data pre-processing and prompting methods evaluation can be found here.

3.4 Original System

We then designed our original system and conversational schema based on the prompts we explored. The general logic of the system that handles user input, system response and system output, can be seen in Figure 2. The user interface of the system can be seen in Figure 3.

Our system is designed to handle various Khan Academy learning modules, as described 3.1 and it is flexible so that additional modules could be added. Say we got $1, ..., N$ modules, we want to be able to classify whether a question q belongs to any of the $\{1, 2, ..., N\}$ modules. If it does not, then the system replies that it has not yet learned that particular topic. If we find that q belongs to a module in $\{1, 2, ..., N\}$, which is identified by the Module Classifier in Figure 2, then we are able to route the system to generate a multi-step conceptual conversation that is specific to that particular module. This guarantees higher accuracy on the hints and concepts provided to the user, which is

Q: "Write an expression to represent: Eight more than the product of two and a number x."

A: $8 + 2x$

(a) Standard (baseline) which consists of zero-shot prompting. This serves as our baseline when experimenting with various prompting methods.

Here is a math question: "Write an expression to represent: Eight more than the product of two and a number x."

What concepts can we use to solve this?

We can use the concepts of addition and multiplication to solve this.

How can we approach this problem?

We can start by looking at what we know. We know that the expression represents the result of adding eight to the product of two and some number x. This gives us a good starting point. We can see that the expression involves adding eight to the product of two and some number x.

How would you solve this?

The expression would be: $8 + 2x$

(b) ConceptChain is our prompting method to prompt for various conceptual steps in a math problem and the answer all the way at the end. Visualized is the complete chain conversation.

Here is a math question: "Write an expression to represent: One more than the quotient of a number x and 4."

What concepts would you use to solve this? Think step by step.

Step 1: What is the problem asking for? In this problem, you are being asked to write an expression that represents "two more than the quotient of a number x and 3."
Step 2: What is the quotient of a number x and 3? The quotient of a number x and 3 is the number of times that 3 goes into x, or x divided by 3.
Step 3: How do you represent "two more than" in an expression? In an expression, you can represent "two more than" by adding 2 to a number.
Step 4: Putting it all together, the expression that represents "two more than the quotient of a number x and 3" is $2 + \frac{x}{3}$.

(c) ConceptChain + "Think Step by Step" method. A variation to our ConceptChain prompting method, which prompts for concepts to solve the problem but also has the appended phrase "think step by step".

Q: "Write an expression to represent: Three minus the product of two and a number x."

A: If we subtract $2x$ from 3, we have $3-2x$. So the answer is $3-2x$.

Q: "Write an expression to represent: Nine more than the quotient of two and a number x."

A: If we add 9 to $\frac{2}{x}$, we have $\frac{2}{x}+9$. So the answer is $\frac{2}{x}+9$.

(d) This is Chain of Thought (CoT) prompting, where the pink highlight indicates the manually added thought chain.

Figure 1: All the four prompting methods described in Section 3.3 using questions from the Writing Expressions module of the Khan Academy Dataset. Green highlight indicates GPT-3's output. Pink highlight indicates the manually added CoT – only present in (d).

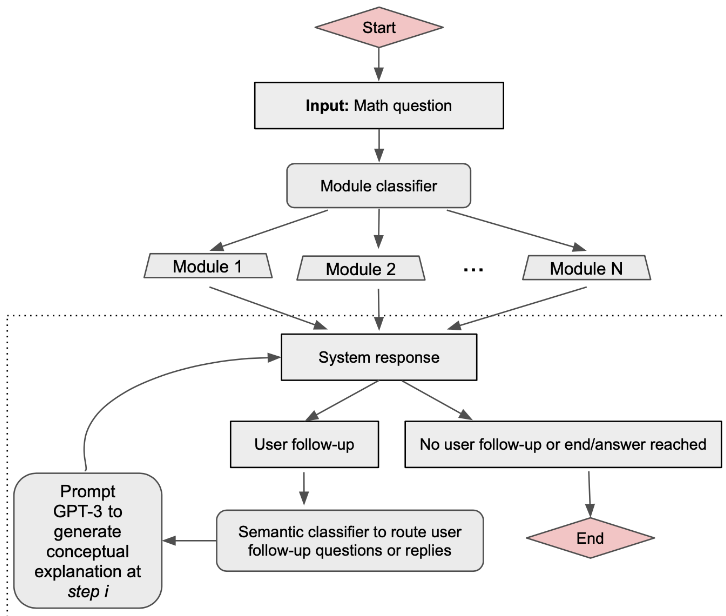

Figure 2: The logic schema behind the user-system conversation. The section boxed by a dotted line shows the part of the system that handles the back-and-forth conversation until the conversation ends.

preferred over a system that prompts for concepts but lacks the context of the question. Figure 2 shows this logic schema *outside* the dotted box.

To allow the user to be able to develop their problem-solving skills, we avoid giving the answer to the problem until they have stepped though the multi-step conversation that explains the problem by giving conceptual hints and the underlying concepts to a math problem. Figure 2 shows this logic schema *inside* the dotted box, where the Semantic Classifier handles user inputs to continue the conversation, such as user requests for additional hints or concepts (when the user has a follow-up for additional help). To generate a conceptual explanation for the user, we leverage GPT-3 with the prompts we designed and described in Section 3.3. The GPT-3 prompts used are not shown to the user, but they generate the system response when handling a hint or concept request from the user.

The conversation ends once the user reaches the answer either given by the system (last step of the conversation schema) or the user does not need any more conceptual hint or explanations. Finally, it is important to note that the system can handle a question from the user from a module that we support, or the user can ask for an example question which instructs the system to pick a random question from the database of questions. As noted in the abstract, the original system **code** can be found here.

3.4.1 Evaluation

To evaluate our original system, we design a user study methodology, described in Section 3.5.

3.5 User Study Methodology

Given the academic quarter time constraints and the difficulty of testing the system with K-8 kids without numerious permissions (such as the IRB), we designed a preliminary user study that would guide us to improve the system and study methodology for future studies with K-8 kids. We include a draft of the IRB protocol in the Appendix A.

3.5.1 User study baseline

Our baseline user study involves a single hint system instead of our multi-step conceptual system. The system hint system refers to the user only asking for a hint and an answer to a math problem. Here is the example simple hint prompting used:

- Here is the math problem:
- Give me a hint, please.
- What's the answer?

We recruited five subjects for the baseline user study.

3.5.2 User study with original system

Our user study methodology for our system consists of three components: Control/baseline user study, Scenario 1, and Scenario 2. The control/baseline user study uses a simple hint system (not using our original multi-step conceptual system), while for Scenario 1 and Scenario 2 uses our original system. We recruited six subjects for the original system study encompassing Scenario 1 and 1.

We designed 2 scenarios to test our chatbots' ability to understand users and support users' math learning. Scenario 1 involved the system presenting users with an random example problem and asking them to work through it with the chatbot's assistance. Scenario 2 involved asking the user to select a random problem from our database and give it to the chatbot to work through it.

Throughout the scenarios, users were asked to act as if they were completely unfamiliar with the problem at hand. This included questions such as "Do you know what kind of problem this is?" and "Do you understand how to solve this problem?" The users should ask follow-up questions to gather additional information to solve the problem. If the user's question was not understood by the chatbot, they were asked to rephrase their question in order to receive the additional hint. This is an important aspect of the test, as it allows us to evaluate the effectiveness of our semantic classifier in understanding user questions.

The guiding questions of our proxy user study are the following:

- How many users completed the task successfully?
- What were the most common mistakes made by users during the task?
- Did users have any feedback or questions about the system and its performance?

These questions will help us to evaluate the effectiveness of our chatbot and identify areas for improvement.

4 Results & Evaluation

4.1 Original System Demo

As mentioned in the abstract, a **demo** of the system during a user study can be found here. Figure 3 also shows a screenshot of the chatbot.

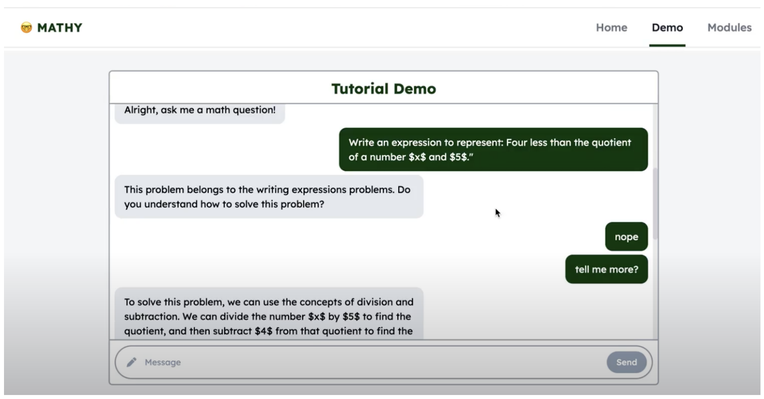

Figure 3: Screenshot of our original system during a user study screen recording.

4.2 Prompting Methods Evaluation

We use `text-davinci-002` to evaluate the various prompting methods described in Section 3.3, unless otherwise noted since we used `text-davinci-003` for additional accuracy benchmarking last minute.

4.2.1 General Methods Evaluation

For the general benchmarking of the methods explored, we used the Khan Academy dataset test set we curated and the GSMK8 dataset test set as described in Section 3.1 and Table 1.

As we see in Table 1, we focus on evaluating four promising prompting methods. The four prompting methods are the following: (1) Chain of Thought (CoT), which is was introduced by Wei

Test Set, Prompting Method Answer Accuracy (%)				
Dataset	CoT	ConceptChain	ConceptChain + "Think Step by Step"	ConceptChain + "Think Step by Step" + `text-davinci-003`
Khan Academy[1]	58.33%	61.19%	65.67%	74.24%
GSMK8[2]	46.9%	22.5%	48.75%	65.03%

Table 1: Various prompting methods we explored. All values are based on GPT-3's model `text-davinci-002` unless otherwise noted. **CoT** = Chain of Thought. **ConceptChain** was our initially designed prompting method to mainly extract conceptual explanations, however here we use it to also generate the answer to the math question. "Think step by step" is the method that appends that exact phrase at the end of the prompt.

[1] Khan Academy overall best encompassing all the modules described in 3.1.

[2] CoT value for GSMK8 is the best value reported by [11]. And for all values other than the CoT value, we evaluate on a test set proxy, a significant subset randomly sampled from the GSMK8 test set of 1.2K problems since we manually verify the output for each problem.

	Standard		**ConceptChain Answer**		**ConceptChain Concept**		
Khan Academy Module	Train	Test	Train	Test	Train	Test	Complexity
Writing expressions	91.67%	90.00%	100.0%	90.00%	100.0%	80.00%	4
Dividing mixed numbers	10.53%	30.00%	11.11%	60.00%	55.56%	80.00%	5
Common fractions and decimals	100.0%	100.0%	71.43%	53.85%	95.24%	94.87%	1
Arithmetic sequences	96.43%	100.0 %	55.56%	71.43%	94.44%	89.29%	6
Age word problems	10.00%	16.67%	33.33%	16.67%	77.78%	50.00%	7
Fractions with unlike denominators	6.25%	28.57%	22.22%	14.29%	72.22%	67.86%	2
Decimals word problems	88.24%	100.0%	87.50%	100.0%	91.67%	100.0%	3
Overall	51.66%	73.13%	53.33%	61.19%	82.72%	82.18%	

Table 2: The accuracy (%) breakdown of how well our conceptual prompting method, ConceptChain, does compared to standard prompting (baseline) method to generating answers for each of the Khan Academy modules described in 3.1. While our method is mostly designed to generate conceptual explanations (**ConceptChain Concept** column), it can also be used to generate an answer (**ConceptChain Answer** column) given the conceptual context of the question. **Standard** prompting is the baseline prompting, which is explained in 3.3. The **Complexity** rank indicates the relative level of difficulty of the modules, where a higher value indicates it is taught after the lower ranked modules.

et al. and is widely known to improve performance on mathematical questions – we use this as our **baseline**; (2) ConceptChain, our method as described in Section 3.3, and which we hypothesized in Section 3.2, second half of hypothesis (2), that it will aid in extracting both concepts and answers to math problems since it will give GPT-3 conceptual context of the question; (3) ConceptChain + "Think Step by Step", our method with a variation of the idea recently introduced by Zhang et al. to append the phrase "think step by step" to the GPT-3 prompt; and last but not least (4) ConceptChain + "Think Step by Step" using `text-davinci-003`.

First, we note that the latter half of hypothesis (2) described in 3.2 was wrong. While the final answer accuracy for Khan Academy test sets increases *slightly*, by 2.86 percentage points, the answer accuracy drops by half for the GSMK8 test set compared to the accuracy of CoT prompting. Second, we note an increase in accuracy from the previous best accuracy for each test set, +2.05 percentage points for GSMK8 and +4.48 percentage points for Khan Academy, when we prompt using our ConceptChain method with the "Think Step by Step" variation. Finally, the most dramatic increase in accuracy for both test sets comes from the ConceptChain + "Think Step by Step" prompting method using `text-davinci-003`, giving a +16.28 percentage point boost to the GMSK8 and a +8.57 percentage point boost to Khan Academy. Thus, the only difference in accuracy between the last two columns in Table 1, is the model used. Therefore, we can say that the latest GPT-3 model has an improved quantitative reasoning performance compared to the previous GPT-3 model. See Table 1 for full results.

4.2.2 Evaluating Khan Academy Modules

For the **baseline**, we first evaluate the answer accuracy of GPT-3 using standard zero-shot prompting. We found that GPT-3 was able to accurately generate answers for writing expressions, common fractions and decimals, arithmetic sequences, and adding and subtracting decimal word problems, with accuracy rates of 88% or higher in both datasets. However, GPT-3 struggled to provide correct answers for divide mixed numbers, age word problems, and adding and subtracting fractions with unlike denominators word problems, with accuracy rates below 30%. The overall accuracy of GPT-3 using simple prompting in the test dataset was 73.13%, which was higher than the accuracy in the training dataset, 51.65%. See Table 2 Standard column for more details.

To evaluate our method, we then compared GPT-3's performance using our designed concept-chain methodology compared to the baseline. We were surprised to find that GPT-3's accuracy in dividing mixed numbers problems significantly improved, with accuracy rates of 60% in the test datasets. This was significantly higher than the accuracy of simple prompting, which only reached 30% in the test. While the overall accuracy in the test dataset using our concept-chain methodology was 10% lower than simple prompting, the overall accuracy in the training dataset, 53.33% was higher than the baseline. See Table 2 under the ConceptChain Answer column.

In addition to numerical accuracy, we also paid close attention to the concept accuracy of our prompting method. Our goal was to support students' conceptual understanding in various math domains. We were pleased to see that all the modules achieved relatively high concept accuracy, particularly in writing expression, common fractions and decimals, arithmetic sequences, and adding and subtracting fractions with unlike denominators word problems, with accuracy rates above 90%. The most promising result was that the overall accuracy in both the training and test datasets was higher than 82%. See Table 2 under the ConceptChain Concept column. For full results see Table 2.

4.2.3 GPT-3 Performance vs. Problem Complexity

We are curious to investigate the general performance of GPT-3 in solving problems of varying complexity (Table 2 under Complexity column), drive by our hypothesis (1) presented in Section 3.2. Originally, we expected GPT-3 to demonstrate higher accuracy in tackling lower complexity math topics. However, GPT-3's performance in providing answers and hints does not appear to be linked to the complexity or grade level of the problem, with no correlation at all ($r = -0.086599$). More qualitatively, it failed to provide the appropriate answer and hint to a comparatively easy problem involving adding and subtracting fractions with unlike denominators, yet exhibited superior accuracy in more complex problems such as those involving division of mixed numbers. Therefore, our hypothesis (1) in Section 3.2 was wrong.

4.3 User Study

As discussed in Section 3.5, due to the short duration of the class and required documentation and permissions to test this system with k-8 kids according to Priscilla's advisor at the School of Education, we were unable to test with elementary school kids. Therefore, based on advice from Monica and Priscilla's advisor, we pursued a *proxy* study.

In our proxy study, we conducted usability testing experiments with two groups of participants, who were friends and family members of the researchers. The first group interacted with a simple hint system described in Section 3.5.1. The second group interacted with our original system, which was based on the concept-chain prompting method.

4.3.1 Baseline user study

In our experiment with the simple hint system, we invited 5 people to interact with the system. We found that although 80% of the provided hints were correct, only 1 of the 5 participants found the hint helpful. The simple hint system also had poor accuracy in providing correct answers, with only a 20% success rate. Some users indicated that they expected more explanation and additional hints in solving the problem, rather than just a high-level rephrasing of the question. For example, some feedback included comments such as "The hint and the answer are both wrong! I expected to have some equations to solve the problem"; and "I don't find the hint helpful. It basically just talks about how to solve the question in general terms. I expected something more specific, like how to convert different fractions to fractions with the same denominator. Also, the answer is incorrect." These responses suggest that the simple hint (baseline) system may not be effective in providing the level of detail and support that users need to solve math problems. For detailed information, see Table 3.

Table 3: Baseline user study

Subject	Whether the hint is correct	Whether the answer is correct	Whether hint is helpful
Lucy	Yes	No	No
Klint	Yes	No	Yes
Ray	No	No	No
Sanne	Yes	No	No
Annie	Yes	Yes	No

4.3.2 User Study with our original system

There were six participants in testing the original system, and they were asked to work on both scenarios. We found that participants generally showed more interest in the second scenario, where they were able to freely choose a question rather than being provided with one by the system. This suggests that allowing users to ask their own questions may be more engaging and effective in prompting systems.

Before starting the experiments, we provided a detailed explanation of the project, instructions on how to interact with the systems, and the objectives of the study. Overall, the participants found the hints provided by the concept-chain prompting system to be helpful, but some of them also pointed out that the hints were repetitive and not always consistent. For instance, one participant said: "The system is really cool, I would love to use it if I was an elementary school student." However, another feedback suggesting that the system could be improved, such as "It gives LONG explanations and I kind of lose the point here, I don't expect children to have the patience to read through all of this" and "Where are the instructions? What kind of questions can I ask the chatbot?".

In addition to the findings discussed above, our user study also found that users were able to ask the right questions most of the time. Some examples of successful questions included "Wait, I'm still confused"; "Could you walk me through it?"; and "Tell me more". However, we also noticed that participants sometimes failed to ask the right questions to activate the chatbot to provide additional hints. For instance, our semantic classifier struggled to recognize questions such as "That's right, so?"; "Hmmm... a little bit?"; and "What is the algebraic subtraction?" as requests for extra help. This suggests a potential limitation in our classifier and highlights the need for further development in this area. For detailed information, see Table 4 and Table 5 for Scenario 1 and 2 respectively.

5 Conclusion – What We Learned

The project provided us with the opportunity to explore the use of cutting-edge large language models (GPT-3), develop multi-step concept-chain prompting methods, and build an original system called MathyAI. Our study shows that the concept-chain prompting method achieved high accuracy in providing hints, with rates above 90%. However, GPT-3 struggled to provide correct answers, even though it was able to provide helpful hints.

Subject	What was the module that the system gave?	Which step, if any, did they get stuck on?	What was the first way they asked for a hint?	Did they reach the answer?	Did they find the explanations helpful?
Laura	writing expressions problems	1: Oh?; 2: so?	what's the next step?	yes	yes
Clara	adding and subtracting with fractions problems	1: Not at all.; 3: that's right so?	Tell me more	yes	The answer is not consistent.
Iris	writing expressions problems	1: I need help on this; 2: help me on that	I'm not sure	yes	yes
Yanni	writing expressions problems.	1:hmmm... a little bit?; 2: okay?	okay?	yes	The answer is not consistent.
Raymond	adding and subtracting with fractions problems	2:Nope; 3: not really	How come	yes	yes
Lan	arithmetic sequences problems	2; not rly	Okay, tell me more?	yes	yes

Table 4: User Study Scenario 1

We also conducted a user study to explore the usability of our original system, MathyAI. Most users preferred interacting with the concept-chain system over the simple hint system. They found the hints provided by our system to be more helpful than those provided by the baseline system. We also identified some limitations with our current application, such as the semantic classifier not always being able to accurately recognize when a user is asking a question. Overall, our study suggests that the concept-chain prompting method has potential for providing helpful support for students' math learning by providing conceptual hints and integrating that logic to an educational virtual assistant tutor.

Subject	What was the module that the system gave?	Which step, if any, did they get stuck on?	What was the first way they asked for a hint?	Did they reach the answer?	Did they find the explanations helpful?
Laura	adding and subtracting with decimals problems	1; What? 2; What is the algebraic subtraction?	How to do that?	yes	Some hints are repetitive.
Clara	arithmetic sequences problems	2; Interesting; 3;nope;	How to find it	yes	They are helpful
Iris	a conversion problem.	3; nice	How to?	yes	They are helpful
Yanni	common fractions and decimals problems	4; Aha yes	Wait, I'm still confused	yes	Hints are helpful
Raymond	writing expressions problems.	2; so?	i still don't get it	yes	The are helpful
Lan	arithmetic sequences problems	3; Okay, so?	Could you walk me through it?	yes	They are helpful

Table 5: User Study Scenario 2

References

[1] Luciana Benotti, Marıéa Cecilia Martıénez, and Fernando Schapachnik. "Engaging high school students using chatbots". In: *Proceedings of the 2014 conference on Innovation & technology in computer science education*. 2014, pp. 63–68.

[2] Rishi Bommasani et al. "On the opportunities and risks of foundation models". In: *arXiv preprint arXiv:2108.07258* (2021).

[3] Karl Cobbe et al. "Training verifiers to solve math word problems". In: *arXiv preprint arXiv:2110.14168* (2021).

[4] Joshua Grossman et al. "MathBot: Transforming online resources for learning math into conversational interactions". In: *AAAI 2019 Story-Enabled Intelligence* (2019).

[5] Kenneth R Koedinger and Vincent Aleven. "Exploring the assistance dilemma in experiments with cognitive tutors". In: *Educational Psychology Review* 19.3 (2007), pp. 239–264.

[6] Aitor Lewkowycz et al. "Solving quantitative reasoning problems with language models". In: *arXiv preprint arXiv:2206.14858* (2022).

[7] Chris Piech et al. "Deep knowledge tracing". In: *Advances in neural information processing systems* 28 (2015).

[8] Sherry Ruan et al. "Supporting children's math learning with feedback-augmented narrative technology". In: *Proceedings of the Interaction Design and Children Conference*. 2020, pp. 567–580.

[9] Johannes Schult et al. "Did students learn less during the COVID-19 pandemic? Reading and mathematics competencies before and after the first pandemic wave". In: *School Effectiveness and School Improvement* (2022), pp. 1–20.

[10] Parag Verma et al. "A statistical analysis of impact of COVID19 on the global economy and stock index returns". In: *SN Computer Science* 2.1 (2021), pp. 1–13.

[11] Jason Wei et al. "Chain of thought prompting elicits reasoning in large language models". In: *arXiv preprint arXiv:2201.11903* (2022).

[12] Zhuosheng Zhang et al. "Automatic Chain of Thought Prompting in Large Language Models". In: *arXiv preprint arXiv:2210.03493* (2022).

A Appendix: IRB Draft

Key personnel:	Priscilla Zhao (zp111611@stanford.edu); Alejandrina Gonzalez (alegre@stanford.edu)
Protocol Summary:	The purpose of this project is to test the feasibility of a STEM chatbot by conducting experiments with K-6 students from Michigan Ypsilanti Community Schools. The experiments will involve students interacting with the chatbot and completing a user feedback survey. We will collect screen recordings of the child-chatbot interactions, but these recordings will not include any identifying information about the participants.
Study Procedures:	The experiments will be conducted in public spaces, such as school classrooms and teachers' offices. Each student will participate for 10-20 minutes, including an introduction to the chatbot and instructions on how to interact with it, interacting with the chatbot, and filling the user feedback survey. We will obtain permission from participants and their parents to record the screen of child-chatbot interactions.
Audio or Video Recording:	We will conduct the screen recording of the child-chatbot interaction. It will not include any identifying information about the participants. We will obtain permission from participants and their parents to record the child-chatbot interactions and audio.
Subject Population:	The study will include K-6 students from Michigan Ypsilanti Community Schools.
Recruitment:	We will send invitation letters to households with children enrolled in Ypsilanti Community Schools, and participants will be those whose parents consent to their participation in the study.
Privacy:	The screen recordings of the child-chatbot interactions will be collected in a classroom setting. Participants will be informed of the purpose of the recordings and will be asked to provide their consent.
Confidentiality:	The researchers will be able to connect a participant's responses to their identity, but this information will not be released to anyone else. Participants' data will be kept anonymous.

Risks:	Participants may encounter incorrect information from the chatbot. The researchers will explain to participants which answers provided by the chatbot are incorrect, in order to minimize this risk.
Benefits:	Participants will have access to additional digital learning resources through their interaction with the chatbot. They will also gain knowledge about how to solve certain types of questions.
Consent Procedures:	We will obtain consent from both parents and children to ensure that they are comfortable with participating in the study.
Payment:	Children participants will receive a Stanford T-shirt as a thank-you gift for their participation.
Translation:	The study will be conducted only in English.
Copies of Questionnaires, Interview Questions:	Please see the attached document A.1.
International Studies:	The study will conduct in Michigan.

A.1 IRB Protocol Questionnaire

- How easy was it to interact with the chatbot?
- Did the chatbot provide accurate answers to your questions?
- Did the chatbot provide helpful hints or suggestions?
- Did you feel comfortable talking to the chatbot?
- Did you find the chatbot engaging or interesting?
- Did you learn anything new from interacting with the chatbot?
- Would you like to use the chatbot again in the future?
- Did you have any difficulty understanding the chatbot's responses?
- Overall, how would you rate your experience with the chatbot?
- Do you have any other comments or suggestions about the chatbot?