Combining Large Language Models with Tutoring System Intelligence: A Case Study in Caregiver Homework Support

Devika Venugopalan
Carnegie Mellon University
Pittsburgh, PA, USA
devikav@cs.cmu.edu

Ziwen Yan
Wellesley College
Wellesley, MA, USA
zy103@wellesley.edu

Conrad Borchers
Carnegie Mellon University
Pittsburgh, PA, USA
cborcher@cs.cmu.edu

Jionghao Lin
Carnegie Mellon University
Pittsburgh, PA, USA
jionghal@cs.cmu.edu

Vincent Aleven
Carnegie Mellon University
Pittsburgh, PA, USA
aleven@cs.cmu.edu

Abstract

Caregivers (i.e., parents and members of a child's caring community) are underappreciated stakeholders in learning analytics. Although caregiver involvement can enhance student academic outcomes, many obstacles hinder involvement, most notably knowledge gaps with respect to modern school curricula. An emerging topic of interest in learning analytics is hybrid tutoring, which includes instructional and motivational support. Caregivers assert similar roles in homework, yet it is unknown how learning analytics can support them. Our past work with caregivers suggested that conversational support is a promising method of providing caregivers with the guidance needed to effectively support student learning. We developed a system that provides instructional support to caregivers through conversational recommendations generated by a Large Language Model (LLM). Addressing known instructional limitations of LLMs, we use instructional intelligence from tutoring systems while conducting prompt engineering experiments with the open-source Llama 3 LLM. This LLM generated message recommendations for caregivers supporting their child's math practice via chat. Few-shot prompting and combining real-time problem-solving context from tutoring systems with examples of tutoring practices yielded desirable message recommendations. These recommendations were evaluated with ten middle school caregivers, who valued recommendations facilitating content-level support and student metacognition through self-explanation. We contribute insights into how tutoring systems can best be merged with LLMs to support hybrid tutoring settings through conversational assistance, facilitating effective caregiver involvement in tutoring systems.

CCS Concepts

• **Human-centered computing** → **Human computer interaction (HCI)**; • **Applied computing** → **Education**.

Keywords

large language models, tutoring systems, hybrid tutoring, K-12, mathematics education, caregivers

1 Introduction

Caregiver (i.e., parents and members of a child's caring community) involvement positively contributes to student outcomes, including academic performance and motivation [10]. However, limited research has studied caregiver involvement in learning analytics applications, such as tutoring systems and student dashboards. While tutored students learn at approximately the same rate per practiced problem-solving step, large-scale analyses found notable variation in prior knowledge [14]. If learning rates are near-constant and prior knowledge is variable, more practice opportunities could help close knowledge gaps. Motivational human support beyond cognitive tutoring, which caregivers can provide [10], is needed to help increase opportunities to practice. To do so, our field must study systems and analytics that enhance human support in tutoring system practice; a key idea we investigated was the need for caregivers to receive instructional support, along with guidance on effective tutoring strategies.

One emerging topic of interest in learning analytics is hybrid tutoring, where students are supported by an intelligent system and a human tutor [41]. Hybrid tutoring provides both instructional and motivational guidance [41]; this is similar to roles caregivers assert in homework support [10]. However, caregivers often struggle in proving adequate instructional homework support to students [31]. To address this issue, recent research has investigated novel ways in which caregivers can be involved in tutoring systems [27]. One promising direction suggested by this line of work is providing instructional support to caregivers. This can bridge knowledge gaps that caregivers often report regarding modern math curricula [27, 31]. Other guidance can include suggesting effective tutoring practices for caregivers. However, while prior work suggested that caregivers appreciate conversational recommendations [27], they

also noted a lack of personalization and contextual relevance in pre-generated messages. The current study addresses generation of caregiver message recommendations in a hybrid tutoring context, in real time.

LLMs have demonstrated great potential to support learning analytics applications and learners, including conversational support for debugging during problem solving [22], contextual reflection triggers in collaborative learning [26], prediction of self-regulated learning [47], and virtual teaching assistants [19], among others. These advancements promise to improve human learning through conversational tutoring. While promising, a fundamental challenge in the design of these tools is that foundation models are domain-general: they lack expert knowledge on domain-specific instruction. To circumvent this issue, recent research has suggested using instructional material as prompting aids that induce automated tutoring in a domain [36]. Other work has suggested prompting foundation models with pedagogically meaningful guardrails, such as ensuring that answers are withheld, as is practiced in effective human tutoring [30].

We study the integration of LLMs into a middle school tutoring system for equation-solving, Lynnette [21], in a hybrid tutoring context, an emerging area of interest in learning analytics [42]. We designed the *Caregiver Conversational Support Tool (CCST)*, which provides personalized, contextually relevant message recommendations through LLMs for caregivers supporting learners in tutoring systems. To our knowledge, this study is the first to evaluate LLM-generated message recommendations based on both contextual information supplied by a tutoring system (e.g., hint use, accuracy, and next viable problem-solving steps) and tutoring principles. The current study presents two angles on developing and evaluating LLM-integrated tutoring systems. First, we describe prompt engineering experiments combining the tutoring system's instructional model, tutoring principles, and LLM instruction. Second, we describe an evaluation study with ten middle school caregivers involved in hybrid tutoring using the CCST, including prototype feedback and perceptions of generated conversational support during live student support. We answer these two primary research questions: **RQ1:** How can an LLM best generate conversational recommendations for middle school caregivers based on problem-solving context? **RQ2:** How do middle school caregivers perceive these conversational recommendations?

2 Background

2.1 Caregiver Support in Learning

Caregivers play a key role in their child's academic success [10]. As homework increasingly moves online [23], technology-integrated caregiver support offers new involvement opportunities [25]. Yet, research on supporting caregivers with learning analytics is limited, aside from studies on AI acceptance [24]. Previous work has focused on indirect support, like notification features [4]. However, these do not enable caregivers to provide direct instructional assistance during homework, missing the potential for caregivers to actively enhance academic performance [10].

Notably, different *types* of caregiver support can have varying impacts on their child's academic performance [7]. Given the importance of effective caregiver involvement in homework [10], there

is a research opportunity to explore how learning analytics can support caregivers in enhancing student learning. Many caregivers struggle with homework support due to unfamiliarity with modern curricula or lack of tutoring experience [12, 27]. Providing instructional insights to bridge these gaps is a promising but underexplored approach. Our study addresses this gap by generating chat message recommendations to support caregiver involvement through a tutoring system.

2.2 Conversational Support and Hybrid Tutoring

Conversational interactions are central to learning and teaching [28, 37]. Effective dialog can guide students through problem-solving processes and provide real-time feedback tailored to their individual needs. Recent research has explored how LLMs can enhance tutoring conversations [17, 30, 36]. For example, Khan Academy developed Khanmigo [13], an LLM-powered tutoring system that uses conversational interactions to assist students across various subjects.

LLMs offer the potential to enhance hybrid tutoring and support human tutors. Lin et al. [17] utilized prompting and fine-tuning approaches to develop an automated feedback system for tutor training, guiding tutors in delivering effective tutoring strategies during conversations. Despite the potential of LLMs [30, 36], a significant gap remains between these LLM-based systems and traditional intelligent tutoring systems (ITS), which are designed with instructional models that evaluate learner performance and provide adaptive instruction [21]. In contrast, most current LLM applications in education primarily offer feedback based on textual interactions, and rarely incorporate real-time log data to support student learning. This limitation might reduce the ability of LLMs to offer the level of adaptive, data-driven instruction typical for ITS [11]. Our work bridges this gap by integrating LLM strengths with the adaptive capabilities of ITS.

2.3 Large Language Models in Learning Analytics Application

LLMs have become increasingly relevant in learning analytics [46], demonstrating their effectiveness in various educational tasks, such as providing automated feedback to assist student writing [6]. LLMs can understand natural language in text form (e.g., students' responses to open-ended questions) from the learning process and generate text-based learning support [46]. Notably, recent research employed LLMs for enhancing teaching and learning through dialog-based ITS [36]. These applications of LLMs in education show the potential to tailor instruction, aiding student comprehension, and aligning with learning analytics goals to enhance educational outcomes [8]. Despite their potential, LLMs face limitations and open questions in educational technology applications such as ITS.

A major concern is their tendency to hallucinate or produce incorrect information [43], which can lead to the dissemination of false or misleading content, confusing students or reinforcing misconceptions [29]. Another concern is that while LLMs can provide correct answers, they often do so without sound instructional

principles, which can obstruct students' deeper learning and comprehension. To address these limitations, recent research has increasingly focused on the Retrieval-Augmented Generation (RAG) approach [16]. RAG allows LLMs to retrieve relevant information from external repositories, such as authoritative educational literature, to enhance their responses. The RAG approach can help reduce the likelihood of LLMs generating hallucinated or pedagogically unsound responses by grounding the generated content in credible, context-specific information [9]. Despite its potential, the application of RAG techniques in the context of tutoring remains underexplored, which is one of the focuses of the present study.

3 Computational Methods, Recommendation Design, and Prompt Engineering

This study aimed to engineer a tool providing LLM-generated conversational recommendations, incorporating content-level support and expert tutoring principles for caregivers helping their student in a tutoring system.

3.1 Tool and Instructional Context

3.1.1 Overview. The CCST aims to guide caregivers as they support their child during ITS practice. We embedded the CCST in the equation-solving Lynnette tutor [21]. Like most ITS, Lynnette provides students immediate feedback as they work through multi-step equation solving, including error-specific messages and hints. Students can practice equations with Lynnette alone, as with a typical ITS; however, the CCST also provides the opportunity for caregivers to join their child's practice and assist them through a chat panel. The CCST monitors the interaction and provides conversational support to caregivers in the form of (1) message recommendations and (2) problem-solving path previews.

One key advancement of the CCST is including intelligence from instructional models into prompting for addressing pedagogical limitations of LLMs [15, 39]. We used problem-solving context and adaptive instruction from Lynnette to generate message recommendations for caregivers helping their student during math practice. Like many ITS, Lynnette features a domain model for assessment, feedback, hints, and other adaptive instructional support. In the CCST, we leverage the same student action recorders used for tutoring that trace student behaviors, to prompt LLMs. We classify types of recorded student behaviors to sample from evidence-based human tutoring [41]. For instance, if the ITS graded the last attempt as incorrect, dialog related to reacting to errors would be sampled.

3.1.2 Caregiver-Student Interaction Design. As students solve math problems with the Lynnette ITS, the CCST provides a button to notify their caregiver through SMS. This SMS includes a link for the caregiver to open Lynnette in their browser and join their child. The following descriptions highlight the caregiver's interactions with the tool.

When the caregiver joins the problem set, a chat panel appears on both the student's and caregiver's screens. The CCST includes the following features for caregivers (see Figure 1). The first is a real-time display of their child's problem-solving screen. Through synchronization and displaying live problem-solving steps, caregivers can see their child's practice progress. This includes seeing

the last submitted graded problem-solving attempt and hint requests. The next feature is instructional guidance through two dropdowns. One dropdown provides suggested next steps in solving the current equation, and is updated as the student submits problem-solving attempts. The second dropdown provides chat message recommendations; caregivers can select and edit these recommendations to send via the chat panel to their child. After each problem solving step attempt and chat interaction, the message recommendations dropdown is updated, as an LLM generates new recommendations based on the current problem-solving context (see Section 3.1.3).

3.1.3 CCST Features and Modules. The main components of the CCST can be summarized in three modules: the tutoring system's frontend, its instructional model, and LLM Python server, all of which are visualized in Figure 2.

The frontend is where caregivers and students interact with each other and the tutoring system. The underlying instructional model of the tutoring system includes student action recorders that trace student and caregiver behaviors for prompt generation and also aid in adaptive instruction. If the student submits an attempt at solving a problem step, or if either the student or caregiver sends a chat message, relevant contextual information is sent from the client to a backend Python server. This includes the current equation being worked on, the last-attempt accuracy, hint usage, and previous chat messages. The tutoring system's instructional model also constructs solution paths to the current equation based on rules outlining viable transformations [21]. After classifying whether the solution paths lead closer to the final solution from the current state, up to three next steps, ordered based on their proximity to the desired end state of the problem (i.e., a solution for X) are included in the context sent from the client to the backend Python server.

The real-time problem-solving context described above, along with additional content (e.g., persona, example responses integrating best practices for tutoring, specifications), are assembled into a prompt (for prompt engineering experiments, see Section 3.2) that invokes the Llama 3 LLM. This prompt captures problem-solving context and instructs the LLM to provide tutoring advice to the caregiver. The LLM is prompted to generate three message responses in a single run to optimize runtime while also providing caregivers with multiple recommendations. These responses are processed and sent back to the client, appearing in a dropdown on the caregiver interface, as shown in Figure 1. For this study, we chose to use the open-source Llama 3 8B Instruct model, which is the smallest version of the Llama models. This decision was motivated by our preference to (1) use an open-source model and (2) use a model that could be locally run. The instruct variant is fine-tuned for conversational support [1], and the 8B parameter size was appropriate for running locally. To run the model on our production server, we used Ollama [18], an open-source framework to locally run LLMs. Compared to closed-source, proprietary models such as OpenAI's ChatGPT, open-source models bear the advantage of stable and transparent versioning while ensuring equitable access to tool use, irrespective of financial resources and opportunities for comprehensive auditing [5].
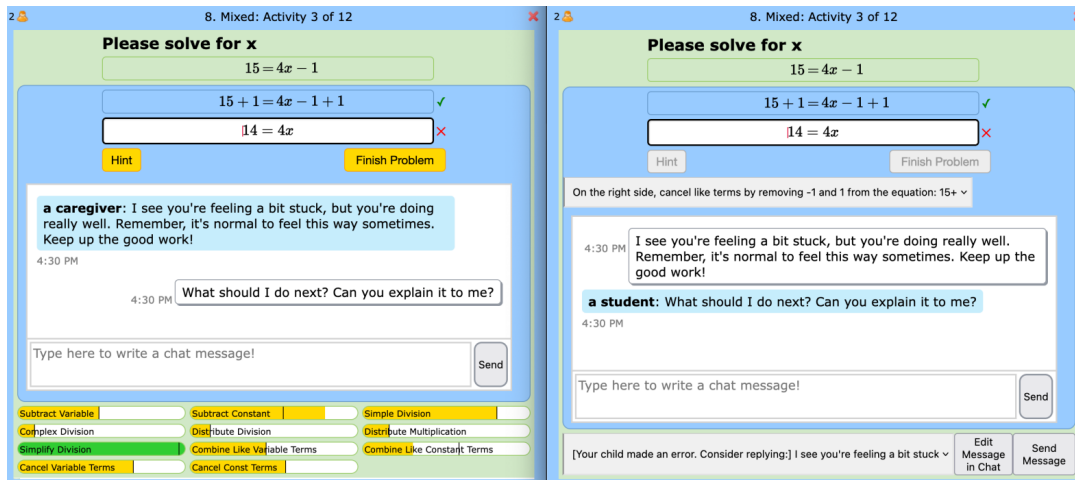
Figure 1: Child (left) and Caregiver (right) interface. The caregiver has a live view of their child's problem-solving steps in Lynnette (top of the screen, both left and right). The caregiver also has two dropdown menus, containing: (1) dynamically-generated chat recommendations (bottom) and (2) next-step recommendations based on potential problem-solving pathways (top).
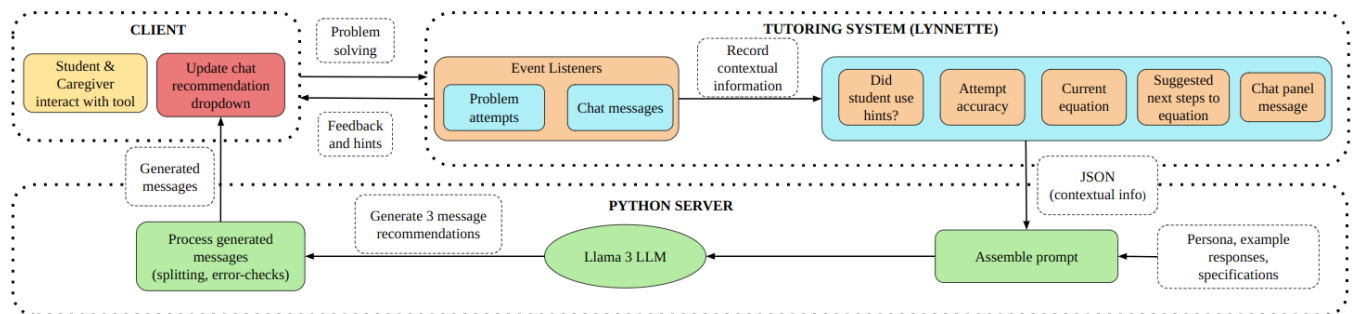

Figure 2: CCST Features and Modules. This figure displays the three main components of the CCST: (1) client, (2) instructional model, and (3) Python server

The tutoring system client and instructional model were hosted using a simple web server running on HTML+JavaScript, which communicated with an additional backend WSGI-compliant Python web server running the LLM. Our server could support the model with 32GB of RAM and 8 Intel Xeon CPUs. As each instance of (a) the caregiver or student sending a message in the chat panel or (b) the student submitting a problem-solving attempt results in a POST request to the server, we further implemented a load balancer that only allows a request if more than 30 seconds has elapsed since the last request. This timeframe was determined based on estimated average prompt processing times in the LLM module. The client sends the aforementioned contextual information in JSON format. This is an example payload of a request resulting from a sample interaction: {"chat message": "Caregiver: hey what do you need", "next step": ["Subtract 1 from both sides: x-1+1 = 3-1"], "used hint": "False", "accuracy": "error", "question": "1+x=3"}.

3.2 Prompt Engineering Iterations

The CCST's caregiver chat message recommendations are generated by invoking the Llama 3 LLM with a prompt integrating contextual information. We defined the following properties as desirable for these messages. First, following tutoring best practices, such as assessing a student's prior knowledge, responding to errors in a way that increases motivation and engagement, and giving effective praise that acknowledges effort [41]. Second, including brief, explanatory text at the beginning of each message (formatted as [explanation]: message), where [explanation] highlights the main goal of the message. Examples of these explanations are "Ask to self-explain," and "Praise your child for a correct response." Including these explanations at the front of messages allows caregivers to understand the objective of each message recommendation [27]. Third, being contextually relevant to the live tutoring taking place. A key novelty of our approach is integrating user-specific problem-solving context, along with principles of effective tutoring, while

generating message recommendations. Rather than having generalized messages, the LLM-generated messages are personalized to the current interaction, taking into account factors like the equation being worked on, hint usage, and the student's accuracy. These are reflected in generated responses, as shown in Table 1.

We evaluated our prompt engineering experiments using the CLEAR Framework for Prompt Engineering [20], a method to optimize interactions with LLMs. The five core principles of this framework involve determining whether a prompt is concise, logical (structured and coherent), explicit (clear output specifications), adaptive (customizable), and reflective (continuous evaluation and improvement of prompt). We conducted seven main prompt engineering rounds, iterating upon previous rounds until the quality of generated responses was satisfactory.[1]

We grouped the prompts used in the seven rounds into three categories, each which served as responses to shortcomings of the previous category. The CLEAR framework does not prescribe specific sample sizes, so we chose to evaluate about 50 to 80 examples of generated responses per prompt experiment.

The prompts in **Category 1** (prompts 1 - 3) followed zero-shot prompting, in which no examples of desired output were provided. These prompts provide information about the task and formatting guidelines [17, 44]. Limited contextual information (only previously sent chat panel messages) was provided. Prompt 1 instructed the LLM to assume the appropriate persona [44] and included a list of all previously sent chat panel messages. Prompt 2 specifically stated the purpose of using the list of chat panel messages (to generate a message that a caregiver would say to their child at this point in the conversation). Prompt 3 provided guidelines for length and explicitly noted that chat panel messages were delimited [17], allowing the model to differentiate between them and the instructions in the prompt.

The prompts in **Category 2** (prompts 4 and 5) used few shot prompting, in which examples of desirable responses were provided. These prompts incorporated techniques from recent advancements in prompt engineering and were primarily motivated by RAG, which involves the use of content from external sources to enhance an LLM's capabilities [9]. We integrated knowledge of tutoring principles [41] by providing literature-supported example responses in a section of the prompt. Tutorial dialog is most effective when it elicits opportunities to learn and reflect, based on the theory of "accountable talk" by Resnick et al. [34]. Such opportunities arise when dialog allows students to reflect, explain, and challenge their position. Accordingly, our integrated tutoring practices include responding to student errors, assessing what a student knows, and providing effective praise, all known to increase self-efficacy and engagement [41]. Prompt 4 provided seven example tutoring responses, each of which was preceded by detailed explanations of why those responses are characteristic of effective tutoring. Prompt 5 removed these lengthy explanations of tutoring strategies, and instead separated responses into three general categories: responding to errors, determining what a student already knows, and giving praise. This prompt included 3 examples per category of response.

---

[1] The seven main prompts are included in this GitHub repository, along with example code for our server architecture involving the use of different contextual information and assembling them into a prompt: https://github.com/devika-prog/Caregiver-Conversational-Support-Tool

The prompts in **Category 3** (prompts 6 and 7) followed our novel approach of incorporating instructional intelligence received from the ITS. These prompts continued to use few-shot prompting as the prompts in Category 2 did, but included a new section integrating session-specific problem-solving context. This content included (1) whether the student's problem-solving attempt was correct or not, (2) whether the student used hints, (3) a list of all previously sent chat panel messages, (4) the current equation needing to be solved, and (5) suggested next steps to solving the equation. Notably, these suggested next steps were generated using the tutoring system's underlying instructional model, as described in Section 3.1.3. Prompt 7 was structurally the same as Prompt 6, but added examples of including short explanatory messages to the front of each generated response [27]. This feature aids caregivers in deciding which of the messages to send, by briefly summarizing the objective of each message.

Figure 3 outlines the structure of Prompt 7. The following are 3 examples of recommendations that the LLM generated in response to Prompt 7: **(1)** [*Walk through hints*] *I love how you tried very hard and focused on the problem! Can you tell me what you understood from the hint about dividing both sides by the coefficient of x?* **(2)** [*Ask to self-explain*] *Can you walk me through your thinking in this step? Why do you think we should divide both sides by 6?* **(3)** [*Praise your child for a correct attempt*] *Great job on getting this far! It looks like you're close to solving it. Can you tell me what you did first?*

After developing each prompt, we evaluated prompt quality with the CLEAR framework [20] before using the prompt to invoke the LLM. Our prompts were (1) concise, providing solely the information needed, (2) logical, with a clear structure of components, as shown in Figure 3, (3) explicit, stating criteria for messages, including parameters like length and content, (4) adaptive, as illustrated by the integration of real-time problem-solving context, and (5) reflective, as shown by our multiple iterations of prompt engineering to address shortcomings of responses from previous prompts.

3.3 Prompt Engineering Results

3.3.1 Generated Response Characteristics. In the following section, we describe the characteristics of generated responses when invoking the LLM with prompts from each of the 3 main categories described in Section 3.2. Each response was evaluated against the 'desirable' standards established in Section 3.2.

Category 1: Chat panel messages were recognized, and content from the messages was integrated into the LLM's generated recommendations. It appeared, however, that the LLM would benefit from additional contextual information in the prompt (e.g., the problem being worked on) since the generated responses often seemed to lack context. Some responses also had phrasing that would not be used in normal conversation in this context, as highlighted in this example: *"Ah okay sweetie! I'm here to help. Take a deep breath and let's take a look at the problem together. Which one is giving you trouble? Is it a multiplication or division question? Or maybe it's something with decimals or fractions? Let me see what you're working on and we'll figure it out together!"*

Category 2: Chat recommendations integrated best practices for tutoring and closely resembled provided examples. Since little problem-solving context was provided, the recommendations were

You are a parent providing assistance to your middle-school child for their math homework. — Informing LLM of persona to assume

Here are some examples of good tutoring practices:
When responding to errors, say something like: {3 example messages for responding to errors}
When determining what a student already knows, say something like: {3 example messages for determining what student knows}
When giving praise, say something like: {3 example messages for giving effective praise} — Providing examples of best tutoring practices

{Hint advice based on prior hint use}
{Advice on accuracy of previously submitted attempt: correct/error}
This is the equation your child is working on: {equation}. They need to solve for x.
Here are suggested next steps: {list of next steps} — Integrating session-specific problem-solving context

The elements in this list are messages that have been sent in a conversation between a middle school student and their parent about a math problem (in order).
Use these messages to generate 1 to 2 sentence responses that a parent would say to their child at this point in the conversation.
Include a short justification in square brackets at the beginning of each message, such as [Ask to self explain], [Praise your child for a correct attempt], [Your child has made an error]
Do not include quotation marks. Do not give away the answer.
This is the list, delimited with square brackets: [{5 most recent chat messages, in form user: message}] — Final instructions and specifications
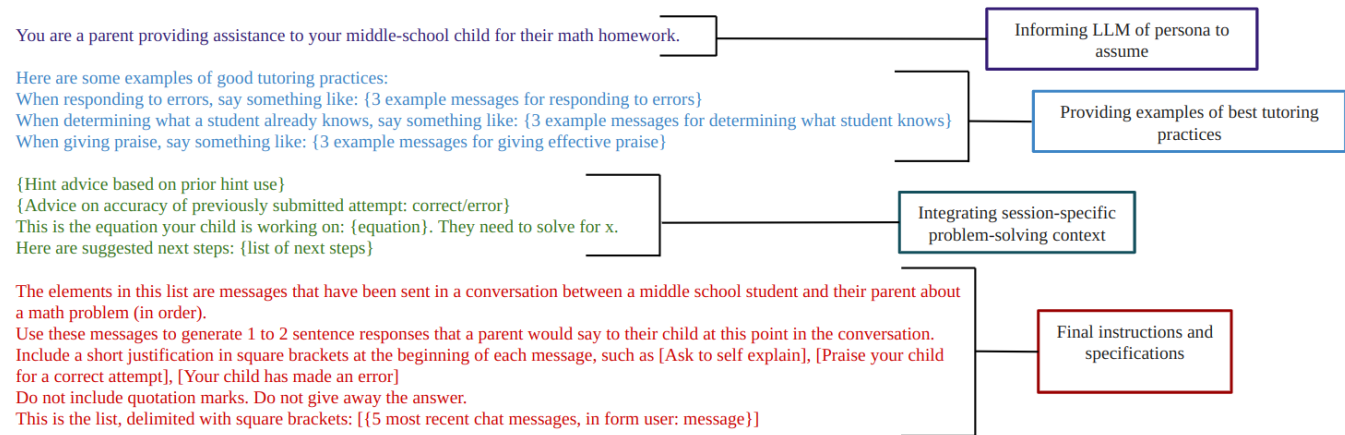
Figure 3: Prompt 7 Breakdown. Prompt components with their purpose. The section integrating session-specific problem-solving context is unique to each interaction, and its different components are represented by the placeholders in the figure. This is an example of what that section of the prompt could include: "Your child did use a hint, so ask them what they understood from the hint. Your child made an error, so you should focus on responding to the error as described earlier. Here are suggested next steps: Subtract 5 from both sides: 15-5 = -2x-5+5. This is the equation your child is working on: 15 = -2x-5. They need to solve for x."

still not specific to the given problem-solving context, as shown in this example *"Let's try solving the problem together. Can you tell me what you did first?"*

Category 3: The recommendations were more specific to the current interaction, appropriately integrated problem-solving context, incorporated best tutoring practices, and included explanatory messages to categorize different types of responses, as evidenced in this example: *"[Ask to self-explain] Can you walk me through your thinking in this step? Why do you think we should divide both sides by 6?"* Responses generated by Category 3 prompts were evaluated to be the most desirable according to our standards, mentioned in Section 3.2.

3.3.2 Incorporation of Contextual Information. An objective of this study was to determine how integrating real-time problem-solving context into a prompt allows the LLM to generate personalized, contextually relevant recommendations. Table 1 demonstrates how different values for the 5 problem-solving context variables (see Section 3.2) resulted in different content in generated recommendations. The examples included in the table are from simulated trials where a research team member provided different inputs to the tutoring system (e.g., submitting correct and erroneous responses, sending chat panel messages, using hints) to determine how generated recommendations were affected.

4 Prototyping Study Methods

The objective of this case study was to explore how LLM-generated conversational recommendations can support caregivers during synchronous collaboration with their child, as their child practices with a tutoring system. Caregivers, participating alongside their children in dyads, were recruited through a summer program for middle school students at a university in the Northeastern United States. A semi-structured 60-minute prototyping interview was conducted during the second day of a two-day design workshop. The same protocol was also conducted with caregiver and student dyads who opted to participate online via Zoom. This study focuses solely on caregiver perspectives, although the workshop protocol also included activities and perspectives of students.

4.1 Sample

4.1.1 Recruitment and Compensation. Participants for the two workshops were recruited through an outreach center affiliated with a university in the Northeastern United States. IRB approval for this study was obtained, and informed consent was taken from human subjects. For the remote interviews, caregiver-student dyads were recruited using contacts from prior design interviews in the project, primarily through social media and email lists from local caregiver outreach centers. Remote interviews were conducted after the workshops to reach a satisfactory sample size, determined by reaching saturation of qualitative insights related to our research questions, as is common practice in qualitative research [35]. Participants in the workshops received a $70 Amazon gift card each upon completion of both 2-hour sessions. For remote interviews, each caregiver and student received a $25 Amazon gift card as compensation.

4.1.2 Participation. A total of ten caregivers participated in the study, with four attending the workshops and six attending remote interviews. The participants were 40% Asian and 60% White, comprising 20% male and 80% female, with an average age of $M = 47.9$ years ($SD = 6.3$). The language of participation for all caregivers was primarily English, though one participant with limited English proficiency responded to some questions in Chinese.

Table 1: Integration of Problem-Solving Context. Demonstrates how different values for problem-solving context variables modified characteristics of generated recommendations.

Problem Solving Context and Generated Recommendations				
	Example 1	**Example 2**	**Example 3**	**Example 4**
Current Equation	6x = 12	6x = 12	6x = 12	6x = 12
Suggested Next Steps	Divide both sides by the coefficient of x, which is 6: 6x/6 = 12/6	Divide both sides by the coefficient of x, which is 6: 6x/6 = 12/6	Divide both sides by the coefficient of x, which is 6: 6x/6 = 12/6	Divide both sides by the coefficient of x, which is 6: 6x/6 = 12/6
Used Hints	True	False	True	False
Attempt Accuracy	Error	Error	Error	Correct
Chat Panel Message	Student: "I need help"	Student: "I need help"	Student: "Should I divide by 6?"	Student: "I think I got it"
Generated Response (Output)	[Walk through hints] Can you tell me what you understood from the hint about dividing both sides by the coefficient of x?	[Request a hint] How about you request a hint and walk me through it?	[Ask to self-explain] Can you walk me through your thinking in this step? Why do you think we should divide both sides by 6?	[Praise your child for a correct attempt] Great job on getting this far! I appreciate how hard you worked on this problem.
Evidence of Instructional Intelligence	Student did use hints and made an error, so recommendation prompts student to explain hint, to encourage further thought about the problem.	Student has not used hints and made an error, so recommendation is to request a hint to assist with their problem-solving.	Student sent a chat message about dividing by 6, and since they made an error, recommendation asks them to explain their thought process.	Student's attempt accuracy was correct, so recommendation is praising student in a way that follows best tutoring practices.

4.2 Material and Procedure

4.2.1 Workshop. The workshop aimed to gather suggestions for improving the design of the CCST based on caregiver needs and explore their perceptions of LLM-generated conversational support in caregiver-student interactions (RQ2).

Participants were divided into groups of one to three depending on the number of participants and facilitators, with each group guided by a designated facilitator, who recorded conversations using audio recorders. To simulate a remote homework support session, caregivers and students were seated separately and communicated through the chat panel. During this time, they worked on problem sets in the ITS together, with caregivers instructed to send chat messages to the student, check the hint function, and provide comments on these features. The facilitator guided these interactions and asked caregivers questions about the length, tone, quantity, and response time of the messages, as well as typical homework routines, opinions on tool functionality, and conversational support. To obtain more in-depth feedback, we provided a printout of additional message recommendations, which were based on examples generated through prior testing of the CCST as well as best tutoring practices [41]. Caregivers also annotated screenshots of the interface, which show a sample interaction, to suggest design improvements [38]. While the tool was fully implemented during the session, these inquiries and additional materials, along with discussions on design feedback and various use case scenarios, provided a deeper understanding of caregiver preferences.

4.2.2 Remote Interview. Each interview session lasted for one hour, and followed the same procedure as the in-person protocol. The student and caregiver were seated in separate rooms and interacted with the tool independently to simulate remote use. Screenshot annotations were omitted due to the challenges of printing and sharing annotations via Zoom. Rather, we asked participants to verbally describe any changes they would like to see and how they would implement them, allowing us to gain insights into the design

solutions they envisioned. The sessions were recorded using Zoom's built-in recording feature, capturing both the breakout rooms and the main room.

4.3 Data and Processing

Following best practices from prior work [47], all audio records of participants' sessions were transcribed using OpenAI Whisper [32] or Zoom's built-in transcription tool. Whisper transcribed and translated utterances in Mandarin Chinese from one participant with limited English proficiency. A research team member fluent in both Mandarin and English reviewed the translations and confirmed that they were of sufficient quality for thematic analysis. Another team member recorded the handwritten notes and drawings from the annotated screenshots into a spreadsheet for further analysis.

4.4 Data Analysis Methods

The qualitative data (i.e., interview and annotation data) was analyzed using a thematic analysis approach with an open coding scheme. Two research team members independently conducted a first round of inductive open coding to establish initial descriptive codes [45]. A final round of discussion and consolidation of the resulting topic centers was held to eliminate individual coder bias [40].

5 Prototyping Study Results

The following sections address RQ2 by presenting themes describing caregiver preferences for conversational support.

Theme 1: Caregiver preference for content-level support over motivational support.

Caregivers demonstrated a preference for recommendations providing mathematical guidance rather than motivational support. Six caregivers preferred more direct, content-focused recommendations, two caregivers found both types of support useful, and one caregiver preferred to provide motivational help independently of

the system. We identified two subthemes that further highlight the underlying reasons for this preference. **Caregivers find content-level messages valuable for providing the support they often feel unequipped to offer**: Messages providing content-level guidance are especially beneficial for caregivers who may feel less confident in their ability to assist with math homework. Three caregivers highlighted how these recommendations help guide them through complex problems, providing the support needed when they are unsure of the correct answers or fear giving incorrect guidance. As Caregiver C10 noted, "I think it's nice to have these options [recommended messages] because sometimes you get frustrated when you don't know the right things to say . . . you might say the wrong thing." **Caregivers did not find motivational support as helpful due to its misalignment with their current tutoring practices**: Caregivers who favored content-level support typically adopt a direct, results-oriented approach when tutoring their children, leading them to find motivational support as not useful. Caregiver C6 described motivational support as "just filler" and "not as useful." Other caregivers noted that their children "wouldn't probably react well to praising their effort" (C10) or "just want to solve the problem" (C4).

Theme 2: Caregivers preferred messages that prompt student metacognition (i.e. explanation of thought processes), as it provided them with deeper insights into their child's thinking. While the CCST generally allowed for live synchronization of student problem-solving steps, allowing the caregiver to see a reflection of their child's screen (see Section 3.1.2), caregivers articulated a desire for more in-depth insight into their child's thinking. Five caregivers preferred messages that prompt students to explain their reasoning process more thoroughly. For example, C6 mentioned asking "tell me your thought process" and "how you're thinking about this." C4 emphasized the importance of getting more information by looking at the "work paper" of the student, which may refer to scratchpad notes students craft before entering problem-solving step attempts (visible to caregivers) into the tutoring system. Similarly, C8 suggested asking "why does the work help you solve the equation." Caregivers noted that understanding student thought processes can allow them to measure knowledge gaps, which in turn allow for more targeted support. As C9 noted, "Walk me through it and can you explain to me what you did here. I think those are good they reinforce that the need to be able not just to come up with an answer but to explain and to show your work in whatever manner is required." Another caregiver C8 added that "I would say ask them just to speak out their thought process loud, so I know where the mistake is. So, you know, it can be a careless error right and or they just don't know how to do it. So I want to find out why they did it wrong."

5.1 Technical Feedback and Feasibility of Generated Messages

Caregivers also provided feedback covering technical aspects of the messages (i.e. length, style), as well as the effectiveness of incorporating problem-solving context. **Contextual Information:** The inclusion of problem-solving context (see Table 1), specifically, incorporating whether the student used hints or made errors, was

recognized and appreciated, since it provided additional opportunities to engage with and understand instructional content. C8 emphasized that "They [messages] should be based on the mistakes or the steps of which [student] got wrong to have customized." C5 expressed "Oh that's good, 'How about you request a hint and walk me through it?' Cuz he may have just asked me before he hit the hint." This context allows caregivers to gauge the student's current understanding and guide them according to their current progress. **Message Length:** Six out of nine caregivers found the messages to be too long, making them hard to process during live tutoring. For example, C6 noted that: "Like having to read through this text is cumbersome." **Number of Messages:** While eight out of nine caregivers felt the number of messages (i.e., three recommendations at a time) was adequate, one parent preferred a narrowing down to only one message. **Message Style:** Caregiver feedback on tone was mixed. Three out of seven caregivers found the tone of the message recommendations lacking, each citing that the tone is inauthentic and artificial. C5 noted: "Glad you're focused just seems artificial to me." Conversely, C10 valued the gentle tone, stating, "It's not too harsh when it says you might have made a mistake, like, using the word 'might' or 'small'. I think the tone was good." The remaining four caregivers found the tone positive and adequate, highlighting that it is encouraging. C8 remarked, "I think the tone is good. I mean, it's very encouraging. It's very positive." **Latency:** Two out of six caregivers felt that the message generation was somewhat slow, while four found it to be reasonably quick. C9 noted that "it's faster than us," highlighting that the performance is generally satisfactory, but could potentially be improved.

6 Discussion

The objective of the current study is to determine how to effectively generate conversational support for caregivers providing guidance as their child practices with an ITS. We did so by combining the instructional model of a tutoring system and the language abilities of an LLM, aiming to address LLM limitations for instruction identified in prior research [39]. We then determined caregiver perceptions on LLM-generated conversational support by conducting interviews in which we tested the Caregiver Conversational Support Tool (CCST) with middle-school caregivers.

6.1 Conversational Support Message Generation

RQ1 focused on how tutoring system capabilities and best tutoring practices can be provided to an LLM to generate conversational recommendations. We conducted prompt engineering experiments integrating tutoring system context data (e.g., correctness, problem-solving pathways) with LLM instructions using the open-source Llama 3 LLM.

Recent research has argued that LLMs lack instructional principles, such as determining a student's prior knowledge, to suffice as tutoring agents [39]. We found that the LLM was able to adequately integrate problem-solving context into generated message recommendations. We *also* provided the LLM with guidance as to *how* it could use that data from a tutoring system to provide suggestions in generated responses (see Figure 3). We observed that such instructions, alongside few shot examples on effective tutorial dialog [41], were especially useful for generating message recommendations

attuned to student actions. Further, by providing specific solution options as to how the student may solve the equation, we observed no issues related to providing incorrect, hallucinated math advice. While the tutoring system can generate correct solutions without error due to its instructional model, an LLM alone lacks arithmetic abilities [9, 48]. We only observed such hallucinations when limited tutoring system context was given in Category 1 prompts (see Section 3.2). Taken together, we observed that generated messages were not only pedagogically fit for tutorial dialog [41], but also more reliable in terms of arithmetic and tutoring advice.

6.2 Prototyping Study Discussion

RQ2 focused on caregiver perceptions of conversational recommendations integrating tutoring system and LLM intelligence. Design research employing the CCST with ten middle school caregivers provided tangible insights into how caregivers can be best supported in homework support, which is crucial for student learning [10, 27].

Caregivers appreciated content-level support and desired further opportunities to engage with content: Caregivers appreciated both content-level and motivational support messages during tutoring but saw greater value in content-level support. They appreciated messages that integrated instructional guidance in messages, including the "Suggested Next Steps," which provided concrete instructions as to how to solve an equation. These steps were particularly helpful for caregivers who felt less confident in providing accurate math help, as they offered clarity and reassurance, which aligns with prior research on caregiver support needs [27, 31]. Caregivers also valued the integration of real-time problem-solving context (i.e., student hint use and errors) within content-focused messages, especially when it helped them and their child to engage with the content. Future designs should aim to integrate such data-driven contextual information while ensuring the student-specific guidance remains actionable. This involves enhancing message specificity to align with student actions and potentially avoiding the repetition of information available in the tutoring system. Some caregivers also noted that motivational messages were less aligned with their homework support style. Future design revisions could allow caregivers to customize which messages to receive.

Conversational recommendations can help caregivers to understand student metacognition: We found that caregivers favored messages that encouraged self-explanation, an effective instructional strategy [2] that helps students articulate their reasoning and reflect on their problem-solving [33]. Future improvements should focus on tailoring these self-explanation prompts to better align with caregiver preferences. Specifically, caregiver feedback underscores the need for personalized tone in messages, and potentially improved relevance and timeliness. One major challenge identified is balancing the specificity of context with the timeliness of responses, given delays in message generation. Future designs may adapt recommendations to caregiver voice and investigate when the generative capabilities of LLMs, as opposed to pre-generated messages, may be most beneficial for conversational support in tutoring.

6.3 Implications

Caregivers describe lack of instructional support as a key barrier to getting involved with their child's online homework [27]. Hence, one central goal of the CCST was to aid caregivers in bridging knowledge gaps while supporting their child during tutoring system practice. Our results suggest that this instructional support could help caregivers in providing instructional interventions in their child's math homework [10]. The caregivers we interviewed in this study especially favored message recommendations that targeted content-level support, which were informed by tutoring system instruction and log data. Hence, as a broader implication to the field, content-level conversational support through LLMs may augment increasingly common hybrid tutoring scenarios to improve student learning [41]. Moreover, addressing recently identified instructional limitations of LLMs [39], our findings imply that tutoring system instruction and log data collected during practice can be productively integrated into LLMs to help improve the instructional rigor of LLM generations in educational contexts similar to the one studied here.

For productive merging of tutoring systems and LLMs, we note that providing tutoring system data into prompting is not enough; it requires instructions and examples as to *how* these data should be used. Thus, other applications providing context to better inform foundation models should accompany data with *explanation* when prompting an LLM, for example, by describing tutoring best practices, as done in this study [41]. This method of using an LLM and ITS in conjunction is a key contribution to the present study.

Our designs and methods may be transferred to other systems supporting learners that are common in learning analytics. For instance, collaborative learning systems, where students work together to solve problems, might benefit from the incorporation of an LLM providing instructional support. It can be challenging for middle-school students to effectively collaborate, even when rule-based support that interprets conversations is present [3], so integrating an LLM that provides suggestions based on instructional context from tutoring systems could guide student collaboration. Our findings may also inform systems involving no live human interaction. For instance, the JeepyTA [19] uses an LLM as a virtual teaching assistant. Although JeepyTA's initial evaluation found it effective in providing information about a course, it lacked capabilities to sustain student motivation and contribute to effective learning, which might be addressed through responsive and motivational message support reported here. Overall, integrating effective tutoring practices and tutoring system intelligence into LLMs may enhance personalized support in online learning. However design research with end-users will be important for sustainable adoption and improved learning. There is also a potential to expand the scope of our tool. Our current focus is math, but could be broadened to other subjects, as tutoring systems span several STEM domains such as physics, chemistry, and logic [2].

6.4 Limitations and Future work

We acknowledge three study limitations. First, our hardware and resource limitations of this project limited the speed of message generations when running LLMs, which could be further improved through GPUs. Average latencies for message recommendations

in this study were about 25 s, which at times was perceived as too slow by caregivers, especially when reference was given to specific actions in the tutoring system. Future work may investigate how recommendations can better deal with latency, for example, by updating recommendations to include less specific advice if the tutoring system detected the student to have moved on from when the request was first sent. Another point of exploration is using different LLMs beyond Llama 3.

Second, our prompt engineering method was purely qualitative, using the CLEAR framework [20]. While this method is suited for quick prompt iterations, future research may study the emerging characteristics of message recommendations through sentence embeddings, including but not limited to, clustering, to refine prompting. Quantitative curation of message recommendation may further be used to fine-tune LLMs toward more desirable output.

Third, participants may not necessarily be representative of the larger caregiver population. Considering that participants were volunteers, there is a potential bias that these caregivers are especially engaged in homework support. In addition, most were women, and all were either White or Asian. To ensure equity and inclusivity of our tool's use and design, future work could involve recruiting a larger and more diverse sample for design research.

7 Summary and Conclusions

This study investigated integrating tutoring systems and LLMs to improve conversational support in hybrid tutoring, where caregivers and tutoring systems assist students in problem solving. While previous research highlights the value of caregiver involvement in homework, little learning analytics research has studied how caregivers can be supported in asserting practice support roles. Our results show that using instruction and log data from tutoring systems in prompts enables LLMs to generate contextually relevant messages for hybrid tutors. The LLM adapted its responses based on problem-solving context, and we found that effective prompting requires not just data, but also example responses. Based on design research, caregivers valued content-level message recommendations from our system and especially appreciated those prompting their child to explain their thought process. Such metacognitive prompts show promise in helping caregivers support students in their homework constructively, which not all caregivers do or are able to. A key contribution of this work is demonstrating how LLMs and tutoring systems can work together during problem solving, using real-world data to provide caregivers with effective tutoring guidance. This approach enhances personalized, contextually relevant conversational support, advancing learning with intelligent tutoring systems.

Acknowledgments

References

[1] Lisa C. Adams, Daniel Truhn, Felix Busch, Felix Dorfner, Jawed Nawabi, Marcus R. Makowski, Keno K. Bressem, and Linda Moy. 2024. Llama 3 Challenges Proprietary State-of-the-Art Large Language Models in Radiology Board–style Examination Questions. *Radiology* 312, 2 (2024).

[2] Vincent AWMM Aleven and Kenneth R Koedinger. 2002. An effective metacognitive strategy: Learning by doing and explaining with a computer-based cognitive tutor. *Cognitive science* 26, 2 (2002), 147–179.

[3] Conrad Borchers, Kexin Yang, Jionghao Lin, Nikol Rummel, Kenneth R Koedinger, and Vincent Aleven. 2024. Combining Dialog Acts and Skill Modeling: What Chat Interactions Enhance Learning Rates During AI-Supported Peer Tutoring?. In *Proceedings of the 17th International Conference on Educational Data Mining*.

[4] Zachary Broderick, Christine O'Connor, Courtney Mulcahy, Neil Heffernan, and Christina Heffernan. 2011. Increasing parent engagement in student learning using an intelligent tutoring system. *Journal of Interactive Learning Research* 22, 4 (2011), 523–550.

[5] Stephen Casper, Carson Ezell, Charlotte Siegmann, Noam Kolt, Taylor Lynn Curtis, Benjamin Bucknall, Andreas Haupt, Kevin Wei, Jérémy Scheurer, Marius Hobbhahn, et al. 2024. Black-box access is insufficient for rigorous ai audits. In *The 2024 ACM Conference on Fairness, Accountability, and Transparency*. 2254–2272.

[6] Wei Dai, Yi-Shan Tsai, Jionghao Lin, Ahmad Aldino, Hua Jin, Tongguang Li, Dragan Gašević, and Guanliang Chen. 2024. Assessing the Proficiency of Large Language Models in Automatic Feedback Generation: An Evaluation Study. *Computers and Education: Artificial Intelligence* (2024), 100299.

[7] Greta L. Doctoroff and David H. Arnold. 2017. Doing homework together: The relation between parenting strategies, child engagement, and achievement. *Journal of Applied Developmental Psychology* 48 (2017), 103–113.

[8] Dragan Gašević, Shane Dawson, and George Siemens. 2015. Let's not forget: Learning analytics are about learning. *TechTrends* 59 (2015), 64–71.

[9] Zifei FeiFei Han, Jionghao Lin, Ashish Gurung, Danielle Thomas, Eason Chen, Conrad Borchers, Shivang Gupta, and Ken Koedinger. 2024. Improving Assessment of Tutoring Practices using Retrieval-Augmented Generation. In *AI for Education: Bridging Innovation and Responsibility at the 38th AAAI Annual Conference on AI*.

[10] Kathleen V Hoover-Dempsey and Howard M Sandler. 1997. Why do parents become involved in their children's education? *Review of educational research* 67, 1 (1997), 3–42.

[11] Yun Huang, Nikki G Lobczowski, J Elizabeth Richey, Elizabeth A McLaughlin, Michael W Asher, Judith M Harackiewicz, Vincent Aleven, and Kenneth R Koedinger. 2021. A general multi-method approach to data-driven redesign of tutoring systems. In *LAK21: 11th International Learning Analytics and Knowledge Conference* (Irvine CA USA). ACM, New York, NY, USA.

[12] Carmel Kent, Benedict du Boulay, and Mutlu Cukurova. 2022. Keeping the parents outside the school gate—A critical review. *Educ. Sci. (Basel)* 12, 10 (Oct. 2022), 683.

[13] Meet Khanmigo. 2024. Khan Academy's AI-powered teaching assistant & tutor. *Khan Academy. URL: https://www.khanacademy.org/khan-labs [Accessed 2024-01-19]* (2024).

[14] Kenneth R Koedinger, Paulo F Carvalho, Ran Liu, and Elizabeth A McLaughlin. 2023. An astonishing regularity in student learning rate. *Proceedings of the National Academy of Sciences* 120, 13 (2023), e2221311120.

[15] Kenneth R Koedinger, Albert T Corbett, and Charles Perfetti. 2012. The Knowledge-Learning-Instruction framework: Bridging the science-practice chasm to enhance robust student learning. *Cognitive science* 36, 5 (2012), 757–798.

[16] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems* 33 (2020), 9459–9474.

[17] Jionghao Lin, Eason Chen, Zeifei Han, Ashish Gurung, Danielle R Thomas, Wei Tan, Ngoc Dang Nguyen, and Kenneth R Koedinger. 2024. How Can I Improve? Using GPT to Highlight the Desired and Undesired Parts of Open-ended Responses. *arXiv preprint arXiv:2405.00291* (2024).

[18] Fei Liu, Zejun Kang, and Xing Han. 2024. Optimizing RAG Techniques for Automotive Industry PDF Chatbots: A Case Study with Locally Deployed Ollama Models. *arXiv preprint arXiv:2408.05933* (2024).

[19] Xiner Liu, Maciej Pankiewicz, Tanvi Gupta, Zhongtian HUANG, and Ryan S BAKER. 2024. A Step Towards Adaptive Online Learning: Exploring the Role of GPT as Virtual Teaching Assistants in Online Education. *Manuscript under review* (2024).

[20] Leo S Lo. 2023. The CLEAR path: A framework for enhancing information literacy through prompt engineering. *The Journal of Academic Librarianship* 49, 4 (2023), 102720.

[21] Yanjin Long, Kenneth Holstein, and Vincent Aleven. 2018. What exactly do students learn when they practice equation solving? refining knowledge components with the additive factors model. In *Proceedings of the 8th International Conference on Learning Analytics and Knowledge*. 399–408.

[22] Qianou Ma, Hua Shen, Kenneth Koedinger, and Sherry Tongshuang Wu. 2024. How to Teach Programming in the AI Era? Using LLMs as a Teachable Agent for Debugging. In *International Conference on Artificial Intelligence in Education*. Springer, 265–279.

[23] Paula Magalhães, Diogo Ferreira, Jennifer Cunha, and Pedro Rosário. 2020. Online vs traditional homework: A systematic review on the benefits to students' performance. *Computers & Education* 152 (2020), 103869.

[24] Marijn Martens, Ralf De Wolf, and Lieven De Marez. 2024. Algorithmic Decision-Making and Education: The Acceptance of Learning Analytics by Secondary School Students and Parents. *Technology, Knowledge and Learning* (2024), 1–16.

[25] Alissa J Mrazek, Michael D Mrazek, Joshua R Ortega, Rosie R Ji, Sana S Karimi, Chelsea S Brown, Chelsie A Alexander, Maliha Khan, Rhozhen Panahi, Madeline Sadoff, et al. 2021. Teenagers' smartphone use during homework: an analysis of beliefs and behaviors around digital multitasking. *Education Sciences* 11, 11 (2021), 713.

[26] Atharva Naik, Jessica Ruhan Yin, Anusha Kamath, Qianou Ma, Sherry Tongshuang Wu, Charles Murray, Christopher Bogart, Majd Sakr, and Carolyn P Rose. 2024. Generating Situated Reflection Triggers About Alternative Solution Paths: A Case Study of Generative AI for Computer-Supported Collaborative Learning. In *International Conference on Artificial Intelligence in Education*. Springer, 46–59.

[27] Ha Tien Nguyen, Conrad Borchers, Meng Xia, and Vincent Aleven. 2024. Designing tools for caregiver involvement in intelligent tutoring systems for middle school mathematics. In *Proceedings of the 18th International Conference of the Learning Sciences - ICLS 2024*. International Society of the Learning Sciences.

[28] Andre Nickow, Philip Oreopoulos, and Vincent Quan. 2020. The Impressive Effects of Tutoring on Prek-12 Learning: A Systematic Review and Meta-Analysis of the Experimental Evidence. *Randomized Social Experiments eJournal* (2020). https://api.semanticscholar.org/CorpusID:220933901

[29] Benjamin D Nye, Dillon Mee, and Mark G Core. 2023. Generative Large Language Models for Dialog-Based Tutoring: An Early Consideration of Opportunities and Concerns.

[30] Sankalan Pal Chowdhury, Vilém Zouhar, and Mrinmaya Sachan. 2024. AutoTutor meets Large Language Models: A Language Model Tutor with Rich Pedagogy and Guardrails. In *Proceedings of the Eleventh ACM Conference on Learning@ Scale*. 5–15.

[31] Praise-El K Pelemo. 2022. *Parental Involvement: Investigating the Roles, Barriers, and Strategies of Effective Parent-Teacher Relationships in Middle School Math*. Ph. D. Dissertation. Southeastern University.

[32] Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine Mcleavey, and Ilya Sutskever. 2023. Robust Speech Recognition via Large-Scale Weak Supervision. In *Proceedings of the 40th International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 202)*, Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (Eds.). PMLR, 28492–28518.

[33] Alexander Renkl. 1999. Learning mathematics from worked-out examples: Analyzing and fostering self-explanations. *European Journal of Psychology of Education* 14, 4 (1999), 477–488.

[34] Lauren B Resnick, Christa SC Asterhan, Sherice N Clarke, et al. 2018. Accountable talk: Instructional dialogue that builds the mind. *Geneva, Switzerland: The International Academy of Education (IAE) and the International Bureau of Education (IBE) of the United Nations Educational, Scientific and Cultural Organization (UNESCO)* (2018).

[35] Benjamin Saunders, Julius Sim, Tom Kingstone, Shula Baker, Jackie Waterfield, Bernadette Bartlam, Heather Burroughs, and Clare Jinks. 2018. Saturation in qualitative research: exploring its conceptualization and operationalization. *Quality & quantity* 52 (2018), 1893–1907.

[36] Robin Schmucker, Meng Xia, Amos Azaria, and Tom Mitchell. 2024. Ruffle & Riley: Insights from Designing and Evaluating a Large Language Model-Based Conversational Tutoring System. In *International Conference on Artificial Intelligence in Education*. Springer, 75–90.

[37] Robert E Slavin. 1987. Making Chapter 1 make a difference. *The Phi Delta Kappan* 69, 2 (1987), 110–119.

[38] Jaemarie Solyst, Ellia Yang, Shixian Xie, Amy Ogan, Jessica Hammer, and Motahhare Eslami. 2023. The Potential of Diverse Youth as Stakeholders in Identifying and Mitigating Algorithmic Bias for a Future of Fairer AI. *Proceedings of the ACM on Human-Computer Interaction* 7, CSCW2 (Sept. 2023), 1–27.

[39] John Stamper, Ruiwei Xiao, and Xinying Hou. 2024. Enhancing LLM-based feedback: Insights from Intelligent Tutoring Systems and the learning sciences. (May 2024). arXiv:2405.04645 [cs.HC]

[40] Maxwell Szymanski, Jeroen Ooge, Robin De Croon, Vero Vanden Abeele, and Katrien Verbert. 2024. Feedback, Control, or Explanations? Supporting Teachers With Steerable Distractor-Generating AI. In *Proceedings of the 14th Learning Analytics and Knowledge Conference (LAK '24)*. ACM.

[41] Danielle Thomas, Xinyu Yang, Shivang Gupta, Adetunji Adeniran, Elizabeth Mclaughlin, and Kenneth Koedinger. 2023. When the Tutor Becomes the Student: Design and Evaluation of Efficient Scenario-based Lessons for Tutors. In *LAK23: 13th International Learning Analytics and Knowledge Conference*. 250–261.

[42] Danielle R Thomas, Jionghao Lin, Erin Gatz, Ashish Gurung, Shivang Gupta, Kole Norberg, Stephen E Fancsali, Vincent Aleven, Lee Branstetter, Emma Brunskill, et al. 2024. Improving student learning with hybrid human-AI tutoring: A three-study quasi-experimental investigation. In *Proceedings of the 14th Learning Analytics and Knowledge Conference*. 404–415.

[43] Shen Wang, Tianlong Xu, Hang Li, Chaoli Zhang, Joleen Liang, Jiliang Tang, Philip S Yu, and Qingsong Wen. 2024. Large language models for education: A survey and outlook. *arXiv preprint arXiv:2403.18105* (2024).

[44] Jules White, Quchen Fu, Sam Hays, Michael Sandborn, Carlos Olea, Henry Gilbert, Ashraf Elnashar, Jesse Spencer-Smith, and Douglas C Schmidt. 2023. A prompt pattern catalog to enhance prompt engineering with ChatGPT. (Feb. 2023). arXiv:2302.11382 [cs.SE]

[45] Harry F Wolcott. 1994. *Transforming qualitative data: Description, analysis, and interpretation*. Sage.

[46] Lixiang Yan, Roberto Martinez-Maldonado, and Dragan Gasevic. 2024. Generative artificial intelligence in learning analytics: Contextualising opportunities and challenges through the learning analytics cycle. In *Proceedings of the 14th Learning Analytics and Knowledge Conference*. 101–111.

[47] Jiayi Zhang, Conrad Borchers, Vincent Aleven, and Ryan S Baker. 2024. Using large language models to detect self-regulated learning in think-aloud protocols. In *Proceedings of the 17th International Conference on Educational Data Mining (EDM)*.

[48] Liang Zhang, Jionghao Lin, Conrad Borchers, Meng Cao, and Xiangen Hu. 2024. 3DG: a framework for using generative AI for handling sparse learner performance data from intelligent tutoring systems. *arXiv preprint arXiv:2402.01746* (2024).