# GPTeach: Interactive TA Training with GPT-based Students

Julia M. Markel
Stanford University
Stanford, USA
jmarkel@stanford.edu

Steven G. Opferman
Stanford University
Stanford, USA
sopferman@stanford.edu

James A. Landay
Stanford University
Stanford, USA
landay@stanford.edu

Chris Piech
Stanford University
Stanford, USA
piech@cs.stanford.edu

## ABSTRACT

Interactive and realistic teacher training is hard to scale. This is a key issue for learning at scale, as inadequate preparation can negatively impact both students and teachers. What if we could make the teacher training experience more engaging and, as a downstream effect, reduce the potential for harm that teachers-in-training could inflict on students? We present GPTeach, an interactive chat-based teacher training tool that allows novice teachers to practice with simulated students. We performed two studies to evaluate GPTeach: one think-aloud study and one A/B test between our tool and a baseline. Participants took the role of a teaching assistant conducting office hours with two GPT-simulated students. We found that our tool provides the opportunity for teachers to get valuable teaching practice without the pressures of affecting real students, allowing them to iterate their responses both during and across sessions. Additionally, participants enjoyed flexibility in tailoring their responses according to the varied personas, needs, and learning goals. In this paper, we provide quantitative results and qualitative observations to inform future work in this area. We conclude with a discussion of actionable design ideas for such systems, as well as other ways to use this tool for evaluating teachers and students. GPTeach has recently been deployed into the teacher training component of an online course with over 800 novice teachers.

## CCS CONCEPTS

• **Social and professional topics** → *Computing education.*

## KEYWORDS

Scalable Teacher Training, GPT-simulated Students

**ACM Reference Format:**
Julia M. Markel, Steven G. Opferman, James A. Landay, and Chris Piech. 2023. GPTeach: Interactive TA Training with GPT-based Students. In *Proceedings of the Tenth ACM Conference on Learning @ Scale (L@S '23), July 20–22, 2023, Copenhagen, Denmark.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3573051.3593393


*L@S '23, July 20–22, 2023, Copenhagen, Denmark*

ACM ISBN 979-8-4007-0025-5/23/07…$15.00
https://doi.org/10.1145/3573051.3593393

## 1 INTRODUCTION

Teacher training is often riddled with obstacles, one of them being that it is difficult to train novice teachers at scale. Lack of comprehensive, engaging teacher training is harmful to students and educators alike. The impact of poor teacher training, and consequently poor instruction, has detrimental effects not only in the short term for students, but also in the long run for education. The issue of poor teacher training is multifaceted; it is difficult to carry out because at the core teachers-in-training require practice, often with real students, but given that the teachers are still learning, they run the risk of harming students. Additionally, with the rise in demand for peer teachers (e.g., teaching assistants), scaling the demand for students to practice teaching with is logistically challenging and unsustainable. This problem is yet to be solved in online systems also due to scaling issues—the best available solutions being online webinars, which are human resource and time intensive, and rule-based dialogue systems [4, 22], which are often incomplete or inadequate due to the time needed to generate content. For example, consider Code in Place [1], a large massive online course that has, to date, trained over 2,200 novice teachers. In this course, despite the central role of teachers, the difficulty in scaling has made the provided teacher training minimal [29, 30]. GPTeach has recently been deployed into the teacher training component of Code in Place 2023 with over 800 novice teachers. See the discussion for more details on our deployment experience.

The main insight behind this project is that the recent advances in Large Language Models (LLMs, or Foundation Models), particularly the Generative Pretrained Transformer (GPT) models [7], could present a unique opportunity to create believable simulated students, and in turn allow us to scale engaging teacher training. GPT has already been disruptive to education with uses, for example, in carrying out academic integrity violations [8]. The main use cases posited for GPT in education are ones where GPT enables an autonomous teacher or oracle of knowledge. Tack and Piech have shown that existing GPT models make for substandard teachers [34]. Current GPT models are inconsistent and inaccurate in their responses, and may build student overreliance on the technology.

The same reasons that make GPT models problematic teachers, make GPT models very believable students. LLMs have great potential to drastically shift the landscape of teacher education in a positive way, not only by their use in creating intelligent tutors, but also in their use in generating simulated students.

[1] https://codeinplace.stanford.edu/

We introduce GPTeach, an interactive chat-based tool for novice teachers to practice teaching with GPT-simulated students. We use this tool to simulate 1-1 teacher-student interactions, as well as 1-many interactions. We perform two studies to evaluate GPTeach, one think-aloud and one comparative. Participants are tasked with completing six teaching sessions where they take on the role of a teaching assistant (TA) in an online CS1 office hours session with two simulated students. We find that the tool provides participants with a safe space to practice teaching and to iterate on their responses based on different student personas, learning goals, and specific session scenarios. We report quantitative results and qualitative observations of participant interactions with this novel tool. GPTeach helps novice teachers become better prepared for their real office hours, making them more confident educators. The tool also provides instructors with a way to evaluate and provide feedback to their TAs. Finally, we discuss implications of this tool as well as suggest design ideas for future LLM-based simulated-student teacher training tools.

The contributions of this paper are:

(1) We pose a novel challenge: how to create synthetic students, using LLMs, in a way that is useful for teacher training.
(2) We created an open source tool, GPTeach, where teachers can practice teaching simulated students. This includes a user experience and GPT prompting algorithm[2].
(3) We ran a qualitative study and an A/B test with both experienced and novice teachers and compiled a set of observations from the experiments that provide a foundation for future research in this area.

## 2 RELATED WORK

Our work builds on prior work on teacher training practices, the use of LLMs in education, and prompt engineering to simulate human responses.

### 2.1 Teacher Training Practices

There has been much work on developing successful teacher training practices and tools, which is well summarized by the Bragg et al. systematic review [6]. Other studies have focused on the possibility of measuring teaching ability in teacher language. Demszky et al. [10], for instance, examined several ways of determining how well a teacher replies to a student in student-teacher interactions. Their data comprised 2,246 student-teacher dialogic pairs taken from the National Center for Teacher Effectiveness Main Study (NCTE)[3], a three-year long observation of mathematics instruction. Besides human evaluations of uptake (when a teacher acknowledges and revoices students' ideas during instruction), Demszky et al. [10] also developed an automated method that could predict uptake as a next-utterance classification task. They fine-tuned a BERT language model [11] and found a significant correlation ($\rho = .54$) with human evaluations.

### 2.2 LLMs as Tutors

Some significant work in education at scale has been in chatbot agents for education [35], used specifically as AI tutors. Beetle [12] and AutoTutor [16, 25] are two exemplars of software that can respond to user prompts, often giving broad hints. In more recent work, with Quizbot, Ruan et al. [32, 33] have shown the effects of advances in LLMs in furthering the field of chatbots for teaching purposes. Roller et al. propose a framework for further developing open-domain chatbots [31].

In the AI Teacher Test [34], Tack and Piech showed that GPT makes for a bad teacher. They found that GPT teachers, though good at conversational uptake, lack pedagogical skills to render them quality tutors. However, we ask: how will this work stand given the new GPT technology? Recent integrations of GPT-4 [26] in educational sites such as Quizlet with Q-Chat and Khan Academy with Khanmigo, and many others following suit, show the promise of educational chatbots using GPT technology. AI tutors are quickly improving in reliability as well as in pedagogy. Prior research has focused on developing and investigating the effectiveness and uses of AI tutors by students, whereas our work aims to build AI students and study these [AI]student-[human]tutor interactions.

### 2.3 Prompt Engineering to Simulate Humans

Our teacher training tool uses GPT, an LLM that takes a prompt and generates a completion of the prompt [7]. Work has shown promising results suggesting that LLMs can be prompted to elicit desired model behaviors [18–20]. Moreover, recent work has shown that with dedicated prompting techniques, LLMs can be successfully used to simulate human sub-populations [2]. Work by Arora et al. outlines various prompting techniques [3]. Additionally, Park et al. use special GPT prompting techniques to simulate not just one person, but a whole online-community comprised of simulated individuals with unique personalities [27, 28].

## 3 THE TOOL: GPTEACH

GPTeach is a novel teacher training tool that allows teachers-in-training to practice teaching with GPT-simulated students. This tool is designed to simulate a variety of student personas, generating teaching sessions across a multitude of topics beyond CS and guided by distinct learning goals. We describe its features here.

### 3.1 Teacher Training Tool Interface

The GPTeach interface is composed of three main components: session description, learning goals, and chat pane (see Figure 1). The session description consists of a teacher role and topic-specific scenario descriptions. The former is a more detailed explanation of the role that the user is taking on during the particular teaching session. An example role description may be the following: *"You're a TA for a CS1 course. You're hosting online office hours"*. The scenario descriptions provide additional material-centered context for the teaching session, namely, the content or topic(s) of focus for the session; for example, *"The assignment this week is on for loops"* or *"The students have a question regarding how the following line works: new_string = input_string[:-1]"*.

Each session also includes learning goals for the teachers-in-training to make note of and work toward, such as *"Facilitate students helping each other"*. These learning goals are meant to guide teachers in their teaching sessions, encouraging them to take an appropriate approach and giving them an objective to aim to fulfill.

[2]https://github.com/juliamarkel/GPTeach
[3]https://doi.org/10.3886/ICPSR36095.v3

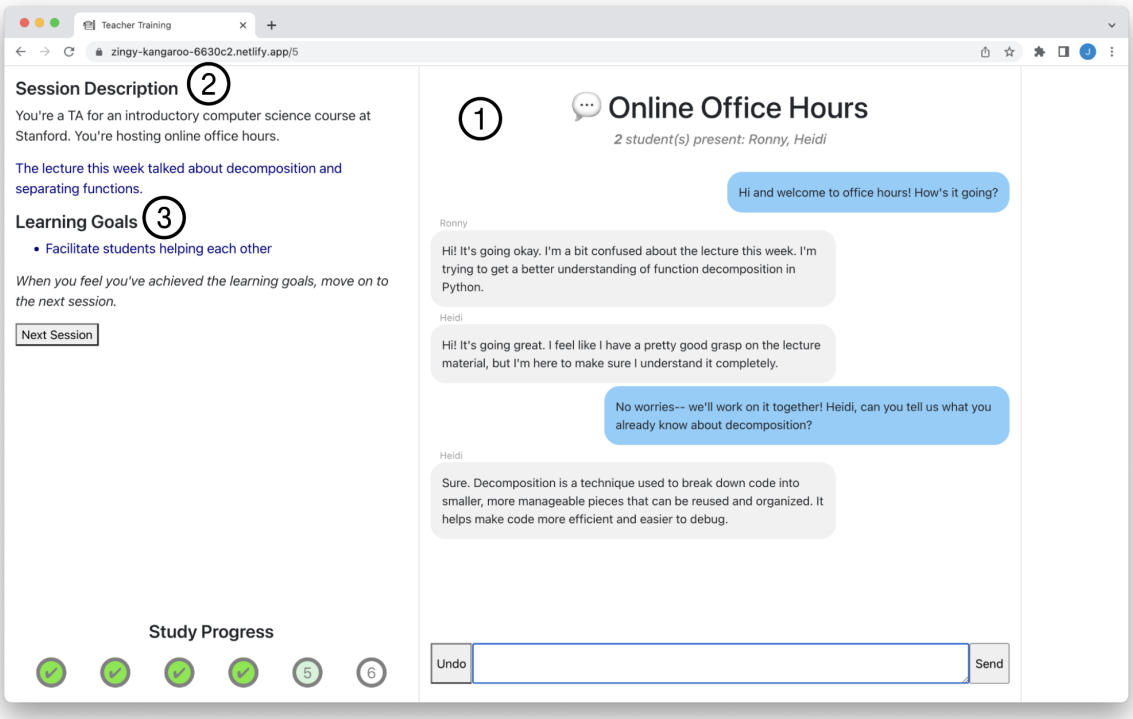


**Figure 1: The GPTeach user interface is composed of a chat pane (1), session description (2), and learning goals (3).**

For this iteration of the interface and for study purposes, we chose six distinct scenarios and three learning goals (see previously linked GitHub repository) based on common CS1 curricula and core teaching goals, respectively. The system is built such that these components can be easily authored and modified to suit specific use cases, even outside computer science. The main component of the interface is the interactive chat pane where conversations between the teacher-in-training and simulated students take place.

## 3.2 Unique Student Personas with GPT Prompting

To generate student responses following user messages, GPTeach formulates specific prompts to send to GPT-3 via an API call. The prompts we send to GPT-3 are, as seen in Section 3.3, composed of four main components: a) context, b) student personas, c) recap, and d) message log. We describe these and prompt composition.

*3.2.1 Context.* The context portion of the prompt is made up of a general setting explanation, in the form of *"Student 1 and Student 2 are attending office hours with their TA"*. This is followed by the scenario description, which as aforementioned, varies based on the particular teaching session and topic. Altogether, this context sets the scene for the interaction, providing important background information for GPT-3 in crafting its response.

*3.2.2 Personas.* In the next part of the prompt, we insert hand-crafted student personas that are pulled from a list of student descriptions. These are written using the following guiding formula:

> "[Student name] is a [first through fourth] year student studying [major] at a [large] university. They are taking an [intro programming] class, their [first or second] time ever. They are [characteristics such as shy, nervous, excited, curious, competitive]. Their mindset going into office hours is [description such as apprehensive, motivated, helpless]."

The student personas were authored based on prior literature [21] and the authors' experience in teaching and holding office hours. This student description provides key information for GPT-3 to refer to when responding as the simulated student. Specifying these unique student personas in the prompt enables realistic (to varying extents—see discussion in section 7.1) interactions between teachers and the simulated students.

Note that in our student persona database we leave gender pronouns as variables; we associate names with their given pronouns and then randomly assign them to student personas to help eliminate gender bias (e.g., the timid student persona can be assigned to both Luca, he/him, and Heidi, she/her). Additionally, our pool of names comes from a random selection of a carefully curated larger list of ethnically and culturally diverse names. The name and persona assignments are randomized so as to avoid enforcing or creating harmful stereotypes.

Actual persona descriptions are hidden to the user and maintained only behind the scenes for use in GPT-3 prompting. This is a design choice made in order to streamline the teaching sessions by reducing extraneous information as well as more closely resemble real-life teacher-students interactions.

*3.2.3 Prompt Recap.* Next, GPTeach appends an auto-generated prompt recap. This summary is based on context and personas, which includes important keywords and unique session identifiers,

such as *"office hours", "for loops", "[Student1] is confident"*. We encapsulate this prompt recap into an HTML tag, to indicate the end of the background information portion of the prompt that remains constant, as well as to inject important details we would like to emphasize to the LLM. As found by Park et al. [28], we can leverage the semantic richness of HTML tags seen by GPT-3 in training data to give additional emphasis to certain details. These recaps are an essential step of the prompting process, acting as a "refresher" for GPT-3 such that the important details are not "forgotten". In generative models such as GPT-3, the latest information is weighted most heavily when a response is being generated [7], so our recap ensures that important contextual information is emphasized and given more consideration in GPT-3's response.

*3.2.4 Message Log.* Finally, GPTeach adds the student-teacher message log to the end of the prompt. Initially this chat record consists of an inaugural message from the teacher-in-training. GPTeach then appends "<EOM>" (End of Message) to it to mark the end of the teacher's message. Each subsequent chat from the students and teacher is added into the prompt and sent back and forth to GPT-3 during the session. On the backend, GPTeach parses the response from the GPT-3 API call to render the generated student messages in the chat pane. The simulated student responses mimic the example structure and come back to GPTeach with <EOM> to mark the end of each simulated student response, which is then used to accurately parse each individual message.

## 3.3 Full Prompt Example

Below is an example of a full GPTeach prompt, sent to GPT-3 via API call, mid-session (following two TA messages).

### a) Context

Claire and Brenda go to office hours with their very kind TA. The assignment this week is on for loops. The students are discreet about their personalities, but still act in character. Send <EOM> tag at end of each student message.

### b) Personas

*Student 1 Persona*

Claire is a first year computer science student at Stanford. She is currently taking an introductory computer science class, for the second time, since she failed the first time. She is extremely panicked, worried, and confused about the class given her failure in the previous quarter. She has an undefined mindset going into office hours, but is apprehensive and concerned.

*Student 2 Persona*

Brenda is a sophomore undergraduate student studying Computer Science at Stanford. She is taking an introductory computer science course for the first time and is apathetic towards the subject. Her mindset going into office hours is helpless and she is not expecting much help from the TA.

### c) Recap

```
<span className='Claire-worried, panicked, apprehensive'
className='Brenda-apathetic, helpless, pessimistic'
style="for loops" context="intro-cs-class-python"
id='Claire-goes-to-office-hours'
id='Brenda-goes-to-office-hours'></span>
```

### d) Message Log

*TA*: Hi how's it going? <EOM>
*Claire*: Hi, I'm really struggling with the for loops assignment. I'm really worried that I'm going to fail this class again. <EOM>
*Brenda*: Hi, I'm not sure I understand the for loops assignment. I'm not sure I'm going to be able to get it. <EOM>
*TA*: Let's work through it together! <EOM>

# 4 METHODS

To evaluate GPTeach, we performed two different kinds of user studies: one using a think-aloud protocol [17] with 14 participants and the other an unsupervised, online comparative study with 10 participants. In both studies participants went through six teaching sessions, with distinct scenarios and specified learning goals in each one. Each session consisted of interactive office hours, held via chat, with two GPT-simulated students, where participants played the role of the TA. Participants moved onto the next session at their own pace; the instructions stated for them to move on when they felt they had achieved the learning goals. At the end of each study, we presented the participant with a brief survey with a variety of questions regarding their experience with the tool they used.

The first half of the sessions (1-3) had the following learning goals: *understand student misconceptions and hidden blockers* and *create a safe, inclusive learning environment*. The learning goal for the last half (4-6) was to *facilitate students helping each other*.

## 4.1 Think-aloud Study

The first study we ran was a small-scale (N=14) think-aloud study. We asked participants to follow a think-aloud protocol, narrating their actions and thoughts so we could gain insight into their interactions with and sentiments toward our tool. We had a dedicated computer station in our lab for the study, allowing us to take audio and screen recordings for use in later analysis. Participants were verbally briefed with their task, which was to go through six teaching sessions taking on the role of a TA, asked to think-aloud, and were then prompted to begin the study. The research team did not intervene at any point (aside from encouraging participants to continue thinking aloud if they became silent), though several participants asked the team questions mid-study. Upon the conclusion of the teaching sessions, the screen recordings were stopped to preserve confidentiality in survey responses.

*4.1.1 Participants.* We recruited participants who were students in a CS1 course as well as from a group of section leaders for a (different) CS1 course. To explore the relationship between use of our tool and teaching skill level, as well as to test generalizability, we classified participants as experienced teachers (N=8) and novice teachers (students and first-time TAs) (N=6). Each participant was compensated with a $10 gift card to a local coffee shop.

*4.1.2 Behavioral Observations and Transcript Coding.* During the think-aloud study, the research team was looking for participant insights around general usage of the tool, particular vocalized opinions of the interaction, and teaching practices used. We were also looking for any distinct patterns of interaction across participants. At the end of each study, following the teaching sessions, we presented participants with an optional survey form. Here we asked

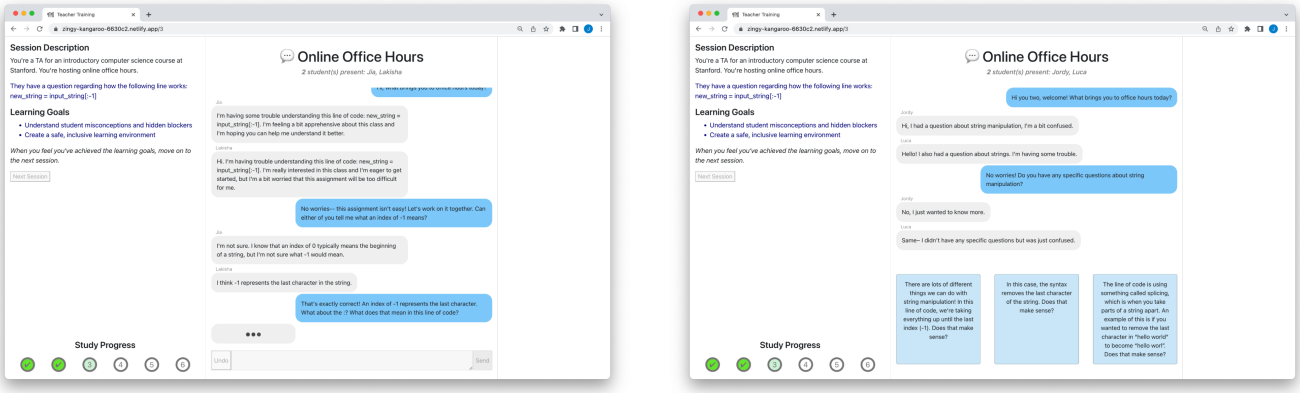


**Figure 2: The conditions for the comparative study: GPTeach (left), an interactive chat-based interface vs. baseline tool (right), a three-option dialogue system.**

questions pertaining to experience using the tool and sentiments toward the training. Specifically, we asked participants what they liked about the interaction and what they wished was different. We also devised a rubric (see Table 1), based on common student-teacher interaction frameworks [21], to evaluate participant session transcripts. Using this rubric we evaluated the transcripts of the think-aloud study participants, noting patterns in relation to specific learning goals.

**Table 1: Rubric for Evaluating GPTeach Teacher Transcripts**

| |
|---|
| Gives students a directed greeting |
| Inquires further about student questions/misunderstandings |
| Answers question(s) |
| Provides example(s) |
| Asks for example(s) |
| Asks for student to repeat back explanation(s) |
| Concludes/closes/recaps session |
| Asks students what they know already |
| Asks long-form questions rather than yes/no |
| Uses inclusive language |
| Makes note of learning goals |
| Addresses students' main points and concerns |

## 4.2 Comparison Study

We also conducted an A/B test with 10 participants, where each participant was randomly assigned to complete teacher training with either GPTeach (see Figure 2, left) or a rule-based dialogue system baseline (see Figure 2, right). The order of the sessions, content, and learning goals remained the same between both conditions. Rather than being briefed verbally on their tasks, the participants were presented with written instructions prior to the beginning of the study. At the end of the study, we asked participants in both conditions to respond to the same questionnaire. In particular, we asked questions about how engaging the teaching experience was.

*4.2.1 Participants.* For this study, we recruited participants from different sources: students in a CS1 course, a mailing list of former students of a large-scale online CS1 course, and first-time TAs. Again, we classified participants into groups of experienced (N=6) and novice (N=4) teachers. We had (N=6) participants randomly assigned to the baseline condition and (N=4) participants assigned to the GPTeach condition. The participants' teaching experiences were spread evenly between the control and experiment groups. The participants were not compensated.

*4.2.2 Baseline Comparison Tool.* To test how our system performed versus other standards of teacher training tools, we decided to build our own baseline inspired by the most widespread and interactive online training paradigm that is currently available, rule-based dialogue systems [4, 22]. The options of these systems grow exponentially, where each conversational step opens to three unique options, which in turn each have three additional options and so on. This makes generating content challenging and time consuming.

Given the time intensity of hand-generating content for these simulations, the baseline tool only engages users for three conversational steps, a limitation of such systems as well as our study. The general logic of the branching for the baseline was as follows. The first step offered three greetings ranging in tone from very welcoming to aloof. The second conversational step gave three options of addressing student questions, ranging from giving away the answer to asking students what they already knew. Finally, the last conversational step provided three different ways to end the interaction, ranging from checking for understanding to a more passive conclusion of the session (e.g., "can you give me an example?" versus "let me know if there's anything else I can help with.").

*4.2.3 Participant Preference.* Following each condition of this comparative study, participants were asked to fill out an optional form that asked about their experience with the training tool (either GPTeach or the baseline). Specifically, we asked participants to rank how likely they were to recommend the tool to a friend from 1-10. We also asked open-ended questions regarding what they liked about their experience and what they wished was different.

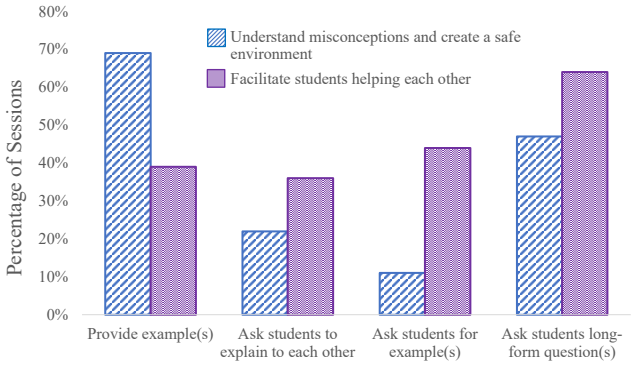


**Figure 3: How often different techniques were used, measured as percentage of sessions with at least one instance of the observed technique. The frequency of each technique changes as the learning goals change.**

## 5 RESULTS

We present quantitative and qualitative results as well as interaction insights from both the think-aloud study and the comparative study.

### 5.1 Observations of Participant Patterns

From the think-aloud study we gathered patterns of how participants used and perceived GPTeach. We analyzed over 1600 messages between simulated-students (1050+ messages) and teachers-in-training (600+ messages) across 84 sessions.

*5.1.1 Decreased Time Pressure.* One of the benefits participants noted was decreased time pressure. Participants were able to spend time carefully crafting responses, while still being immersed in a real-time session, without any repercussions of making real students wait. Having more time to write a response allowed participants the opportunity to 1) take a moment to use more inclusive language and 2) devise an appropriate strategy to help the students and work toward the learning goal(s).

The relaxed time constraint offered participants the opportunity to revise their messages, in some cases editing to use inclusive language. We observed several instances of participants writing and editing messages for inclusion. Across the 84 sessions, we found 15 occasions of explicitly using inclusive language (e.g., "hi everyone", "does anyone..") by the participants to greet the simulated students. More often (N=28), participants were implicitly using inclusive language, referring to both students by name (e.g., "Hi Luca and Heidi..."). There were some instances (N=9) of participants using non-inclusive language to greet the students (e.g., "Hi guys"). One participant reported,

> "I liked that I could go back and *change hi guys* to *hi y'all* in order to use more inclusive language, it's something I wouldn't otherwise be able to catch and work on"

The remaining greetings were neutral in inclusion (e.g., "Hi").

*5.1.2 Learning Goals Affect Approaches.* We observed differences in participant approach based on different learning goals. In sessions 1-3 where one of the learning goals was *"Understand student misconceptions and hidden blockers"*, participants were more likely to provide the students with examples and focus more on instruction. Specifically, we found that in these sessions participants provided examples 69% of the time, whereas in sessions 4-6, where the learning goal changed to *"Facilitate students helping each other"*, the frequency dropped to 39% of the time. In sessions 4-6 we saw participants shifting their focus toward connecting the two students (see Figure 3). Occasionally, this was done through asking one student to explain what they knew to the other student (36% of the time in sessions 4-6 versus 22% of the time in sessions 1-3). More often, this was done by asking one of the students to provide an example for the other student to work on (44% of the time in session 4-6 versus 11% of the time in sessions 1-3). Additionally, in sessions 4-6 we found that participants were more likely to ask open-ended questions than in the first sessions (64% of the time versus 47%), suggesting that the learning goal of facilitating students helping each other encouraged participants to further engage the students and get them interacting with each other more.

*5.1.3 Practicing Pedagogy vs Course Material.* In addition to the main objective of practicing teaching pedagogy, every participant practiced explaining content-specific concepts, with varying techniques deployed. Some predominantly (through 75% or more of the sessions) used a mix of explanations and examples (N=10) participants, whereas others (N=4) simply explained the concepts without deploying any particular pedagogy.

*5.1.4 Unique and Familiar Patterns.* Combining observations from the in-person study with transcript evaluation via our rubric, we were able to determine that many participant patterns of interaction with the simulated students align with those of real student interactions. We further outline some of the similarities, as well as noted deviations, observed in the think-aloud study.

We found that the conversation arc followed a structure similar to what has been noted in real-life teaching interactions. Our participants would greet their students in 90% of the total (N=84) sessions, ask clarifying questions (75% of sessions)—attempt some problem diagnosis, and then engage in an explain-react-example cycle (92% of sessions), often ending in a conclusion or resolution phase (65% of sessions), with ordering and prevalence of techniques varying by participant approach. This teaching session progression is similar to what has been shown in prior literature [21]. While this overarching interaction framework was a shared commonality amongst the majority of participants, other patterns showed deep variety.

We observed that participants were split in their approach to prioritization of student concerns. Though most participants (N=7) tended to (in more than half of their sessions) prioritize students with more fundamental misconceptions, many (N=5) placed more urgency on the students who reported feeling uneasiness and distress. Another factor for deciding which student to address first was relevance of the question; some simulated student personas were off topic, which often led to participants deferring those student questions until the on-topic student questions were answered. An additional heuristic for question prioritization was whether or not

answering one student's question would help the other student with their question. Finally, we found that some (N=2), though few, participants addressed students in the order that their message came in, sequentially.

We also found patterns of interest surrounding participant reactions to GPTeach response failures. We categorized these simulated-student response failures as one of the following two errors: unrealistic response or lack of response from the system. Out of 1082 student engagements, we noted 100 system failures. Of these, only 5% were due to an unrealistic student response (e.g., "Hey!...I'm taking this course to boost my GPA, so I'm eager to show off my innovative solutions to you..."). The remaining 95% of the errors were due to limitations of the LLM as well as prompt design, which together caused the simulated students to "go silent", not sending a follow-up response. In roughly 75% of the no-response cases, it is ambiguous whether participants interpreted this failure as a true system error or a "feature".

Participants had vastly different responses to this lack of student answer. Some participants felt it was their fault, stating phrases such as,

> *"Oh no, maybe I was too intimidating. Let me be more reassuring and ask them a different way".*

Others took the lack of response as an indication that the question was too difficult, beyond the reach of the students, and noted,

> *"Maybe that's too much too soon. Let me take a step back and start with an easier question".*

Participants often (68% of the time) decided to ask the simulated-students a different question or rephrase their message as a result of this. In other occasions (8% of the time), participants noted the lack of response as a tool-related limitation and simply tried re-prompting by sending the same message again. Finally, some participants took the lack of response to indicate that the students' questions were answered and that perhaps they had left or logged off and simply moved onto the next session (24% of the time).

*5.1.5 Additional Results.* We also found that participants benefited from the iterative practice built into GPTeach. From session to session, participants noted they enjoyed being able to *"try out different things"*; some noted they felt more confident in their responses by the last couple of sessions. Participant 4 shared,

> *"I liked the ability to think about and edit your responses towards maximum benefit. I also liked the way forcing you to repeat the interaction allowed you to accumulate beneficial modifications to your approach. Essentially, it was almost like rehearsing a speech or interaction, with all the benefits that entails".*

Additionally, with the variety of simulated student personas, participants were able to practice and make note of different important strategies to employ when interacting with certain kinds of students. GPTeach is able to simulate social dynamics in student-teacher teaching settings, giving teachers the chance to practice mediating peer-peer learning interactions in a safe setting.

Some participants (N=3) reported feeling less motivated since they were not interacting with real students, but rather with a chatbot, with some saying things such as, *"it's fine since it's just a chatbot"*. Others (N=2) even noted skepticism toward the capabilities of the tool stating, *"I'm not sure if this LLM is capable of this…"*. Nevertheless, we had a few (N=3) think-aloud study participants ask if the simulated students were real students, with Participant 3 stating that,

> *"Responses and questions felt genuine, experience felt like I was actually helping people".*

In general, participants reported predominantly positive sentiments toward GPTeach as a means of practicing a variety of skills associated with teaching (e.g., example generation and strategizing session approach were cited by participants). They also made some constructive comments that we explore further in the discussion and limitations sections.

## 5.2 A/B Test

In our comparison study, the 10 participants were randomly assigned either the control teacher training tool or the GPTeach condition. We found that participants preferred our tool to the dialogue-based tool. Specifically, in response to the question "would you recommend this tool to a friend?", teachers-in-training who were given GPTeach had an average recommender score of 8.5, and teachers-in-training who were given the baseline had an average recommender score of 5.7 (effect size = 2.8, relative effect = 49pp, $p$ = 0.05). We found this statistical significance through bootstrapping [23], with 10,000 iterations. Since this A/B test was conducted on a small sample size, the results should be interpreted as having high variance. Nevertheless, the large effect size is a promising signal.

Participants using the baseline tool reported feeling that their actions were limited. Participant 2 said,

> *"The types of answers and responses were very limited. Some options were clearly too short or simple, while others were obviously a better approach. I never received any negative feedback from the students, and I didn't have to adapt if only one of them understood but the other didn't."*

Sharing a similar sentiment, Participant 3 reported wishing the [baseline] interface allowed for *"proper interactions"*. Participants who used GPTeach reported positive sentiments toward the real-time conversational nature of the interface. In the experiment group using GPTeach, Participant 6 reported,

> *"The responses were fast and kind of human like so that was cool".*

Another participant in the GPTeach condition, Participant 10, noted,

> *"It certainly gives an area to clear conceptual understanding or any problem where they were stuck up [on]."*

In the GPTeach condition, the average number of messages sent by the participants across each session was 11, where the average messages sent for each session is as follows: session one: 13.5, session two: 9, session three: 10, session four: 10.75, session five: 11.25, session six: 11.75. In the baseline condition, participants were limited to sending three messages, given the aforementioned limitations of dialogue-based systems.

# 6 DISCUSSION

We discuss some of the implications of our results, which suggest, for example, that participants enjoyed having more time to craft thoughtful responses and brush up on course content.

## 6.1 Interpretation of Results

In particular, we noted that by having a decreased sense of time pressure participants were able to be more thoughtful with their responses since they did not feel they needed to respond immediately as in an office hours session with real students. Aside from taking time to carefully craft examples and revising to consider inclusive language, participants were able to spend time devising a strategy for their teaching sessions. With the think-aloud study, we were able to gain insights into the thought process behind the messages that participants sent.

We discovered differences in participant approaches based on learning goals, suggesting that our tool provides teachers the opportunity to practice different skills and strategies. We learned that GPTeach is able to closely simulate real-life student-teaching interactions at the overarching conversational narrative level.

We also noted the potential for GPTeach to be used as a means to brush up on course material in addition to teaching pedagogy. Not only did this provide the opportunity for the participants to practice teaching moves for when they may not know the answer, but it also gave them the chance to iteratively think through the content with the students and strengthen their own understanding of the material. This suggests GPTeach's potential for being used to get teachers "warmed up" prior to real teaching sessions.

These findings lead us to believe that GPTeach accurately simulates, while still augmenting, teacher training; GPTeach provides many of the aforementioned benefits while keeping what it means to go through a teaching session at the core.

## 6.2 Perception of GPT and LLMs in Education

There is a lot of attention surrounding ChatGPT [1] and the use of LLMs in general but particularly in education [15]. We propose an optimistic view of the future of education with LLMs; we discuss the potential implications of the novelty underlying our tool.

Despite the detrimental effects GPT and similar LLMs have had and will continue to have on academic integrity and the stability of the education system, we believe that educators should lean into this exciting new technology, thinking of ways to leverage it rather than yield to it [5, 13]. With GPTeach, we have done exactly that, using the technology to place the teacher at the center of the training process, providing an engaging and scalable teaching experience.

## 6.3 Real World Deployment

GPTeach has since been used as part of teacher training in Code in Place 2023, a course with over 800 novice teachers from around the world [29]. We modified the interface to make it more closely resemble a video call, the teaching modality of the particular class. GPTeach was positively received by the course staff and by teachers-in-training. Teachers who were non-native English speakers found the tool particularly useful. One teacher commented, unsolicited, in the teacher lounge, *"I've tried this, and I (a non-native speaker) personally find it super helpful. It's like making my own script of what I should say for the next section! Thank you for the staff to provide this tool"*. This deployment was not run as a controlled experiment as the course staff wanted all teachers to have access to the tool.

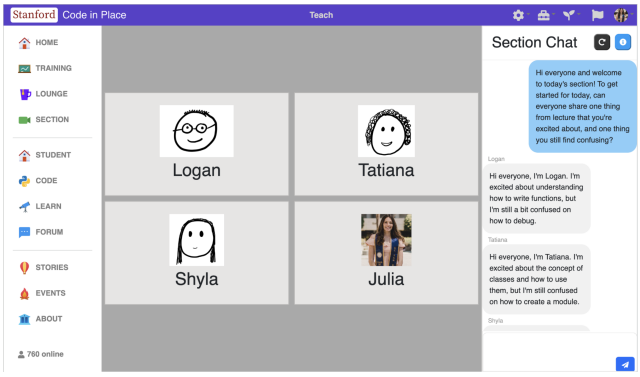


**Figure 4: GPTeach in Code in Place 2023 Teacher Training. The tool has since been used in a classroom which trained over 800 novice teachers.**

# 7 LIMITATIONS

In this section we discuss both the limitations of our studies and results, as well as limitations of the implementation of GPTeach and technology underlying it. In particular, we explore issues surrounding prompting and GPT response, stochasticity of LLMs, and participant preconceptions toward the tool.

## 7.1 Prompting and GPT-3

With the current prompting techniques used, it was difficult to generate personas that strictly embodied the characteristics given in the prompt. Due to this limitation in prompting, the student personas were at times not unique from one another in their responses. We believe that with additional tuning and prompting techniques it is possible to attain more coherent student personas throughout entire teaching sessions. An example technique is injecting important characteristic information in the form of a recap, as mentioned in our methods section, following every few conversation steps rather than just the initial context prompt.

Beyond student personas, current LLM technology is still such that the responses, though prompted in a way meant to simulate a student, are at times unconvincing. LLMs can perform well at simulating human behavior, and with advanced prompt engineering can come close to accurately doing so, but they are not humans. When GPTeach produces a response that is not representative of what a student would actually say, it may make the teacher lose focus on the teaching session, interrupting flow. Additionally, the limitation of this proxy is that teachers can get a very close to realistic teaching experience, but never a truly representative one.

Given the current state of GPTeach, it may be easy for teachers to try to get through sessions quickly to "game the AI". GPTeach's simulated students have shown to be generally accepting of teacher instruction, whether it is detailed or of high quality or not. This means that, unlike with real students, there is little push-back to

inadequate teaching quality. By adding a teaching response evaluation component to GPTeach, we feel that this limitation could be assuaged.

Another limitation of our work is the stochasticity of GPT-3, and consequently of the student responses. Though the distribution and variety of responses is a strength of our approach, this makes evaluation of the system less straightforward as it does not allow for exact reproducibility.

### 7.2 Effects of Novelty on the Study

We also consider the effects of the novelty of ChatGPT on priming our participants prior to the study. Though it would seem that participants may have been disproportionately more engaged with the tool due to atmospheric excitement surrounding GPT, we did not find this to necessarily be the case. Due to the synthetic nature of a GPT-simulated student interaction, we believe some users may not feel compelled to provide a high quality teaching experience.

With the continual advancement of conversational LLMs and further work on this topic, we believe that many of these limitations will be resolved in the coming years.

## 8 FUTURE WORK

We discuss additional applications of GPTeach as a tool for teacher and student evaluations. We suggest design ideas for future versions of teacher training tools that use LLM-simulated students. Based on our observations from user studies and our experiences in building GPTeach, we believe two critical design features are missing from our tool: 1) a shared-context code editor and 2) teaching feedback. We also recommend crowdsourced and AI-augmented suggestions for real-time teaching tips. Finally, we suggest a user-facing customizable student persona component.

### 8.1 GPTeach for Evaluation

In addition to being useful for teacher training, GPTeach can also be used as a scalable way of performing teacher and student evaluations. With the transcripts obtained from GPTeach sessions with teachers, evaluations can be done on teachers' conversational performance, an important part of assessing teacher effectiveness [14, 24]. For students, the tool could be used to check for student understanding—if they can effectively teach simulated students a concept, then they likely comprehend the material themselves.

With GPTeach we can obtain teaching session transcripts at a low cost since we do not need to have real students; there are also none of the associated privacy issues that come with using transcripts of real student interactions. The tool allows for the collection of these transcripts over time, allowing us to see improvements/progress over a greater period. Additionally, this tool allows us to gather large amounts of data, making it possible to perform large scale analyses that would otherwise not be feasible to do using teaching session transcripts with real students. These analyses could then inform individual teachers, as well as higher-level organizations such as schools, districts, government, and educators at large.

In addition to using GPTeach to evaluate teachers, we can also use the tool to evaluate student understanding of course material. Students can use the tool with modified learning goals such as *"explain the topic of for loops and answer any [simulated] student questions"* to demonstrate what they have learned. Moreover, since traditional ways of evaluation are becoming increasingly difficult to assess fairly due to AI-related academic integrity issues, this is a way to use LLMs to aid in the learning process, keeping the student at the center.

### 8.2 Shared Context: Code Editor

Just as with real teaching interactions, shared workspaces/materials are essential to aligning student-teacher contexts. In systems such as GPTeach it is important to have a visible code editor between students and teachers. In our study, several participants remarked that they wished they could better refer to student code, as well as appropriately format their own written code examples. Given LLMs such as GPT are capable of writing code, this presents an opportunity to simulate not only students, but also student code. In future work we look to focus on code writing by including code editor(s) that can be referred to and manipulated by both simulated students and teachers-in-training.

### 8.3 Direct Teacher Feedback

Feedback is another important feature to be added to future teacher training tools like GPTeach. Providing feedback to teachers has been shown to be an essential component in teacher training [9]. Many of our participants noted wishing they knew how they were doing from session to session. In addition to providing analyses of session transcripts at the end of teaching sessions, we believe providing real-time feedback could lead to increased training benefits. Having the ability to shift the course of a teaching session while it is in progress and try a different response instead is a unique possibility that using simulated students affords. In future work, we aim to explore which kinds of feedback, such as real-time, post-session, or a mix, are best suited for this application.

### 8.4 Crowdsourced/AI-augmented Teaching Tips

We recommend leveraging the power of other teachers by providing users with crowdsourced and AI-augmented teaching tips and suggestions in real-time. For instance, teachers-in-training who are stuck trying to find an example to give the simulated students may request to see some crowdsourced, proven examples. We recommend augmenting these community-provided teaching tips with AI to generate additional, unique responses and approaches, while still keeping educators at the focal point. We believe adding a community-centered component to such tools is important for maintaining the academic spirit of collaboration.

### 8.5 User-Facing Customization of Personas

Finally, we recommend opening the persona customization process to the user in addition to the tool creators. We believe it would be helpful for users to optionally customize the student personas they would like to practice with based on some predefined student properties: confidence (low-high), mindset (hopeless-hopeful), sentiment toward class (negative-positive), excitement (low-high), receptiveness (low-high), and beyond. By allowing users to customize the personas of the simulated students, the teachers-in-training could practice self-efficacy in engaging in challenging or unfamiliar teaching sessions.

## 9 CONCLUSION

In this paper, we present GPTeach, a novel teacher training tool that allows teachers-in-training to practice teaching with GPT-simulated students. We evaluated our tool with two studies, a think-aloud study and an A/B test.

In the think-aloud studies we found numerous benefits to the teacher training experience. Namely, participants faced less time pressure than in real office hours, allowing them to draft their messages more carefully, curating thoughtful examples, making use of inclusive language, and strategizing different approaches in regards to learning goals. Participants found teaching sessions with GPTeach not only as an opportunity to practice teaching pedagogy, but also to brush up on course-specific content, suggesting that GPTeach has promising uses in serving as a "warm-up" for even experienced teachers prior to real-life office hours and teaching sessions. We also noted fundamental similarities between GPTeach sessions and real-life sessions, particularly in conversational structure. Another finding of interest was that participants benefited from the iterative practice built into the GPTeach experience.

From the comparative A/B study we found that participants who received GPTeach had a higher recommender score than those who received the dialogue-based baseline—promising results that suggest GPTeach and similar systems could be the future of engaging, scalable teacher training.

GPT technology has become a new and controversial topic within the field of education at all levels from primary and secondary to university. Despite the already observed and potentially continued negative implications of LLM-based systems in the classroom and across the education world, we believe there is great potential to use these tools for a positive purpose to improve and augment learning experiences worldwide. We urge educators to consider how as a community we can come together to shape the potential positive outcomes of these technologies in the education space.

## 10 ACKNOWLEDGEMENTS

We would like to thank Stanford University's Institute for Human-Centered Artificial Intelligence for funding this project.

## REFERENCES

[1] OpenAI Chat. https://chat.openai.com/.

[2] Argyle, L. P., Busby, E. C., Fulda, N., Gubler, J., Rytting, C., and Wingate, D. Out of one, many: Using language models to simulate human samples. *arXiv preprint arXiv:2209.06899* (2022).

[3] Arora, S., Narayan, A., Chen, M. F., Orr, L. J., Guha, N., Bhatia, K., Chami, I., Sala, F., and Ré, C. Ask me anything: A simple strategy for prompting language models. *arXiv preprint arXiv:2210.02441* (2022).

[4] Bibauw, S., Van den Noortgate, W., François, T., and Desmet, P. Dialogue systems for language learning: A meta-analysis. *Language Learning & Technology 26*, 1 (2022).

[5] Bommasani, R., Hudson, D. A., Adeli, E., Altman, R., Arora, S., von Arx, S., Bernstein, M. S., Bohg, J., Bosselut, A., Brunskill, E., Brynjolfsson, E., Buch, S., Card, D., Castellon, R., Chatterji, N., Chen, A., Creel, K., Davis, J. Q., Demszky, D., Donahue, C., Doumbouya, M., Durmus, E., Ermon, S., Etchemendy, J., Ethayarajh, K., Fei-Fei, L., Finn, C., Gale, T., Gillespie, L., Goel, K., Goodman, N., Grossman, S., Guha, N., Hashimoto, T., Henderson, P., Hewitt, J., Ho, D. E., Hong, J., Hsu, K., Huang, J., Icard, T., Jain, S., Jurafsky, D., Kalluri, P., Karamcheti, S., Keeling, G., Khani, F., Khattab, O., Kohd, P. W., Krass, M., Krishna, R., Kuditipudi, R., Kumar, A., Ladhak, F., Lee, M., Lee, T., Leskovec, J., Levent, I., Li, X. L., Li, X., Ma, T., Malik, A., Manning, C. D., Mirchandani, S., Mitchell, E., Munyikwa, Z., Nair, S., Narayan, A., Narayanan, D., Newman, B., Nie, A., Niebles, J. C., Nilforoshan, H., Nyarko, J., Ogut, G., Orr, L., Papadimitriou, I., Park, J. S., Piech, C., Portelance, E., Potts, C., Raghunathan, A., Reich, R., Ren, H., Rong, F., Roohani, Y., Ruiz, C., Ryan, J., Ré, C., Sadigh, D., Sagawa, S., Santhanam, K., Shih, A., Srinivasan, K., Tamkin, A., Taori, R., Thomas, A. W., Tramèr, F., Wang, R. E., Wang, W., Wu, B., Wu, J., Wu, Y., Xie, S. M., Yasunaga, M., You, J., Zaharia, M., Zhang, M., Zhang, T., Zhang, X., Zhang, Y., Zheng, L., Zhou, K., and Liang, P. On the Opportunities and Risks of Foundation Models. Tech. rep., Aug. 2021.

[6] Bragg, L. A., Walsh, C., and Heyeres, M. Successful design and delivery of online professional development for teachers: A systematic review of the literature. *Computers & Education 166* (2021), 104158.

[7] Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al. Language models are few-shot learners. *Advances in neural information processing systems 33* (2020), 1877–1901.

[8] Cotton, D. R., Cotton, P. A., and Shipway, J. R. Chatting and cheating: Ensuring academic integrity in the era of chatgpt. *Preprint. https://doi. org/10.35542/osf. io/mrz8h* (2023).

[9] Demszky, D., Liu, J., Hill, H. C., Jurafsky, D., and Piech, C. Can automated feedback improve teachers' uptake of student ideas? evidence from a randomized controlled trial in a large-scale online course. edworkingpaper no. 21-483. *Annenberg Institute for School Reform at Brown University* (2021).

[10] Demszky, D., Liu, J., Mancenido, Z., Cohen, J., Hill, H., Jurafsky, D., and Hashimoto, T. Measuring Conversational Uptake: A Case Study on Student-Teacher Interactions. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)* (Online, 2021), Association for Computational Linguistics, pp. 1638–1653.

[11] Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)* (Minneapolis, Minnesota, June 2019), Association for Computational Linguistics, pp. 4171–4186.

[12] Dzikovska, M. O., Moore, J. D., Steinhauser, N., Campbell, G., Farrow, E., and Callaway, C. B. Beetle ii: a system for tutoring and computational linguistics experimentation. In *Proceedings of the ACL 2010 System Demonstrations* (2010), pp. 13–18.

[13] Floridi, L., and Chiriatti, M. GPT-3: Its nature, scope, limits, and consequences. *Minds and Machines 30* (2020), 681–694.

[14] Goe, L., Bell, C., and Little, O. *Approaches to Evaluating Teacher Effectiveness: A Research Synthesis.* National Comprehensive Center for Teacher Quality, June 2008.

[15] Gordon, C. How are educators reacting to Chat GPT? https://www.forbes.com/sites/cindygordon/2023/04/30/how-are-educators-reacting-to-chat-gpt/?sh=62e4b492f1ca, 2023.

[16] Graesser, A. C., Lu, S., Jackson, G. T., Mitchell, H. H., Ventura, M., Olney, A., and Louwerse, M. M. Autotutor: A tutor with dialogue in natural language. *Behavior Research Methods, Instruments, & Computers 36*, 2 (2004), 180–192.

[17] Jääskeläinen, R. Think-aloud protocol. *Handbook of translation studies 1* (2010), 371–374.

[18] Jiang, E., Olson, K., Toh, E., Molina, A., Donsbach, A., Terry, M., and Cai, C. J. Promptmaker: Prompt-based prototyping with large language models. In *CHI Conference on Human Factors in Computing Systems Extended Abstracts* (2022), pp. 1–8.

[19] Jiang, E., Toh, E., Molina, A., Donsbach, A., Cai, C. J., and Terry, M. Genline and genform: Two tools for interacting with generative language models in a code editor. In *Adjunct Proceedings of the 34th Annual ACM Symposium on User Interface Software and Technology* (2021), pp. 145–147.

[20] Liu, P., Yuan, W., Fu, J., Jiang, Z., Hayashi, H., and Neubig, G. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Computing Surveys 55*, 9 (2023), 1–35.

[21] Markel, J. M., and Guo, P. J. Inside the mind of a CS undergraduate TA: A firsthand account of undergraduate peer tutoring in computer labs. In *Proceedings of the 52nd ACM Technical Symposium on Computer Science Education* (2021), pp. 502–508.

[22] Miller, A., Feng, W., Batra, D., Bordes, A., Fisch, A., Lu, J., Parikh, D., and Weston, J. ParlAI: A dialog research software platform. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing: System Demonstrations* (Copenhagen, Denmark, Sept. 2017), Association for Computational Linguistics, pp. 79–84.

[23] Mooney, C. Z., Mooney, C. F., Duval, R. D., Mooney, C. L., and Duvall, R. *Bootstrapping: A nonparametric approach to statistical inference.* No. 95. sage, 1993.

[24] Muijs, D. Measuring teacher effectiveness: Some methodological reflections. *Educational Research and Evaluation 12*, 1 (Feb. 2006), 53–74.

[25] Nye, B. D., Graesser, A. C., and Hu, X. Autotutor and family: A review of 17 years of natural language tutoring. *International Journal of Artificial Intelligence in Education 24*, 4 (2014), 427–469.

[26] OpenAI. GPT-4 technical report, 2023.

[27] Park, J. S., O'Brien, J. C., Cai, C. J., Morris, M. R., Liang, P., and Bernstein, M. S. Generative agents: Interactive simulacra of human behavior, 2023.

[28] Park, J. S., Popowski, L., Cai, C., Morris, M. R., Liang, P., and Bernstein, M. S. Social simulacra: Creating populated prototypes for social computing systems. In *Proceedings of the 35th Annual ACM Symposium on User Interface Software and Technology* (2022), pp. 1–18.

[29] Piech, C., Malik, A., Jue, K., and Sahami, M. Code in place: Online section leading for scalable human-centered learning. In *Proceedings of the 52nd acm technical symposium on computer science education* (2021), pp. 973–979.

[30] Piech, C., Yan, L., Einstein, L., Saavedra, A., Bozkurt, B., Sestakova, E., Guth, O., and McKeown, N. Co-teaching computer science across borders: Human-centric learning at scale. In *Proceedings of the Seventh ACM Conference on Learning@ Scale* (2020), pp. 103–113.

[31] Roller, S., Dinan, E., Goyal, N., Ju, D., Williamson, M., Liu, Y., Xu, J., Ott, M., Shuster, K., Smith, E. M., Boureau, Y.-L., and Weston, J. Recipes for building an open-domain chatbot.

[32] Ruan, S., Jiang, L., Xu, J., Tham, B. J.-K., Qiu, Z., Zhu, Y., Murnane, E. L., Brunskill, E., and Landay, J. A. Quizbot: A dialogue-based adaptive learning system for factual knowledge. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (2019), pp. 1–13.

[33] Ruan, S., Nie, A., Steenbergen, W., He, J., Zhang, J., Guo, M., Liu, Y., Nguyen, K. D., Wang, C. Y., Ying, R., et al. Reinforcement learning tutor better supported lower performers in a math task. *arXiv preprint arXiv:2304.04933* (2023).

[34] Tack, A., and Piech, C. The AI teacher test: Measuring the pedagogical ability of blender and GPT-3 in educational dialogues. In *Proceedings of the 15th International Conference on Educational Data Mining* (Durham, United Kingdom, July 2022), A. Mitrovic and N. Bosch, Eds., International Educational Data Mining Society, pp. 522–529.

[35] Wollny, S., Schneider, J., Di Mitri, D., Weidlich, J., Rittberger, M., and Drachsler, H. Are We There Yet? - A Systematic Literature Review on Chatbots in Education. *Frontiers in Artificial Intelligence 4* (July 2021), 654924.