

The Effects of Generative AI on Computing Students' Help-Seeking Preferences

Irene Hou
Temple University
Philadelphia, PA, USA
irene.hou@temple.edu

Sophia Mettillie
Temple University
Philadelphia, PA, USA
sophia.mettillie@temple.edu

Zhuo Li
Temple University
Philadelphia, PA, USA
tuq23149@temple.edu

Owen Man
Temple University
Philadelphia, PA, USA
owen.man@temple.edu

Cynthia Zastudil
Temple University
Philadelphia, PA, USA
cynthia.zastudil@temple.edu

Stephen MacNeil
Temple University
Philadelphia, PA, United States
stephen.macneil@temple.edu

ABSTRACT

Help-seeking is a critical way that students learn new concepts, acquire new skills, and get unstuck when problem-solving in their computing courses. The recent proliferation of generative AI tools, such as ChatGPT, offers students a new source of help that is always available on-demand. However, it is unclear how this new resource compares to existing help-seeking resources along dimensions of perceived quality, latency, and trustworthiness. In this paper, we investigate the help-seeking preferences and experiences of computing students now that generative AI tools are available to them. We collected survey data (n=47) and conducted interviews (n=8) with computing students. Our results suggest that although these models are being rapidly adopted, they have not yet fully eclipsed traditional help resources. The help-seeking resources that students rely on continue to vary depending on the task and other factors. Finally, we observed preliminary evidence about how help-seeking with generative AI is a skill that needs to be developed, with disproportionate benefits for those who are better able to harness the capabilities of LLMs. We discuss potential implications for integrating generative AI into computing classrooms and the future of help-seeking in the era of generative AI.

CCS CONCEPTS

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

KEYWORDS

Generative AI, ChatGPT, computing education, help-seeking

ACM Reference Format:

Irene Hou, Sophia Mettillie, Zhuo Li, Owen Man, Cynthia Zastudil, and Stephen MacNeil. 2024. The Effects of Generative AI on Computing Students' Help-Seeking Preferences. In *Australian Computing Education Conference (ACE 2024)*, January 29-February 2, 2024, Sydney, NSW, Australia. ACM, New York, NY, USA, 10 pages. <https://doi.org/10.1145/3636243.3636248>

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
ACE 2024, January 29-February 2, 2024, Sydney, NSW, Australia
© 2024 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 979-8-4007-1619-5/24/01...\$15.00
<https://doi.org/10.1145/3636243.3636248>

1 INTRODUCTION

Not so long ago, the internet ushered in a new wave of help-seeking resources for students [62]. Students could ask questions and receive answers crowdsourced from experts around the world on Stack Overflow [12, 47, 66] and find tutorials and lectures on YouTube [6]. Over the last year, a new and widely available resource has emerged in the form of generative AI tools, such as GPT-3 and ChatGPT. A recent study revealed GPT-3 has an ability to produce code explanations that students rated higher than those crafted by their peers [34]. This raises questions into the extent and circumstances under which students will turn to these tools for assistance. Especially as many other use cases are rapidly emerging in computing education contexts [3, 9, 14, 29, 35, 41, 52, 53, 58, 59, 61]. As a result, students now have instant access to high-quality and personalized help without needing to ask questions publicly to the entire class.

It is critical to investigate the potential impact, positive and negative, these tools might have on the help-seeking preferences of students in computing education classes. We hope to gain a crucial preliminary understanding of when and why computing students choose to seek help and what resources they prefer. In particular, this paper aims to identify advantages and potential barriers to Generative AI as a help-seeking resource. This timely insight into how traditional help-seeking may have evolved with the emergence of generative AI will guide researchers in adapting their pedagogy to most effectively leverage this new help-seeking resource.

This paper explores computing students' preferences regarding generative AI. Through a mixed-methods study, we surveyed 47 students and conducted 8 interviews with additional students. Our goal was to understand the factors that influence help-seeking preferences and help resource utilization. Participants were asked to compare their experiences and preferences with traditional resources to their use of generative AI tools such as ChatGPT and GitHub Copilot. We investigated the following research questions:

- RQ 1:** How does the frequency of help-seeking resource usage vary across tasks for computing students after the advent of Generative AI?
- RQ 2:** What factor(s) may have influenced computing students' decisions in using (or not using) generative AI tools over other existing resources?
- RQ 3:** How do generative AI tools compare with other resources along perceived factors like trust, latency, and quality?

Our research uncovered varied utilization patterns and preferences concerning generative AI tools. Students highlighted support for iteration, convenience, and the mitigation of social pressures as primary incentives for their adoption of these tools for help-seeking. However, the range of preferences and experiences seems closely tied to their ability to use these tools. Students who could assess generative model responses and integrate them into their problem-solving process appeared to derive the most value.

2 RELATED WORK

2.1 Help-Seeking Behaviors and Challenges

Effective help-seeking is crucial for academic success, as it often correlates with higher academic achievement and greater academic self-confidence [57]. Help-seeking is a complex process and successful students tend to demonstrate proficient metacognitive and self-regulatory strategies [4, 28]. However, help-seeking can be challenging for many. Students can encounter socio-emotional barriers to help-seeking [16] and decision-making barriers related to which help resources to utilize and when [1, 7].

These barriers guide whether students decide to seek help, which resources to consume, and when they engage with these resources such as peers, instructors, or online resources [16, 27, 48, 54, 67].

When students seek help, they often encounter socio-emotional barriers. For example, the fear of appearing incompetent, worries about being burdensome, and the anxiety of potential rejection can significantly reduce their willingness to seek help [27, 32]. Other fears stem from social capital; students may fear harming the trust and relationships they have cultivated with the peers in their community. While knowledge exchanged through social interactions can be invaluable, students may feel that knowledge comes at a social cost [8].

Students also face decision-making barriers when choosing which help resources to utilize. Students must implicitly or explicitly evaluate the quality of the feedback source, determine receptivity of help providers, and formulate effective help requests [1, 7]. Additionally, help-seeking is not a static decision-making process. Students tend to iteratively engage with the most convenient help resources first, then move on to scarcer information resources [11, 67]. This iterative process has been called a “recursive try-again” loop [21]. Students engage with increasingly effort-intensive resources until their need for help has been met. This iterative process is further supported by a study which found that students begin their help-seeking process with informal (i.e., detached) resources like peers or internet resources, to more formal (i.e., attached) resources like office hours or classrooms [67]. This behavior suggests an underlying narrative that students are likely to prioritize help-seeking resources based on their current stage in the learning process and the resource’s perceived quality and availability, while minimizing the barriers they face.

2.2 Help-Seeking in Computing Education

Undergraduate computing students are exposed to a wide variety of academic resources, both formal and informal; this includes office hours with TAs and instructors, peers online and offline, internet forums such as StackOverflow, course materials, and online resources [67]. When learning how to program there are a variety of

challenges that students face which can be highly demoralizing. For example, they may struggle with exceedingly common syntactic issues, conceptual misunderstandings which impact their mental models of code and how it runs, and strategic knowledge about how to develop solutions to programming tasks, amongst other issues [55]. Another frustrating yet common sticking point for computing students is learning how to identify and resolve bugs [15, 46]. Navigating when and how to seek help during debugging presents a unique challenge for programming students. Moreover, the widespread availability of internet resources and prevailing industry values may play a role in how these student help-seeking behaviors differ compared to other fields. Distinctively, there is often a focus on troubleshooting and self-directed exploration, fueled by the abundance of digital resources and industry emphasis on self-directed learning [45, 69].

Doebeling and Kazerouni studied how computing students accessed help resources and found students start with online resources then move on to peers and finally to instructors only if necessary [11]. This shows how students prioritize self-learning and rely on other sources as a next step, further corroborating the idea that students may feel the need to solve problems they encounter on their own before reaching out to other help sources. When using online resources, students tend to ask specific technical questions and issues that require quick solutions rather than more conceptual questions. Additionally, students view the use of online help resources more like a tool to utilize continuously throughout the coding process. More conceptual questions are reserved for when students are requesting help from sources such as their peers and instructors [11]. The prioritization of online tools by computing students when seeking help, coupled with the surging popularity of generative AI, could impact which resources students favor and when they choose to use them.

2.3 Generative AI in Computing Education

Recently, there has been significant excitement about the potential for using generative AI in computing classrooms [33, 51, 70] with researchers already demonstrating many remarkable capabilities of generative AI including generating code explanations [34, 39, 42], enhancing error messages [35], identifying bugs [38], solving multiple-choice programming problems [59, 61], solving Parsons problems [22], and creating instructional materials and programming assignments [14, 58, 65]. A study on the impacts of chatbots on learning performance and motivation found students were motivated to receive help from a chatbot because it supported self-paced learning [68]. Recently, a study showed that LLM-generated explanations of code even surpassed that of peers in a computing classroom [34]. Another study investigating how generative AI tools respond to student help requests have noted that GPT-3.5 in particular is helpful in identifying problematic areas of student code [20]. LLMs have also been shown to be effective at providing on-demand, scalable programming assistance in computing education classes [36]. Previous studies have also identified that novice programming students write better code when supported by LLMs without decreasing their ability to manually modify code [29]. This is promising for help-seekers as they can solve problems that might

otherwise be outside of their ability without losing their agency to adapt and revise the code themselves.

Given these capabilities ranging from personalized support to timely feedback, generative AI tools could emerge as a robust source of help for computing students. However, whether or not students prefer these AI tools in comparison to other resources is a question that has been left largely unexplored. Research is needed to understand how and why students rely on these tools in the context of other existing help resources.

3 METHODOLOGY

To evaluate our research questions, we conducted a survey and interview study. All research was conducted after receiving approval from our university's Institutional Review Board.

3.1 Participant Recruitment

To recruit a diverse range of participants, we advertised our study on multiple university campuses. We advertised in student and computing-related Discord servers, Slack channels, and subreddits. We also recruited student organization leaders and faculty to make announcements in their clubs and courses. Participation in the survey was uncompensated and interview participants received a \$10 gift card. All interview participants and survey responses were collected from July-October 2023.

3.2 Survey Study

3.2.1 Participants. We collected 47 survey responses (32 male, 13 female, 2 preferred not to say) from computing students (i.e., students who have taken at least one computing course) across multiple North American universities. Participants included 2 incoming first-years, 19 first-years, 9 second-years, 12 third-years, and 4 fourth-years. 42 out of 47 participants had used ChatGPT before but not necessarily other AI tools (e.g. Midjourney, GitHub Copilot). Students rated the difficulty of their CS courses from Easy (1) to Hard (5), reporting a median difficulty of 3 ($\mu = 3.0$, $\sigma = 0.78$).

3.2.2 Design of survey questions. To understand students' help-seeking preferences, we asked students to rank seven popular help-seeking resources. These included ChatGPT and Github Copilot which are two popular Generative AI tools used by programmers. These also included five "traditional" help-seeking resources based on a previous survey of computing students' help preferences [67].

Students ranked help received from online resources (e.g., Youtube and StackOverflow), course discussion forums, TAs, instructors, friends, ChatGPT, and GitHub Copilot in order of preference for common CS tasks including understanding course concepts, writing or generating code, debugging, and developing test cases. These scenarios were chosen based on a study that determined what kinds of help computing students seek in TA office hours [56] and the stages of programming problem-solving by Loksa et al [37]. Additionally, participants ranked each resource along the same scale for the following dimensions of assistance: perceived quality, trustworthiness, and timeliness.

Participants were asked to rank their preferences for the 7 resources from "Most Preferred" to "Least Preferred." Participants also rated how often they used help-seeking resources on a scale from "Never" to "Hourly."

Participants were also asked to rate the help-seeking resources on a five-point Likert-scale for perceived convenience, quality, anonymity, and potential for supporting follow-up questions. Lastly, participants were asked to evaluate how comfortable they were seeking help from each resource on a five-point Likert-scale. Participants rated their comfort levels with highly anonymous resources (e.g., ChatGPT or online resources) compared to direct resources (e.g., in-class assistance or instructors).

3.2.3 Survey analysis. Survey responses were analyzed by computing the mean, median, and standard deviation for all Likert-scale questions and visualizing rankings of the help resources. We present these summary statistics narratively alongside quotes to contextualize the findings. We also generated data visualizations to summarize the ranked preferences.

3.3 Interview Study

3.3.1 Participants. To gain a more nuanced perspective on student help-seeking preferences, we also carried out 8 semi-structured interviews (1 female, 7 male) which lasted between 30-45 minutes. All participants had used ChatGPT at least once, but not necessarily other generative AI tools (e.g. Midjourney, GitHub Copilot). 6 out of 8 participants were first or second-year computing students, while the others were third-years or above. Participants were able to schedule an interview without having completed the survey.

3.3.2 Design of interview questions. The semi-structured interview format allowed us to deviate from our prepared interview questions, yielding richer and more varied perspectives [43]. Like the survey, students were asked to rank their help-seeking preferences for each task (e.g. understanding course concepts, writing or generating code, debugging, and developing test cases). We also asked participants to share their help-seeking process, what they valued most and least from each resource, and to draw comparisons with their experiences using ChatGPT. For example, if students claimed they preferred asking ChatGPT due to anonymity, we posed follow-up questions such as: "If the questions you ask ChatGPT could be viewed by your peers, would you still ask the same questions? Why or why not?" This allowed us to probe the underlying preferences and values more deeply [17, 18].

3.3.3 Interview analysis. Interviews were conducted on Zoom, which automatically transcribes the conversation. One researcher parsed through the transcripts to anonymize the participant and make corrections to the transcripts. The transcripts were then analyzed using a thematic analysis technique [5]. This technique started with open coding [63] with the goal of enabling the emergence of patterns and categories that might not have been anticipated. These codes were grouped thematically. Next, the researchers looked for connections with established theoretical frameworks such as social comparison theory [64], "recursive try-again loop" [21], and anonymity effects [16, 50]. This approach allowed our qualitative analysis to balance inductive findings grounded in participants experiences and perspectives with existing knowledge about feedback exchange theory. Transcripts were first reviewed individually by three researchers to generate initial codes. Researchers then met twice to collaboratively discuss emerging quotes and themes,

Table 1: Frequency of Resource Usage by Students

Resource	Hourly	Daily	Weekly	Monthly	Never
Online search	9	24	11	2	1
Friends	2	13	16	10	6
Class forum	2	10	14	8	13
ChatGPT	4	7	9	11	16
Instructor	1	6	14	17	9
TA	1	3	15	16	12
GitHub Copilot	1	1	3	2	40

refining their collective understanding of the data to achieve convergence and minimize interpretation biases. Researchers came to a consensus through discussion. The results from this thematic analysis are presented in Section 4.3.

Additionally, we conducted a secondary analysis to support the survey responses. The results for this analysis are presented in Section 4.2. For this analysis we adopted a deductive approach that aligned with the questions asked in the survey. We looked explicitly for instances where participants talked about their usage of help seeking resources including when understanding coding concepts, debugging code, writing code, and creating test cases.

4 RESULTS

4.1 Frequency of Usage

Table 1 displays how frequently students seek help from the seven help-seeking resources in the survey. Searching online was the most popular choice for help seekers with over 70.2% of students using it on either an hourly or daily basis. Students also relied to a lesser extent on friends (31.9%), course discussion forums (25.5%), and ChatGPT (23.4%) to meet these hourly or daily needs. The Internet, friends, and discussion forums were used consistently on a weekly basis by over half of the students surveyed. At least once a week, 44.7% of students consulted instructors and 42.6% of students consulted ChatGPT.

Students depended least on TAs, with 59.6% of students seeking help monthly or even less frequently. One possible reason for the high frequency use of instructor help could be the inclusion of diversely-sized universities; some universities surveyed had smaller classrooms than others, meaning fewer TAs and more dependence on the instructor. Possible reasons for the reluctance to seek help from TAs could include temporal and spatial limitations, as suggested by previous research [67]. This aligns with what interview participants mentioned regarding resource availability. For example, P2 said, “For hard classes there’s too long of a line; sometimes their [TAs and instructors] schedules don’t fit.” Multiple students seeking assistance from a single TA or instructor could make this resource seem less accessible and therefore less appealing.

With regard to the frequency of use, ChatGPT appears to be a slightly polarizing choice; 26.1% of students who have experience with ChatGPT use it daily while 34.0% of students never use it at all. ChatGPT’s bimodal distribution of usage could be due to a mix of lack of awareness, lack of trust (as further explored later in the interview), or concerns regarding academic dishonesty [33, 51, 70].

4.2 Context of Usage

While students may rely more heavily on some help-seeking resources than others, we investigated if specific resources were perceived as more useful for tasks like writing or debugging code. Summarized in Figure 1, students’ use of various help-seeking resources varied according to task. Across tasks, students ranked the Internet as their most preferred method for seeking help. This is consistent with our observation that students rely heavily on web searches to meet their information needs. The quotes presented in this section came from the interviews and helps to contextualize how generative AI tools were used.

4.2.1 Learning new concepts. After internet search, instructors were the next most preferred resources for learning new concepts. However, despite ChatGPT being a relatively nascent resource, it was comparably popular to asking friends or in the discussion forums. The primary reasons for this preference based on our interview data was the iterative nature of responses from ChatGPT and the reduced social pressures of not needing to request help directly from a peer. The following quote demonstrates both aspects:

*“It is easier for me to ask ChatGPT because I don’t have to worry... I feel like **it is kind of rude to continuously ask for more and more details, especially when you’re approaching someone for help.**” (P1)*

This quote illustrates how students are at times hesitant to inconvenience their peers. With ChatGPT students can iterate and develop their understanding without concern for social etiquette. Later, the same student shared their biggest pain point when searching online to learn new concepts:

*“The issue is with the scope of what I’m looking for...**when I try to online search, I can often get sidetracked by all these complicated, really advanced terms or concepts. So I would try to look at the instructor’s items first.**” (P1)*

While internet resources were still highly ranked, P1’s comment highlights how for computing students, the web can be an overwhelming source of information, whereas resources like instructors and TAs can provide more scaffolding and guidance. ChatGPT has the potential to bridge this gap by providing more personalized help without the social pressures associated with other help resources.

P7, who ranked ChatGPT lower than instructors, said that this scaffolding as a ‘process’ was more important than the answer when seeking help:

*“I feel that [getting an answer instantly] kind of defeats the purpose of me asking a question. If I’m asking a question, I’m trying to understand what’s going on. If you just give me the answer, then you’re not describing the process at all. **I much prefer somebody who gives me the process. That’s what I find most valuable**” (P7).*

These viewpoints emphasize the value of scaffolding help-seeking in learning. How easy it is to navigate and make sense of information can be a reason why students choose certain help-seeking resources over others.

4.2.2 Writing code. When students talked about writing code, a theme emerged that students valued ChatGPT’s capacity to support

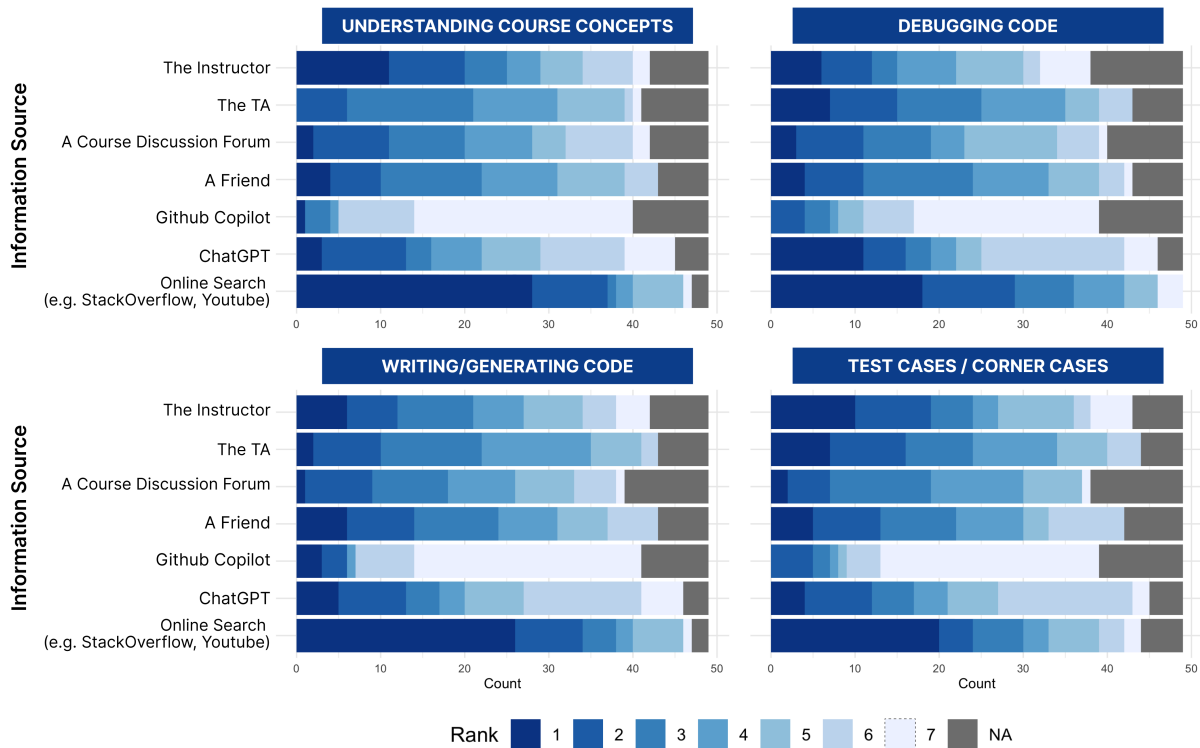


Figure 1: Students survey rankings of four common CS tasks

new ideas (P1, P3, P6, P7). One participant pointed out that it felt more challenging to seek abstract, high-level help from a TA: “I feel like the TA is there for technical help, not for more creative help.” (P3). Students viewed traditional help resources like the TA as more helpful with common pitfalls and coursework: “TAs and instructors have experience from multiple students asking questions about the same problems” (P4). For high-level tasks such as writing code from scratch, P7 noted ChatGPT’s creative potential:

“If I were to ask ChatGPT a math question, it might get it wrong. But in more creative capacities, that’s where I feel like it excels. When I asked it pre-thinking questions...it’ll actually give me a very reasonable list.”

While traditional academic scaffolds, such as TAs, remain invaluable for their historical and course-related expertise, participants tended to describe ChatGPT as being more useful when trying to think outside the box.

4.2.3 Debugging. Students again strongly preferred internet sources for debugging, followed by ChatGPT and TAs, then course discussion forums. Across the tasks, this appeared to be the use case where ChatGPT was mildly preferred compared to other use cases. Similar to when learning new concepts, students that preferred ChatGPT cited its ability to provide multiple, iterative perspectives and direction in solving specific problems. Another factor was the ability to articulate help requests when debugging. This difficulty

of finding relevant solutions when searching online based on an error message was captured by P3:

“For error messages, when I copy an error message from the terminal and paste it to the Internet, it’s not always guaranteed to have a result or a solution.”

In the same breath, the same student highlights an emergent paradigm in debugging scenarios:

“But if you put it [the error message] into ChatGPT, it will come up with something. Even though it may not be right, it will still be helpful, like a correct direction” (P3).

It is unclear if this participant, like many students [44], struggled to articulate their error as a search query. It is also possible that models are better able to infer information and omit irrelevant information (e.g.: the working directory or line numbers) from help requests.

Another participant specifically cited ChatGPT’s ability to provide varied responses as helpful:

“I usually try to give ChatGPT the same prompt a couple times. Based off what it tells me, I’ll see what comes up most often. Then, based off of my actual knowledge, I’ll see which one’s the right answer” (P1).

One notable insight is that P1 appears to use frequency as a heuristic for correctness or reliability. If a solution emerges repeatedly from ChatGPT, P1 considered it as more likely to be helpful. However, prompting ChatGPT multiple times with the same prompt

may not be the most successful approach. It is possible that—similar to help seeking in other contexts—it is better to reformulate help requests than to repeatedly ask the exact same question. Similar to P3, it is unclear if P1 was able to articulate their queries effectively. Additionally, P1’s process of cross-checking ChatGPT’s answers with their own knowledge underscores the importance of domain knowledge when evaluating generative AI outputs.

When asked to draw comparisons between internet sources and ChatGPT, one participant observed: “On Stack Overflow, if somebody asks a question, **the answer that best suits the question is voted to the top of the list**. And with ChatGPT and Copilot, you don’t really get any of that” (P7). This again suggests that students require support in evaluating and prioritizing generative AI responses. When seeking help on the web, students have to carefully formulate their search queries to find relevant results. The ability for ChatGPT to infer some meaning from even poorly formed help requests may become an asset when combined with other help seeking resources.

4.2.4 Developing Test Cases. While not particularly low-ranked in the survey data, participants in the interviews did not talk much about using ChatGPT to find corner cases or to develop test cases. They relied most on course assignment guidelines to test their code and ensure edge cases were covered. P1 said, “I would look at instructor materials for us to see what kind of test cases... they’re specifically asking for.” This shift in preferences may be due to the computing students sampled for the interview. Students at this level are often overly concerned about the requirements of the assignment, affecting how they face challenging course materials [30].

4.3 Factors Influencing Usage

Based on the ranking data, summarized in Figure 2, participants appeared to favor internet resources and ChatGPT over other resources in terms of timeliness; however, these two help resources were perceived as less trustworthy and lower quality than the help received from instructors or TAs. Themes that emerged from interviewed students included trade-offs between rapid feedback and quality, social dynamics, and strategies such as iteration and reformulation when help-seeking.

4.3.1 Trust and trade-offs between convenience and quality. The perceived trade-off between receiving help that is both efficient and accurate emerged as a critical theme from survey and interview data. 68.1% of the students surveyed Agreed or Strongly Agreed that generative AI tools like ChatGPT were more convenient than alternatives. While they offer immediate and convenient feedback, students noted that this can come at the expense of correctness.

The lack of feedback latency was considered a major benefit by nearly every interviewee (P1, P3, P4, P5, P6, P7, P8). On the other hand, all but one interview participant described ChatGPT as an unreliable source of help, and a minority of students believed the trade-off between speed and quality was not worth it. First-year students and self-reported novices tended to be harsher in their judgement of ChatGPT’s reliability (P3, P4, P6, P8). As P8 candidly shared, “**I personally don’t really care if [ChatGPT] is fast, I’ll wait [for a peer or TA]...the most important thing is getting it right.**” It is worth noting that P8 also described a negative experience with ChatGPT hallucinations: “My partner used ChatGPT

to generate a Python switch statement, which doesn’t exist. I wouldn’t trust [ChatGPT] to do it again.” P2 echoed similar sentiments: “**I don’t have a lot of trust in ChatGPT in terms of the harder coding classes and the harder theoretical stuff.**”

Conversely, P7, who had considerable programming experience, found the trade-off between efficiency and quality acceptable:

“I’ll take ChatGPT’s advice with a grain of salt anyway. I will generally use whatever solutions that it gives me...and toss out what doesn’t work.”

Similarly, P5, a further studies student, stated: “I would trust ChatGPT more [than my peers] because you can search it. It has so much data inside. **[ChatGPT] is like many humans with so much information that you can search through and find conclusions from.**” Another student pointed out that if ChatGPT gave inaccurate information, they would treat it as a “third party or an alternate input or perspective” (P4). P4 and P5’s insight emphasizes that those who receive the most value from ChatGPT tend to view the model as a search engine or comprehensive resource, from which they must derive pertinent information on their own. This mental model differs from students like P8, who prioritize an expert-level of accuracy.

Although multiple students cited instances when peers gave incorrect or poor quality help, 7 out of 8 students still reportedly preferred seeking help from peers over ChatGPT. This trend aligned with survey data where 51.1% of students disagreed that ChatGPT provided better quality feedback than alternatives. P2 described the conflicting process of seeking help from peers: “Sometimes they’re not even sure about their answers, and when I get a different answer, we’re both questioning ourselves.” P4 went more into detail on this process: “I’ll take the response from my peers if it makes sense to me. I’ll take it with a good amount of trust. If I apply it and it doesn’t work, that’s OK. The whole purpose is just to get another approach...and if that doesn’t work then it’s not a big deal.” But when he was asked to compare this experience with ChatGPT, he said: “I definitely don’t trust ChatGPT with the accuracy of its response, especially CS stuff.”

Students largely fell into one of two groups: expecting generative AI tools to be comparable to experts or believing that it is their responsibility to steer the models toward giving useful responses. Unlike when seeking assistance from peers, some participants were less lenient with ChatGPT and subsequently more frustrated when they did not get help after issuing a single prompt.

4.3.2 Social Aspects. Social dynamics of help-seeking among computing students involve elements of personal comfort, perceived judgment, and the desire to maintain reciprocal peer relationships. Table 2 indicates that most students reported asking ChatGPT for help as *Very Comfortable*, while other scenarios saw a wider distribution across comfort levels. 68.1% of respondents reported their comfort levels of asking ChatGPT as *Comfortable* or *Very Comfortable*, compared to traditional forms of help, such as TAs/Instructors (66.0%) and peers privately (72.3%) or anonymously (72.3%). This comfort level dropped to 46.8% when students were asked how they felt about publicly seeking help. Although students were most comfortable seeking help from their peers, they still found asking ChatGPT more comfortable than asking a TA/Instructor or asking for help in a public setting. Overall, ChatGPT appears to provide a

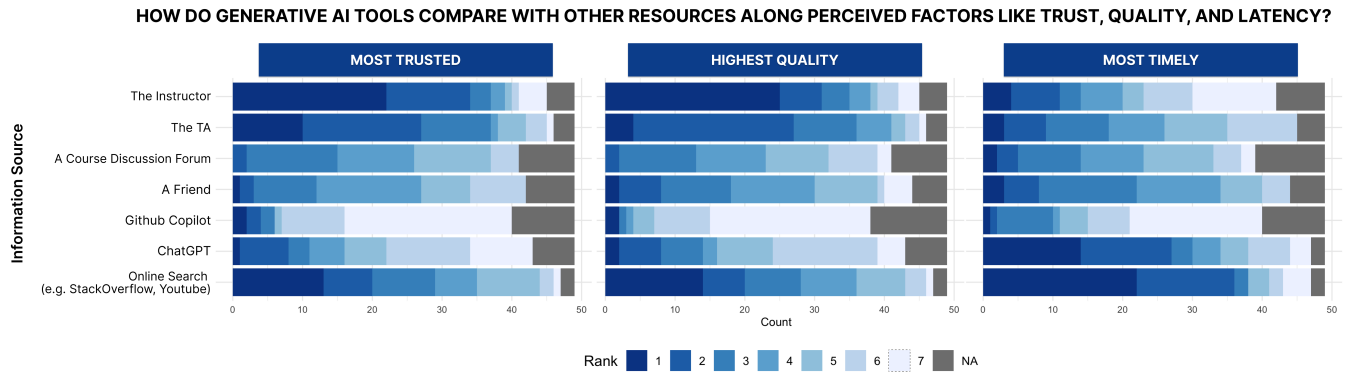


Figure 2: Students' survey rankings of perceived qualities of help seeking resources

Scenario	Very Comfortable	Comfortable	Neutral	Uncomfortable	Very Uncomfortable
Asking ChatGPT for help	24 (51.1%)	8 (17.0%)	8 (17.0%)	2 (4.3%)	5 (10.6%)
Asking a peer privately	23 (48.9%)	11 (23.4%)	7 (14.9%)	4 (8.5%)	2 (4.3%)
Asking a TA/Instructor privately	19 (40.4%)	12 (25.5%)	8 (17.0%)	8 (17.0%)	0 (0%)
Asking other students anonymously	18 (38.3%)	16 (34.0%)	4 (8.5%)	9 (19.1%)	0 (0%)
Asking in front of other students publicly	12 (25.5%)	10 (21.3%)	13 (27.7%)	9 (19.1%)	3 (6.4%)

Table 2: Comfort Level of Seeking Help in Different Scenarios

comfortable learning environment for students, free of potential social pressures and apprehensions.

In interviews, one primary concern expressed was the reluctance to 'burden' peers with frequent help requests.

"I like going to ChatGPT more because I didn't feel like I had to burden my peers at all," (P7)

The desire to avoid straining relationships by over-relying on peers was echoed by most students, such as P1: "If I ask for help sometimes and then I keep asking 5 more times, I do feel really bad, and I might stop asking." P8 admitted that seeking help is challenging and that "When it comes to people that I know personally, or that I already have messed with, I'll ask...but I still feel judgment." Reciprocity considerations were also brought up by participants, most succinctly by P1 who said he does "try to maintain at least a 50/50-ish balance in terms of asking for help...I don't want our relationship to be this kind of strained—like they're giving a lot more than I'm giving." These social dynamics and reputational barriers highlight the continued need for safe student help-seeking spaces. The option to tap into generative AI tools alleviates some fears surrounding being perceived as less knowledgeable or burdensome. The 'always-on' nature of AI tools can be empowering, allowing students to progress past common programming sticking points without expending social energy or social capital. P5 aptly puts it: **"As an introvert, personally, I think ChatGPT is a very good way to get help because talking to people kind of takes away my energy."**

On the other hand, the value of help-seeking in community settings cannot be overstated. While AI tools like ChatGPT could reduce social pressures surrounding help-seeking, students stressed

the significant value of human interaction, mutual support, and community. P8 shared an experience that provided testament to the power of collective learning and peer help-seeking, emphasizing the role of mutual struggle in fostering camaraderie:

"I think the thing that got me through CS was the entire community actively also struggling with their work, learning something new, and having a really, really hard time. I wasn't the only person that was having a hard time...We're all in Discord, we're all in communication...I had no problem asking like, 'Hey, what did you do?'" (P8)

4.3.3 Iteration. The iterative capabilities of ChatGPT and its ability to handle follow-up questions emerged as a positive for 44.7% of students, who Agreed or Strongly Agreed that ChatGPT was better at supporting follow-up questions compared to its alternatives. This potential for iteration was especially valuable for students attempting to grasp unfamiliar concepts or clarify misunderstandings. For example, P4 shared the following process:

"If you want to ask ChatGPT a concept...I feel like you could ask it something, read its response, and ask it to clarify on a response...like follow-up questions...if I don't understand what it says, I'll prompt it again or I'll ask it to specify."

Some students valued the autonomy provided by ChatGPT's rapid feedback, enabling them to continuously seek clarifications and review uncertainties at their own pace. The language that students used to describe how they query ChatGPT centers around repetition and continuity, implying that the process of seeking help was not a one-and-done experience but cyclical in nature.

Additionally, students found value in ChatGPT’s ability to recall past interactions within a chat session. For example, P8 said:

“ChatGPT has the feature of looking at past conversations in the thread. I find that very, very helpful. I can make a dedicated thread on one topic.”

This continuity in conversation mirrors the cognitive processes that might take place when students are still conceptualizing new information, allowing them to return to topics and re-familiarize themselves. This constitutes a sensemaking process, reminiscent of organizing notes.

P1 iterated to find answers using the regenerate button:

“I can basically ask ChatGPT as many times as I want. So that’s why I’d often regenerate responses over and over...the answers I found through regenerating responses can be really, really inconsistent.”

Instead of reformulating their help request, which is a key strategy in help-seeking [28], this student continued asking the same question, expecting an improved answer each time.

5 DISCUSSION

Our results suggest that while generative AI tools like ChatGPT have become a widely used resource in a very short amount of time, significant barriers still exist that reduce students’ use of it for most tasks. ChatGPT usage varied slightly by task, with students using it more for debugging and writing code. Aligning with prior work, students largely made decisions about which help-seeking resources to use based on the quality and availability of the help resource [54, 67]. However, students highlighted features unique to ChatGPT that traditional resources lacked, such as its creative generative potential and its capacity to provide iterative support in the form of follow-up questions. Unlike internet resources like Stack Overflow or YouTube, students also appreciated how ChatGPT actively steered them toward solutions for rare bugs or error messages, even if it did not always provide the exact answer.

Another theme that emerged was related to experience level and trust. Students who were less familiar with these models or who had early negative experiences were much less likely to want to use them. This is partially explained by the concept of calibrated trust [2, 71] where early negative experiences calibrated students to distrust the models. This is further exacerbated by the fact that models can perform well at times, while also hallucinating incorrect information and struggling on easy multiple choice questions [60, 61]. Less experienced students described being especially apprehensive about receiving wrong answers and being unable to discern between correct and incorrect responses. This skepticism is a promising finding given the widespread fears about students blindly relying on these tools [3, 70].

Conversely, experienced students were more lenient with the models. Students mentioned the necessity of applying their own domain knowledge to evaluate the correctness of the model’s responses; hence, more knowledgeable students were better equipped to filter through incorrect responses and find the bits that were valuable or could “guide” their next steps. Across experienced and inexperienced students, distrust did not necessarily mean students failed to receive value from them as we saw most students using the models to varying extents.

Formulating effective help requests appeared to be a common challenge for students. Even without generative AI tools, help-seeking remains a difficult task. When asking peers or instructors for help, students grapple with the ‘epistemic paradox of having to know what they do not know’ in order to receive assistance [31]. This problem appears to persist even with tools like ChatGPT; if students fail to formulate their help request to the model clearly, they do not get the most effective help.

Given this gap in abilities to use these models, more research is needed to understand how well students are able to use generative AI tools. Instructors need to continue to create pedagogical materials or scaffolding [10, 24–26, 40] that can guide students to maximize the utility of these tools. Crucially, if students are unable to formulate effective help requests, the model will not save them. Models can not infer missing information or pose follow-up questions in the same way instructors might be able to form a theory of mind about what the learner knows and does not know in order to tailor explanations accordingly. Students described instances where they prompted the model and only asked it to regenerate responses when the initial response was not appropriate. On the other hand, confident users described using ChatGPT as a conversation where they needed to reformulate their help requests or provide additional context via follow-up questions.

5.1 Limitations

This study offers a important snapshot of how students are initially incorporating generative AI tools into their help-seeking process, providing critical insights into this dynamic landscape. As student usage of these tools is expected to escalate [3, 51], future updates to this study will be essential. Despite the inherent limitations of being one of the first explorations in this rapidly evolving field, our findings shed light on the early stages of student adoption. Notably, our interview study, while relatively smaller (n=8) compared to recent works [33, 70] with 18 and 20 participants respectively, is complemented by a medium-sized survey involving 45+ students. However, concentrating on North American students underscores the need for future research with larger and more diverse samples to extend the relevance of our observations to a broader population as the field continues to evolve. Additionally, our sample included students from varied levels of undergraduate study with a skew towards first-year students. Future work could examine the experiences of a more homogeneous population, such as introductory students or even compare the differences between perceptions of introductory computing students versus more senior students and what other help-seeking resources they may rely on.

6 CONCLUSION

In this study, we examined students’ help-seeking behaviors through surveys and interviews. Our findings reveal diverse patterns in the utilization and preferences for generative AI tools in varying tasks. Students described the potential for rapid iteration, creative ideation, and avoiding social pressures as key motivations for using these tools. Notably, the range of preferences and experiences among participants seems closely tied to their individual mental models of these models’ function. Students either utilized ChatGPT’s responses as an initial step in their problem-solving process

or became discouraged after receiving an incorrect answer. Finally, it appears that students received disproportionate benefit when seeking help from generative AI tools. Students who were willing and able to iterate and reformulate their feedback requests described having more success. While these models undoubtedly offer a valuable new resource for help-seeking, it is crucial to acknowledge that the quality of assistance received is intricately tied to the clarity and precision of the initial help request. In this regard, seeking help from generative AI mirrors the importance and difficulty of articulating a help request to a friend [19, 27, 48] or formulating and reformulating a search query [13, 23, 49].

REFERENCES

- [1] Vincent Alevén, Elmar Stahl, Silke Schworm, Frank Fischer, and Raven Wallace. 2003. Help Seeking and Help Design in Interactive Learning Environments. *Review of Educational Research* 73, 3 (2003), 277–320. <https://doi.org/10.3102/00346543073003277> arXiv:[https://doi.org/10.3102/00346543073003277](https://arxiv.org/abs/https://doi.org/10.3102/00346543073003277)
- [2] Maryam Ashoori and Justin D Weisz. 2019. In AI we trust? Factors that influence trustworthiness of AI-infused decision-making processes. *arXiv preprint arXiv:1912.02675* (2019).
- [3] Brett A. Becker, Paul Denny, James Finnie-Ansley, Andrew Luxton-Reilly, James Prather, and Eddie Antonio Santos. 2023. Programming Is Hard - Or at Least It Used to Be: Educational Opportunities and Challenges of AI Code Generation. In *Proceedings of the 54th ACM Technical Symposium on Computer Science Education V. 1* (Toronto ON, Canada) (SIGCSE 2023). Association for Computing Machinery, New York, NY, USA, 500–506. <https://doi.org/10.1145/3545945.3569759>
- [4] Susan Bergin, Ronan Reilly, and Desmond Traynor. 2005. Examining the Role of Self-Regulated Learning on Introductory Programming Performance. In *Proceedings of the First International Workshop on Computing Education Research* (Seattle, WA, USA) (ICER '05). 81–86.
- [5] Virginia Braun and Victoria Clarke. 2006. Using thematic analysis in psychology. *Qualitative Research in Psychology* 3, 2 (2006), 77–101.
- [6] Nicole Buzzetto-Hollywood. 2014. An Examination of Undergraduate Student's Perceptions and Predilections of the Use of YouTube in the Teaching and Learning Process. *Interdisciplinary Journal of E-Learning and Learning Objects (IJELLO)* 10 (01 2014), 17–32. <https://doi.org/10.28945/1965>
- [7] Kun-Hung Cheng and Chin-Chung Tsai. 2011. An investigation of Taiwan University students' perceptions of online academic help seeking, and their web-based learning self-efficacy. *Internet and Higher Education - INTERNET HIGH EDUC* 14 (07 2011), 150–157. <https://doi.org/10.1016/j.iheduc.2011.04.002>
- [8] Chao-Min Chiu, Meng-Hsiang Hsu, and Eric T.G. Wang. 2006. Understanding knowledge sharing in virtual communities: An integration of social capital and social cognitive theories. *Decision Support Systems* 42, 3 (2006), 1872–1888.
- [9] Paul Denny, Viraj Kumar, and Nasser Giacaman. 2023. Conversing with Copilot: Exploring Prompt Engineering for Solving CS1 Problems Using Natural Language. In *Proceedings of the 54th ACM Technical Symposium on Computer Science Education V. 1* (Toronto ON, Canada) (SIGCSE 2023). Association for Computing Machinery, New York, NY, USA, 1136–1142. <https://doi.org/10.1145/3545945.3569823>
- [10] Paul Denny, Juho Leinonen, James Prather, Andrew Luxton-Reilly, Thezyrie Amarouche, Brett A Becker, and Brent N Reeves. 2023. Promptly: Using prompt problems to teach learners how to effectively utilize ai code generators. *arXiv preprint arXiv:2307.16364* (2023).
- [11] Augie Doebling and Ayaan M. Kazerouni. 2021. Patterns of Academic Help-Seeking in Undergraduate Computing Students. In *Proceedings of the 21st Koli Calling International Conference on Computing Education Research* (Joensuu, Finland) (Koli Calling '21). Association for Computing Machinery, New York, NY, USA, Article 13, 10 pages. <https://doi.org/10.1145/3488042.3488052>
- [12] Pierpaolo Dondio and Suha Shaheen. 2020. Is StackOverflow an Effective Complement to Gaining Practical Knowledge Compared to Traditional Computer Science Learning?. In *Proceedings of the 11th International Conference on Education Technology and Computers* (Amsterdam, Netherlands) (ICETC '19). ACM.
- [13] Carsten Eickhoff, Sebastian Dungs, and Vu Tran. 2015. An eye-tracking study of query reformulation. In *Proceedings of the 38th international ACM SIGIR conference on research and development in information retrieval*. 13–22.
- [14] James Finnie-Ansley, Paul Denny, Brett A Becker, Andrew Luxton-Reilly, and James Prather. 2022. The Robots Are Coming: Exploring the Implications of OpenAI Codex on Introductory Programming. In *Australasian Computing Education Conf*. 10–19.
- [15] Sue Fitzgerald, Gary Lewandowski, Renee McCauley, Laurie Murphy, Beth Simon, Lynda Thomas, and Carol Zander. 2008. Debugging: finding, fixing and flailing, a multi-institutional study of novice debuggers. *Computer Science Education* 18, 2 (2008), 93–116.
- [16] Eureka Foong, Steven P Dow, Brian P Bailey, and Elizabeth M Gerber. 2017. Online feedback exchange: A framework for understanding the socio-psychological factors. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. 4454–4467.
- [17] Batya Friedman. 1996. Value-sensitive design. *interactions* 3, 6 (1996), 16–23.
- [18] Batya Friedman, Peter H Kahn, Alan Borning, and Alina Hultgren. 2013. Value sensitive design and information systems. *Early engagement and new technologies: Opening up the laboratory* (2013), 55–95.
- [19] Zhikai Gao, Bradley Erickson, Yiqiao Xu, Collin Lynch, Sarah Heckman, and Tiffany Barnes. 2022. You Asked, Now What? Modeling Students' Help-Seeking and Coding Actions from Request to Resolution. *Journal of Educational Data Mining* 14, 3 (2022), 109–131.
- [20] Arto Hellas, Juho Leinonen, Sami Sarsa, Charles Koutchme, Lilja Kujanpää, and Juha Sorva. 2023. Exploring the Responses of Large Language Models to Beginner Programmers' Help Requests. In *Proceedings of the 2023 ACM Conference on International Computing Education Research V.1*. ACM.
- [21] Christopher Herring and Joachim Walther. 2016. Academic help-seeking as a stand-alone, metacognitive action: An empirical study of experiences and behaviors in undergraduate engineering students. In *2016 ASEE Annual Conference & Exposition*.
- [22] Irene Hou, Owen Man, Sophie Mettill, Sebastian Gutierrez, Kenneth Angelikas, and Stephen MacNeil. 2023. More Robots are Coming: Large Multimodal Models (ChatGPT) can Solve Visually Diverse Images of Parsons Problems. *arXiv preprint arXiv:2311.04926* (2023).
- [23] Jeff Huang and Efthimis N Efthimiadis. 2009. Analyzing and evaluating query reformulation strategies in web search logs. In *Proceedings of the 18th ACM conference on Information and knowledge management*. 77–86.
- [24] Ziheng Huang, Sebastian Gutierrez, Hemanth Kamana, and Stephen MacNeil. 2023. Memory sandbox: Transparent and interactive memory management for conversational agents. In *Adjunct Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*. 1–3.
- [25] Ziheng Huang, Kexin Quan, Joel Chan, and Stephen MacNeil. 2023. CausalMapper: Challenging Designers to Think in Systems with Causal Maps and Large Language Model. In *Proceedings of the 15th Conference on Creativity and Cognition* (Virtual Event, USA) (C&C '23). Association for Computing Machinery, New York, NY, USA, 325–329. <https://doi.org/10.1145/3591196.3596818>
- [26] Ellen Jiang, Kristen Olson, Edwin Toh, Alejandra Molina, Aaron Donsbach, Michael Terry, and Carrie J Cai. 2022. PromptMaker: Prompt-based Prototyping with Large Language Models. In *CHI Conference on Human Factors in Computing Systems Extended Abstracts*. 1–8.
- [27] Stuart A Karabenick. 2003. Seeking help in large college classes: A person-centered approach. *Contemporary educational psychology* 28, 1 (2003), 37–58.
- [28] Stuart A Karabenick and Jean-Louis Berger. 2013. Help seeking as a self-regulated learning strategy. (2013).
- [29] Majeed Kazemitabaar, Justin Chow, Carl Ka To Ma, Barbara J. Ericson, David Weintrop, and Tovi Grossman. 2023. Studying the effect of AI Code Generators on Supporting Novice Learners in Introductory Programming. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems* (CHI '23). ACM. <https://doi.org/10.1145/3544548.3580919>
- [30] Päivi Kinnunen and Beth Simon. 2012. My program is ok – am I? Computing freshmen's experiences of doing programming assignments. *Computer Science Education* 22, 1 (2012), 1–28. <https://doi.org/10.1080/08993408.2012.655091> arXiv:[https://doi.org/10.1080/08993408.2012.655091](https://arxiv.org/abs/https://doi.org/10.1080/08993408.2012.655091)
- [31] Tom Koole. 2012. The epistemics of student problems: Explaining mathematics in a multi-lingual class. *Journal of Pragmatics* 44, 13 (2012), 1902–1916. <https://doi.org/10.1016/j.pragma.2012.08.006>
- [32] David E Kumrow. 2007. Evidence-based strategies of graduate students to achieve success in a hybrid Web-based course. *The Journal of Nursing Education* 46, 3 (2007), 140–145.
- [33] Sam Lau and Philip J. Guo. 2023. From 'Ban It Till We Understand It' to 'Resistance is Futile': How University Programming Instructors Plan to Adapt as More Students Use AI Code Generation and Explanation Tools such as ChatGPT and GitHub Copilot. In *Proceedings of the 2023 ACM Conference on International Computing Education Research V.1* (ICER '23 V1). ACM. <https://doi.org/10.1145/3568813.3600138>
- [34] Juho Leinonen, Paul Denny, Stephen MacNeil, Sami Sarsa, Seth Bernstein, Joanne Kim, Andrew Tran, and Arto Hellas. 2023. Comparing Code Explanations Created by Students and Large Language Models. *arXiv preprint arXiv:2304.03938* (2023).
- [35] Juho Leinonen, Arto Hellas, Sami Sarsa, Brent Reeves, Paul Denny, James Prather, and Brett A Becker. 2023. Using large language models to enhance programming error messages. In *Proceedings of the 54th ACM Technical Symposium on Computer Science Education V. 1*. 563–569.
- [36] Mark Liffiton, Brad Sheese, Jaromir Savelka, and Paul Denny. 2023. Codehelp: Using large language models with guardrails for scalable support in programming classes. *arXiv preprint arXiv:2308.06921* (2023).
- [37] Dastyi Loksa, Amy J Ko, Will Jernigan, Alannah Oleson, Christopher J Mendez, and Margaret M Burnett. 2016. Programming, problem solving, and self-awareness: Effects of explicit guidance. In *Proceedings of the 2016 CHI conference on human factors in computing systems*. 1449–1461.

- [38] Stephen MacNeil, Paul Denny, Andrew Tran, Juho Leinonen, Seth Bernstein, Arto Hellas, Sami Sarsa, and Joanne Kim. 2023. Decoding Logic Errors: A Comparative Study on Bug Detection by Students and Large Language Models. *arXiv preprint arXiv:2311.16017* (2023).
- [39] Stephen MacNeil, Andrew Tran, Arto Hellas, Joanne Kim, Sami Sarsa, Paul Denny, Seth Bernstein, and Juho Leinonen. [n.d.]. Experiences from Using Code Explanations Generated by Large Language Models in a Web Software Development E-Book. In *Proceedings of the 54th ACM Technical Symposium on Computer Science Education V. 1 (SIGCSE 2023)*. 931–937.
- [40] Stephen MacNeil, Andrew Tran, Joanne Kim, Ziheng Huang, Seth Bernstein, and Dan Mogil. 2023. Prompt Middleware: Mapping Prompts for Large Language Models to UI Affordances. *arXiv preprint arXiv:2307.01142* (2023).
- [41] Stephen MacNeil, Andrew Tran, Juho Leinonen, Paul Denny, Joanne Kim, Arto Hellas, Seth Bernstein, and Sami Sarsa. 2022. Automatically Generating CS Learning Materials with Large Language Models. *arXiv preprint arXiv:2212.05113* (2022).
- [42] Stephen MacNeil, Andrew Tran, Dan Mogil, Seth Bernstein, Erin Ross, and Ziheng Huang. 2022. Generating Diverse Code Explanations Using the GPT-3 Large Language Model. In *Proc. of the 2022 ACM Conf. on Int. Computing Education Research - Volume 2*. ACM, 37–39.
- [43] Danielle Magaldi and Matthew Berler. 2020. Semi-structured Interviews. In *Encyclopedia of Personality and Individual Differences*, Virgil Zeigler-Hill and Todd K. Shackelford (Eds.). Springer International Publishing, Cham, 4825–4830. https://doi.org/10.1007/978-3-319-24612-3_857
- [44] Guillaume Marceau, Kathi Fisler, and Shriram Krishnamurthi. 2011. Mind Your Language: On Novices' Interactions with Error Messages. In *Proceedings of the 10th SIGPLAN Symposium on New Ideas, New Paradigms, and Reflections on Programming and Software* (Portland, Oregon, USA) (*Onward! 2011*). 3–18.
- [45] Robert McCartney, Jonas Boustedt, Anna Eckerdal, Kate Sanders, Lynda Thomas, and Carol Zander. 2016. Why Computing Students Learn on Their Own: Motivation for Self-Directed Learning of Computing. *ACM Trans. Comput. Educ.* 16, 1, Article 2 (Jan 2016), 18 pages. <https://doi.org/10.1145/2747008>
- [46] Tilman Michaeli and Ralf Romeike. 2019. Improving Debugging Skills in the Classroom: The Effects of Teaching a Systematic Debugging Process. In *Proceedings of the 14th Workshop in Primary and Secondary Computing Education* (Glasgow, Scotland, UK) (*WiPSCe '19*). Association for Computing Machinery, New York, NY, USA, Article 15, 7 pages. <https://doi.org/10.1145/3361721.3361724>
- [47] Seyyed Mehdi Nasehi, Jonathan Sillito, Frank Maurer, and Chris Burns. 2012. What makes a good code example?: A study of programming Q&A in StackOverflow. In *2012 28th IEEE International Conference on Software Maintenance (ICSM)*. IEEE.
- [48] Richard S. Newman. 1990. Children's help-seeking in the classroom: The role of motivational factors and attitudes. *Journal of Educational Psychology* 82, 1 (1990), 71–80. <https://doi.org/10.1037/0022-0663.82.1.71>
- [49] Srishti Palani, Zijian Ding, Stephen MacNeil, and Steven P. Dow. 2021. The "Active Search" Hypothesis: How Search Strategies Relate to Creative Learning. In *Proceedings of the 2021 Conference on Human Information Interaction and Retrieval* (Canberra ACT, Australia) (*CHIIR '21*). Association for Computing Machinery, New York, NY, USA, 325–329. <https://doi.org/10.1145/3406522.3446046>
- [50] Ernesto Panadero and Maryam Alqassab. 2019. An empirical review of anonymity effects in peer assessment, peer feedback, peer review, peer evaluation and peer grading. *Assessment & Evaluation in Higher Education* 44, 8 (2019), 1253–1278.
- [51] James Prather, Paul Denny, Juho Leinonen, Brett A Becker, Ibrahim Albluwi, Michael E Caspersen, Michelle Craig, Hieke Keuning, Natalie Kiesler, Tobias Kohn, et al. 2023. Transformed by Transformers: Navigating the AI Coding Revolution for Computing Education: An ITiCSE Working Group Conducted by Humans. In *Proceedings of the 2023 Conference on Innovation and Technology in Computer Science Education V. 2*. 561–562.
- [52] James Prather, Paul Denny, Juho Leinonen, Brett A Becker, Ibrahim Albluwi, Michelle Craig, Hieke Keuning, Natalie Kiesler, Tobias Kohn, Andrew Luxton-Reilly, et al. 2023. The robots are here: Navigating the generative ai revolution in computing education. *arXiv preprint arXiv:2310.00658* (2023).
- [53] James Prather, Brent N. Reeves, Paul Denny, Brett A. Becker, Juho Leinonen, Andrew Luxton-Reilly, Garrett Powell, James Finnie-Ansley, and Eddie Antonio Santos. 2023. "It's Weird That it Knows What I Want": Usability and Interactions with Copilot for Novice Programmers. *ACM Transactions on Computer-Human Interaction* 31, 1 (Nov. 2023), 1–31. <https://doi.org/10.1145/3617367>
- [54] Thomas W Price, Yihuan Dong, and Dragan Lipovac. 2017. iSnap: towards intelligent tutoring in novice programming environments. In *Proc. of the 2017 ACM SIGCSE Technical Symposium on computer science education*. 483–488.
- [55] Yizhou Qian and James Lehman. 2017. Students' Misconceptions and Other Difficulties in Introductory Programming: A Literature Review. *ACM Trans. Comput. Educ.* 18, 1, Article 1 (oct 2017), 24 pages. <https://doi.org/10.1145/3077618>
- [56] Yanyan Ren, Shriram Krishnamurthi, and Kathi Fisler. 2019. What Help Do Students Seek in TA Office Hours?. In *Proceedings of the 2019 ACM Conference on International Computing Education Research* (Toronto ON, Canada) (*ICER '19*). Association for Computing Machinery, New York, NY, USA, 41–49. <https://doi.org/10.1145/3291279.3339418>
- [57] Allison M. Ryan and Huiyoung Shin. 2011. Help-seeking tendencies during early adolescence: An examination of motivational correlates and consequences for achievement. *Learning and Instruction* 21, 2 (2011), 247–256. <https://doi.org/10.1016/j.learninstruc.2010.07.003> Special Section I: Solving information-based problems: Evaluating sources and information Special Section II: Stretching the limits in help-seeking research: Theoretical, methodological, and technological advances.
- [58] Sami Sarsa, Paul Denny, Arto Hellas, and Juho Leinonen. 2022. Automatic Generation of Programming Exercises and Code Explanations Using Large Language Models. In *Proc. of the 2022 ACM Conf. on Int. Computing Education Research - Volume 1*. ACM, 27–43.
- [59] Jaromir Savelka, Arav Agarwal, Marshall An, Chris Bogart, and Majd Sakr. 2023. Thrilled by Your Progress! Large Language Models (GPT-4) No Longer Struggle to Pass Assessments in Higher Education Programming Courses. (2023), 78–92. <https://doi.org/10.1145/3568813.3600142>
- [60] Jaromir Savelka, Arav Agarwal, Christopher Bogart, and Majd Sakr. 2023. Large Language Models (GPT) Struggle to Answer Multiple-Choice Questions about Code. *arXiv:2303.08033 [cs.CL]*
- [61] Jaromir Savelka, Arav Agarwal, Christopher Bogart, Yifan Song, and Majd Sakr. 2023. Can Generative Pre-trained Transformers (GPT) Pass Assessments in Higher Education Programming Courses? *arXiv preprint arXiv:2303.09325* (2023).
- [62] Mohd Sharif and AN Zainab. 2004. Undergraduates in computer science and information technology using the Internet as a Resource. *Malaysian Journal of Library & Information Science* 9, 1 (2004), 1–16.
- [63] Anselm L Strauss and Juliet Corbin. 2004. Open coding. *Social research methods: A reader* (2004), 303–306.
- [64] Jerry Ed Suls and Thomas Ashby Ed Wills. 1991. *Social comparison: Contemporary theory and research*. Lawrence Erlbaum Associates, Inc.
- [65] Andrew Tran, Linxuan Li, Egi Rama, Kenneth Angelikas, and Stephen MacNeil. 2023. Using Large Language Models to Automatically Identify Programming Concepts in Code Snippets. In *Proc. of the 2023 ACM Conf. on Int. Computing Education Research - Volume 2*, Vol. 1. ACM, 563–569. <https://doi.org/10.1145/3568812.3603482>
- [66] Bogdan Vasilescu, Vladimir Filkov, and Alexander Serebrenik. 2013. Stackoverflow and github: Associations between software development and crowdsourced knowledge. In *2013 International Conference on Social Computing*. IEEE, 188–195.
- [67] Elizabeth Wirtz, Amy Dunford, Edward Berger, Elizabeth Briody, Gireesh Guruprasad, and Ryan Senkpeil. 2018. Resource usage and usefulness: academic help-seeking behaviours of undergraduate engineering students. *Australasian Journal of Engineering Education* 23, 2 (2018), 62–70. <https://doi.org/10.1080/22054952.2018.1525889> *arXiv:https://doi.org/10.1080/22054952.2018.1525889*
- [68] Jiaqi Yin, Tiong-Thye Goh, Bing Yang, and Yang Xiaobin. 2021. Conversation Technology With Micro-Learning: The Impact of Chatbot-Based Learning on Students' Learning Motivation and Performance. *Journal of Educational Computing Research* 59, 1 (2021), 154–177. <https://doi.org/10.1177/0735633120952067>
- [69] Carol Zander, Jonas Boustedt, Anna Eckerdal, Robert McCartney, Kate Sanders, Jan Erik Moström, and Lynda Thomas. 2012. Self-Directed Learning: Stories from Industry. In *Proceedings of the 12th Koli Calling International Conference on Computing Education Research* (Koli, Finland) (*Koli Calling '12*). Association for Computing Machinery, New York, NY, USA, 111–117. <https://doi.org/10.1145/2401796.2401810>
- [70] Cynthia Zastudil, Magdalena Rogalska, Christine Kapp, Jennifer Vaughn, and Stephen MacNeil. 2023. Generative AI in Computing Education: Perspectives of Students and Instructors. *arXiv preprint arXiv:2308.04309* (2023).
- [71] Yunfeng Zhang, Q Vera Liao, and Rachel KE Bellamy. 2020. Effect of confidence and explanation on accuracy and trust calibration in AI-assisted decision making. In *Proceedings of the 2020 conference on fairness, accountability, and transparency*. 295–305.