Early Adoption of Generative Artificial Intelligence in Computing Education: Emergent Student Use Cases and Perspectives in 2023

C. Estelle Smith
Colorado School of Mines,
Computer Science
Golden, CO, USA
estellesmith@mines.edu

Kylee Shiekh
Colorado School of Mines,
Physics
Golden, CO, USA
knshiekh@mines.edu

Hayden Cooreman
Colorado School of Mines,
Computer Science
Golden, CO, USA
hcooreman@mines.edu

Sharfi Rahman
Colorado School of Mines,
Computer Science
Golden, CO, USA
sharfirahman@mines.edu

Yifei Zhu
Colorado School of Mines,
Computer Science
Golden, CO, USA
zhu1@mines.edu

Md Kamrul Siam
Colorado School of Mines,
Computer Science
Golden, CO, USA
siam@mines.edu

Michael Ivanitskiy
Colorado School of Mines,
Applied Mathematics & Statistics
Golden, CO, USA
mivanits@mines.edu

Ahmed M. Ahmed
Colorado School of Mines,
Geophysics
Golden, CO, USA
ahmedmohamedahmed@mines.edu

Michael Hallinan
Colorado School of Mines,
Chemical & Biological Engineering
Golden, CO, USA
mhallinan@mines.edu

Alexander Grisak
Colorado School of Mines,
Mechanical Engineering
Golden, CO, USA
agrisak@mines.edu

Gabe Fierro
Colorado School of Mines,
Computer Science
Golden, CO, USA
gtfierro@mines.edu

ABSTRACT

Because of the rapid development and increasing public availability of Generative Artificial Intelligence (GenAI) models and tools, educational institutions and educators must immediately reckon with the impact of students using GenAI. There is limited prior research on computing students' use and perceptions of GenAI. In anticipation of future advances and evolutions of GenAI, we capture a snapshot of student attitudes towards and uses of yet emerging GenAI, in a period of time before university policies had reacted to these technologies. We surveyed all computer science majors in a small engineering-focused R1 university in order to: (1) capture a baseline assessment of how GenAI has been immediately adopted by aspiring computer scientists; (2) describe computing students' GenAI-related needs and concerns for their education and careers; and (3) discuss GenAI influences on CS pedagogy, curriculum, culture, and policy. We present an exploratory qualitative analysis of this data and discuss the impact of our findings on the emerging conversation around GenAI and education.

ITiCSE 2024, July 8–10, 2024, Milan, Italy
© 2024 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-0600-4/24/07.
https://doi.org/10.1145/3649217.3653575

CCS CONCEPTS

• **Social and professional topics → Computing literacy**; **CS1**; **Automation**; **Socio-technical systems**.

KEYWORDS

Generative artificial intelligence; large language model; code generator; image generator; interactive tutoring; policy; survey; student experience; AI literacy; education

ACM Reference Format:
C. Estelle Smith, Kylee Shiekh, Hayden Cooreman, Sharfi Rahman, Yifei Zhu, Md Kamrul Siam, Michael Ivanitskiy, Ahmed M. Ahmed, Michael Hallinan, Alexander Grisak, and Gabe Fierro. 2024. Early Adoption of Generative Artificial Intelligence in Computing Education: Emergent Student Use Cases and Perspectives in 2023. In *Proceedings of the 2024 Innovation and Technology in Computer Science Education V. 1 (ITiCSE 2024), July 8–10, 2024, Milan, Italy.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3649217.3653575

1 INTRODUCTION

Given the increasing public availability of Generative Artificial Intelligence (GenAI), today's computing students now have immediate access to a new class of tools that stand to transform their learning outcomes and career prospects. While existing educational tools have been built and studied for targeted purposes such as tutoring [1], visualizing [28], or explaining code [26], use of these tools typically lies behind curated learning experiences such as lab sessions. In contrast, GenAI is *general purpose* and can generate many different types of content (*e.g.,* text, code, images, music, speech, music) using only natural language prompts, and outside of designated educational contexts. GenAI offers a lower barrier to entry than existing fit-for-purpose tools and can be accessed by students without the instructor as an interlocutor. This informs our guiding question: *how did students make use of emerging GenAI tools and services, and how can we best support their future educational and career needs with regard to GenAI?*

Since the release of OpenAI's ChatGPT in November 2022, GenAI-based technologies have rapidly entered the public consciousness, with extensive impacts across education and industry [18, 24]. In CS education, concerns have been raised for how the code-generating capacity of GenAI might interfere with the learning process at the heart of many CS classrooms [11]. According to the US Department of Education, AI should be designed in alignment with modern learning principles [3]; many curricula are actively integrating AI literacy [4, 6]. Institutions are now also releasing guidelines for GenAI, with some banning or allowing GenAI indiscriminately, and others leaving it to instructors' discretion [5].

As key impacted stakeholders, students' needs and concerns should be central to the development of GenAI policies and tools. Computing educators and researchers may have intuitions, assumptions, and concerns about how students consider and use GenAI. However, rigorous examination of emergent use cases is necessary for evidence-based tool adoption and policy formation, as well as scientific assessment of the impacts of GenAI on CS education. Prather et al. provide an insightful initial summary of the emerging literature on perceptions, uses, and risks of on GenAI in CS education, including rapid uptake of text and code generation-related tasks [24]. However, student usages of GenAI will shift and change over time in nuanced ways as GenAI-based tools, policies, and professional norms continue to evolve. In order to rigorously assess and understand the changing landscape of *types* and *rates* of GenAI usage, the following two resources are needed: a delineated taxonomy of specific use cases; and preliminary benchmarks for the uptake of these use cases. Thus, to capture an invaluable historical snapshot of student behaviors shortly after the release of ChatGPT, we pose two research questions:

- **RQ1:** With limited guidelines, guardrails, or pre-planning, how did computing students adopt GenAI-based tools during the Spring 2023 academic semester?
- **RQ2:** How do computing students envision the role of GenAI within their education and future careers?

To address these questions, we surveyed all computer science majors at a small (< 10k students) engineering-focused R-1[1] university in the USA. We found that most students have tried GenAI tools (*esp.* LLMs) for a variety of writing, coding, and learning use cases. Moreover, students tend to view GenAI tools as beneficial to the field of computing. In our discussion, we synthesize these results to discuss how educators can optimize GenAI-based policies and tools for the educational and professional needs of students.

2 RELATED WORK

Existing Tooling in CS Education. CS education has long grappled with questions around the nature and role of automation and tools in education—*e.g.,* Online Python Tutor [14], interactive E-books [27] or algorithm visualizations [28], and web-based AI/ML literacy tools [8]. Despite increasing publications on AI literacy since 2018 [33], there is limited research on GenAI, since such tools only became publicly available in 2022 [24]. GenAI immediately evokes comparisons to work on intelligent tutoring systems (ITS) [1] and computer-aided instruction. ITS research has focused on data-driven improvement of feedback [25] and investigating interactions between students and "tutors" [2, 20]. Conversational agents are another tool that employ intuitive natural language dialogues [26]. These agents can provide instant and informative responses [31], improve student comprehension [21], and offer personalized assistance that can be difficult for human instructors [12]. Conversational STEM tutoring agents have yielded learning gains comparable to trained human tutors [13]. Bayesian Knowledge Tracing can be also used to model each learner's mastery and improve predictions of student success [35].

Assistive tools can improve student experience, but they might also hinder effective learning. One routine concern for CS educators is "contract cheating" through use of websites like Chegg.com or StackOverflow.com to copy solutions without learning the material [19]. Widespread availability of GenAI systems drastically re-frames many of these prior questions and concerns. Not only are GenAI systems capable of synthesis across a wider range of tasks, but their speed of adoption has given educational institutions little time to respond thoughtfully. These systems can also confidently present incorrect information in a manner that previous tools do not—thus transmitting false beliefs to human users [16].

The Emergence of Generative AI. We consider three broad categories of GenAI: (1) Large Language Model (LLM) chatbots (*e.g.,* ChatGPT, Bard, Bing Chat) in standalone conversational user interfaces; (2) LLM Code Generators (*e.g.,* GitHub Co-Pilot), which are code generation & auto completion tools integrated within code development environments; (3) Image Generators (*e.g.,* Dall-E, Midjourney). GenAI models can generate natural language text that imitates human text with high levels of coherence, complexity, and diversity [22]. The novelty of GenAI tools and the diversity of tasks they can perform, combined with the unexplainability of AI [32], has created difficulty in informing educators about how to interact with GenAI tools. Although some instructors have used them to assist with tasks like lesson planning or creating rubrics [10], most instructors have not adopted GenAI [24]. Educators generally lack understanding about the functionality, limitations, and usage of these tools and are struggling to catch up with students who are exploring GenAI on their own [30, 36]. Some institutions are already adopting custom GenAI tools in computing classes [23]. There is an urgent need for research into understanding the role of GenAI in computing education to encourage positive learning outcomes.

Student-Centered Policy Development. Institutions must develop policies to address AI concerns [9]. The US Department of Education issued a 2023 report emphasizing the need for designing AI interventions based on modern learning principles, strengthening trust, involving educators, appropriately addressing contextual considerations, and developing effective guidelines and guardrails [3]. Given the potential benefits and risks, guidance on the responsible use of GenAI in particular is now needed in this transitional time [7, 17].

[1]According to the Carnegie Classification of Institutions of Higher Education, R-1 indicates universities that offer doctoral degrees and have "very high research activity."

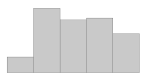
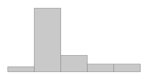
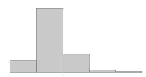
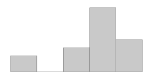
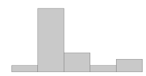
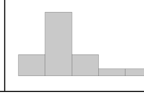
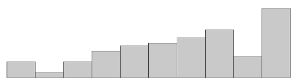
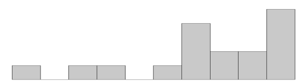
	Undergraduate Students			Graduate Students		
Total Respondents	Total: N = 116			MS: 12	PhD: 5	Total: N = 17
	Years Enrolled	N (% of Undergrad Sample)		Years Enrolled	N (% of Grad Sample)	
	0-1	33 (28.4%)		0-1	3 (17.6%)	
	1-2	43 (37.1)		1-2	2 (11.8%)	
	2-3	21 (18.1%)		2-3	1 (5.8%)	
	3-4	18 (15.5%)		3-4	6 (35.3%)	
	4+	1 (0.8%)		4+	5 (29.4%)	
Frequency of Use	LLM	Code Generator	Image Generator	LLM	Code Generator	Image Generator
X-axis from left to right: only for fun or curiosity; never; once or twice ever; regularly (once or twice/week); nearly everyday						
Perceived Benefit to CS 1 (left): extremely damaging 10 (right):extremely beneficial						

Table 1: Summary of participant demographics and key quantitative survey questions. Histograms are included for visual synthesis of students' frequency of use of GenAI and their ratings of how beneficial GenAI will be to the field of computer science; the results section reports numbers of participants in each category.

Some US institutions have already released guidelines on using GenAI in the classroom [5]. However, it is unknown whether or how student perspectives have been considered during their development. We position the perspectives, needs, and concerns of today's computing students as integral to the formation of GenAI-related policies and tooling environments because of their role in the efficacy, alignment, and facilitation of these policies. At the time of the survey, our institution had not yet released institution-wide policies. This study systematically captures and assesses the uses of GenAI by students at the end of the Spring 2023 semester, amidst the initial wave of excitement around GenAI.

3 METHODS

This study was reviewed and deemed exempt from IRB. Here we describe our survey design, sample, recruitment, and analysis.

Survey Design. The first author and one student assistant first wrote, edited, and uploaded an initial survey draft to QuestionPro software. To identify confusing question phrasing, technical bugs, and gaps in question coverage, we then piloted and refined the survey with members of our research team as well as 3 administrators from our center for educational innovation, leading to a smooth and error-free deployment. The final survey included: consent and eligibility information; rating of frequency of use of LLMs, code generators, and image generators for classes or professional efforts; questions about whether classes had GenAI policies in Sp23 and (for student TAs only) if they suspected AI-generated submissions; Likert ratings of whether Gen-AI will be extremely harmful (1) to extremely beneficial (10) for computing; and three free response questions on uses and perspectives of GenAI. §4 contains the verbatim free response question text.

Sample Selection. Due to the sudden public release of GenAI tools in late 2022, universities had no lead time for conscientious preparations. We felt it urgent to capture a snapshot of emergent use cases ASAP after the release of ChatGPT, both for accurate historical benchmarking, as well as to provide input to our administration prior to Fall23. Although a large scale survey deployment across many schools and departments would have been ideal, this would have required months of preparation and coordination across many administrations, preventing us from achieving our goals. Therefore, we decided to survey all CS majors at our institution as early as we possibly could–the tail end of Sp23.We chose all CS majors rather than students in any CS classes because: (1) the CS dept granted us access to list-servs for all undergraduate and graduate CS majors, offering a convenient and effective recruitment technique; (2) CS majors have a high degree of commitment to and/or experience with computing and should be highly qualified to speak to tooling and policy in CS education; (3) students can arrive in the CS major at our university, enabling us to collect responses from freshman undergrads up through advanced PhDs.

Recruitment. We recruited participants by sending list-serv emails. We collected no identifying information and only two demographic details (undergrad, masters, or doctoral status; # years enrolled). At the end, participants could opt-in to complete a separate form and enter their email in a drawing for one of four $25 gift cards. Table 1 summarizes our participants (and two key quantitative questions). We received 133 responses from eligible consenting students. N = 116 (87.2%) are undergraduate students (equating to 12% of undergraduates in the dept). N = 17 (12.8%) are graduate students (7.6% of graduates in the dept).

Survey Analysis. Data were exported from QuestionPro into a CSV file. We used standard `pandas` and `scikit-learn` Python packages to compute descriptive statistics and perform statistical tests. Prior surveys on the adoption of GenAI in CS education have used thematic analysis [24, 29]. To complement prior work, we chose to instead use directed content analysis to analyze free response questions [15]. To support our goal of providing a high quality historical benchmark measure of the adoption of GenAI, this method ensures rigorous development and definition of codebooks for labeling and counting instances of concepts appearing in natural language data. Across six rounds of iteration, six human coders manually coded subsets of the data, continuously discussing disagreements and refining code definitions until consensus was achieved. We used

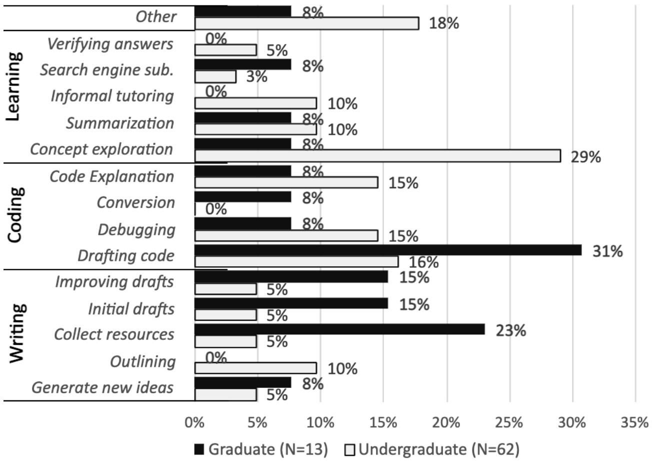


Figure 1: Summary of RQ1 codes applied.

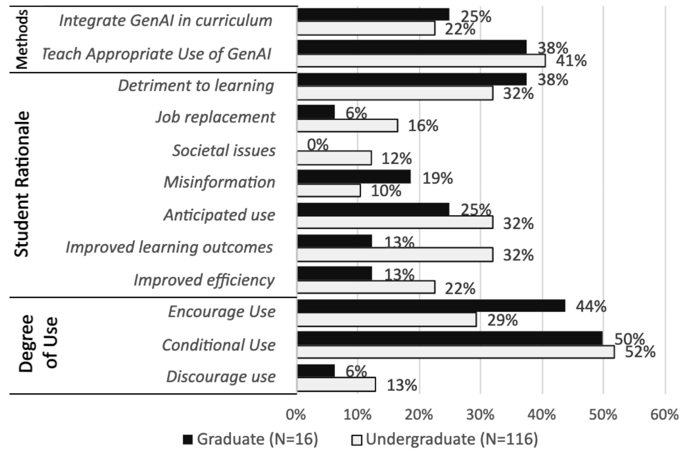


Figure 2: Summary of RQ2 codes applied.

Inter-Rater Reliability (IRR) scores (Krippendorff's alpha) to guide refinements until all IRR scores were greater than 0.6 (a threshold establishing good agreement). Finally, individual coders coded all responses according to the finalized codebooks. Throughout results, **bold typesetting** indicates a high-level codebook category; *italic typesetting* indicates an individual code.

Threats to Validity. We acknowledge the limitation that our sample is small and not statistically representative, therefore it may have limited generalizability to other institutions or majors. Standard survey limitations also apply, including opt-in bias and possibly inaccurate self-assessments. Students may have withheld or misrepresented information about behaviors perceived as cheating; to counteract this, we used messaging encouraging honesty because participation had no ties to their identity. In light of these limitations, we position our results primarily as qualitative and exploratory—*i.e.,* the distributions of codes applied by our analysis may or may not be representative of the true distributions. Nonetheless, our codebooks themselves *do* accurately describe and capture students' emergent use cases and perspectives. Our methodology provides a rigorous and carefully-executed benchmark that will offer future researchers a strong point of comparison. Our survey and codebooks can be re-used in future work to analyze larger samples across different types and sizes of institutions. To support future replications, our verbatim survey questions and complete codebooks, including code definitions with data examples, are available as supplemental materials at bit.ly/GenAICodebook.

4 RESULTS

4.1 RQ1: Students' adoption of GenAI in Sp23

Students reported using LLM chatbots more frequently than code or image generators. For instance, $N = 24$ (18.0%) of students use LLMs everyday; $N = 36$ (27.1%) regularly (once or twice per week); $N = 30$ (22.6%) once or twice ever; $N = 33$ (24.8%) never; or $N = 10$ (7.5%) only for fun or curiosity. Fewer have used code generators; $N = 11$ (8.3%) of students use code generators everyday; $N = 10$ (7.5%) regularly; $N = 22$ (16.5%) once or twice ever; $N = 83$ (62.4%) never; or $N = 7$ (5.3%) only for fun. Interestingly, only 36.1% have ever reported trying an image generator, thus we omit further discussion of image generators.

12.8% of participants reported that at least one class had a formal GenAI policy in the syllabus; 23.3% that instructors stated a formal policy *not* in the syllabus; 30.1% that instructors mentioned only loose guidance; and 33.8% that there was no discussion of GenAI. Of $N = 31$ students who indicated that they were TAs, $N = 13$ (41.9%) did not believe they had encountered AI-completed assignments; $N = 11$ (35.5%) were unsure; and $N = 7$ (22.6%) believe or know they received AI-generated assignments.

In order to understand how computing students used GenAI, we asked the following optional question: *If you have ever used any GenAI-based tool(s) for your classes, research, and/or professional efforts, please tell us about how you have used them, why you used them, and how you feel about your use of these tool(s).* 75 students (56.4% of the full sample) submitted an answer. Figure 1 visualizes code distributions with normalized percentages of responses from undergrad *v.s.* graduate students; the remainder of results reports counts of codes applied across all respondents.

No respondents reported using GenAI to fully complete assignments for them. Rather, they described how their use of GenAI tools supported three different categories of use cases for **learning**, **coding**, and **writing**. We labeled the **context** of these use cases, and found that the majority were either *academic* ($N = 29$) (*i.e.,* for coursework or research) or *unknown/not-specified* ($N = 31$). Although no participants described purely *professional* use cases, some reported *mixed* ($N = 7$) contexts spanning academic/professional. $N = 1$ respondent described a *personal* use context.

Learning: One key result is that 32 respondents described uses of GenAI for self-described improvement of their learning. The code *conceptual exploration* ($N = 19$) was highly prominent: students used LLMs to ask about course topics and gain supplementary perspectives or alternative wordings and thought processes from those provided by instructors, thereby deepening their knowledge. Examples include: *"Sometimes I ask it about a concept in class and it explains it to me and knows how to dumb it down for me."*; *"I used it to explain a topic that our professor didn't speak much*

Detriment to Learning & Authenticity	Improved Learning Outcomes
"I believe that as these AIs become smarter and smarter, they will be very harmful to learning environments. I worry that AI will simply write it's own programs, and that software engineering will become completely obsolete in the next 3 decades, as the last line of human code written will be putting the finishing touches on an AI coder."	*"AI is really powerful for learning, such as if you don't understand a math concept, you can ask it to explain how to do it, and ask for more and more details if needed. It is good for inspiration but shouldn't be used to talk for you. I've successfully used it many times to decipher an error message, explain functionality, and even tried to get it to write code blocks."*
"It is too powerful of a tool and can be too much of a crutch that students can rely on. I have known students who have used it on every assessment, very blindly following whatever instructions it gave. They couldn't even justify why it was wrong or identify when it had made a flagrant error."	*"Instructors should encourage the use of AI as a tool to enhance a student's education, not as a tool that does the student's task of learning for them. Students should use AI to help debug their code, learn from their mistakes, and learn new programming techniques and tools, not to generate all the code for them."*
"It will limit the growth and knowledge someone would be able to achieve without the same capabilities."	*"It serves as a valuable learning tool that helps students understand complex concepts, generate ideas, and receive feedback."*

Table 2: RQ2 examples of student data demonstrating tensions between the potential for GenAI to damage or improve learning.

about."; *"[ChatGPT] helps to explain thought processes behind hard-to-understand concepts."* Students also mentioned *summarization* ($N = 7$) of lengthy documents and *informal tutoring* ($N = 6$) as techniques that helped them digest and interact with course materials, esp. when instructors were unavailable. For example, *"Sometimes [it is] hard to contact professors to get help with homework so it's nice to have something to help."* Some students also used GenAI as a *search engine substitute* ($N = 3$) or as tool for *verifying answers* ($N = 3$) rather than generating them.

Coding: 23 respondents reported using GenAI to help them understand, create, or fix code. $N = 14$ respondents reported *drafting code* that they could then verify, modify, and complete. Examples include: *"I use ChatGPT and Copilot regularly to help write code quicker for many more mundane implementation tasks."*; *"I view GPT as a calculator for coding."* Another common use case was *explanation* ($N = 10$), *i.e.,* providing code snippets to an LLM for an explanation of code behavior. For example, *"I used Co-Pilot on a bit of code I was stuck on, and then I used ChatGPT to explain why Co-Pilot did what it did."* Students also used GenAI for *debugging* ($N = 10$) to find and fix bugs in their own code. *"I use ChatGPT to help me debug my code. it's quicker than crawling through stackoverflow forums and it gives a very detailed explanation of why my code is wrong and how the new way is better so i feel like i'm learning."* $N = 1$ respondent also used GenAI for *conversion* of code from one language to another.

Writing: 21 respondents used GenAI tools for writing support. $N = 4$ students used GenAI to *generate new ideas for consideration* to help them overcome writer's block, diversify their thinking, or figure out what to write about. $N = 6$ mentioned *collecting resources* to support their writing—*e.g., "If I formed a new point or thought that I didn't find a quote for on my first round annotating, ChatGPT would be very useful to help me find useful evidence or decide on a quote to integrate into my paper."* They also described use cases to help with the mechanics of writing, including *outlining* ($N = 6$), *generating initial drafts* ($N = 5$), or *improving drafts* ($N = 5$) of papers and emails. For example, *"Sending an email to my professor vs. sending an email to a friend will be different in terms of format and word choice, and Quillbot helps in this kind of scenario."*[2]

[2]Quillbot is a writing assistant tool that is built on LLMs.

4.2 RQ2: The role of GenAI in CS Education

We asked students to rate how beneficial they feel GenAI will be to computer science. Table 1 visualizes the distribution of ratings. We observe that the distributions tend towards more positive evaluations: the average undergraduate rating is 6.78 ($SD = 2.62$) and grad rating is 7.41 ($SD = 2.67$).

We asked two additional required free response questions: *What do you think the role of generative AI should be in higher education? For example, should instructors be trying to encourage or prohibit use of GenAI for students' coursework? How and why, or why not?*; and *What GenAI-related concerns do you have with regard to the workplace you will soon be entering, and how do you want instructors to prepare you for this workplace?* We concatenated answers to these two questions and applied codes across both since there was substantial overlap in the content of these responses. All 133 participant answers were included in this analysis. Figure 2 visualizes the distribution of codes applied. We coded three categories for students' desired **methods of implementation** to serve their educational needs, the **rationale** behind their opinions, and their desired **degree of use** of GenAI in education.

Methods of implementation: 70 respondents mentioned methods such as *teaching professional use of GenAI* ($N = 53$) to prepare students with the specific GenAI-related skills that they will need for their future workplaces and/or *integrating GenAI in the curriculum* ($N = 30$) to cultivate effective and appropriate uses of GenAI to support learning during their education.

Student rationale: 111 responses also included rationales for their opinions. Importantly, there are split opinions on how GenAI may impact learning (see Table 2 for data examples): $N = 43$ students were concerned about GenAI's potential *detriment to learning and authenticity*, whereas $N = 39$ felt GenAI could *improve learning outcomes* through the types of use cases described in RQ1. Many students *anticipated use* ($N = 41$) of GenAI tools in their future careers and felt it would be necessary for them to learn them in order to be competitive and effective at their jobs. Yet students also voiced concerns about *misinformation* ($N = 15$) produced by GenAI, *societal issues* ($N = 14$) such as unethical use, intellectual property violations or plagiarism, privacy breaches, and unfair advantages or equitable access issues, or *job replacement* ($N = 20$).

Degree of use: 126 respondents reported opinions on what degree of GenAI usage would be most appropriate. Most called for *conditional use* ($N = 68$), meaning instructors should specify under what circumstances GenAI use is allowed, appropriate, and ethical. Others wanted to *encourage use* ($N = 41$) without restrictions, while only a few wanted instructors to entirely *discourage use* ($N = 16$).

Exploratory Statistical Analysis. We queried whether any of the manually applied codes were interrelated using Pearson correlation coefficients; no coefficients exceeded 0.314, suggesting that the codes are not significantly correlated. We used chi-squared tests of independence to examine relationships between qualitative codes and quantitative responses and report on interesting relationships with $p < 0.01$. We found that students who used LLMs more frequently were more likely to rate the benefit of GenAI more highly ($p < .0008$) and to have RQ2 responses coded with *encourage use* or *conditional use* ($p < 0.002$). Students who have not used GenAI provided lower ratings of its benefits ($p < .0001$) and were more likely to have RQ2 responses coded with *discourage use* ($p < .0003$).

5 DISCUSSION

Our results capture a valuable snapshot of student attitudes toward and usage of nascent GenAI tools that stand to transform CS education. Prior work such as [24, 29] suggests that many CS students adopted GenAI for working with text or code, whereas instructors did not; (2) student and instructor perspectives diverge on the clarity of university policies and how well GenAI can provide good coursework guidance; and (3) students and instructors generally align on expecting and needing to use GenAI tools for future success. Complementing prior work, our results show that many CS students rapidly adopted GenAI before the end of Sp23 in our CS department. Moreover, through rigorous derivation and application of our codebooks, we contribute a fine-grained benchmark measure of specific use cases and perspectives that emerged at a time when the capabilities of these GenAI are still being explored and university policies have yet to adapt. In this section, we discuss how our results can inform future GenAI-aware approaches to the educational and career needs of students.

Influences on pedagogy. Our taxonomy distinguishes three major categories for GenAI to support student **learning**, **coding**, and **writing** (Figure 1). Students indicate they view GenAI usage as *"inevitable"* or *"the new google"*. They tend to view GenAI as beneficial to CS, with many seeing GenAI as a *"parallel colleague."* However, our work has also surfaced a central tension between the desire to explore new technologies and the now blurry boundaries imposed by academic integrity policies. Students who have used GenAI shared more positive use cases and experiences, but are split as to what extent GenAI use should be guided or restricted, and to what extent it may improve or impair learning.

One way to resolve this tension is to encourage students to engage with GenAI through constructivist pedagogy. In particular, variations on inquiry-based learning can frame how students experiment with different inputs to GenAI tools and – importantly – verify or evaluate their outputs. The capabilities of GenAI tools will likely improve in the future, but there are still fundamental issues with respect to "hallucination" in their output as well as correct interpretation of user input. Students are already checking their understanding of topics against the output of GenAI; educators should encourage students to also perform the inverse check.

Influences on curriculum. One first step is to understand and explore the emerging use cases in Fig. 1 and critically assess how such use cases relate to the learning objectives for different courses and developmental stages of students. In our results, students express real interest in receiving training on GenAI tools to prepare them for their future professional environments; however, it is yet unclear when or how GenAI should be introduced in CS curricula. For example, if GenAI were used too early in CS education, students might fully complete course assignments without developing a solid understanding of fundamental course concepts, hindering their ability to approach higher-level concepts later in their curriculum where GenAI might exceed its limit to help. Consequently, an intro CS instructor could hypothetically encourage uses such as *concept exploration*, *code explanation*, or *answer verification* but expressly forbid *drafting code* or *debugging*. A senior-level design course might allow unrestricted use, and encourage students to consider how different inputs to the GenAI ("prompt engineering") can influence the quality or veracity of the output.

Culture and policy. Clarity from instructors and institutions is necessary to address internal and interpersonal tension within the student body on when and how to use GenAI—*i.e.,* what constitutes cheating or academic dishonesty *v.s.* an allowable and helpful use? With GenAI tools now freely available, creating a culture of honesty and accountability is essential to the success of such policies. Moreover, given that some institutions may buy GenAI licenses [34] or develop custom in-house GenAI-based tooling [23], future research and innovation should explore how to enforce guardrails that provide educators with the ability to technically prohibit certain use cases while allowing others. We believe a balanced mix of culture, policy, and tools can enhance educational outcomes, but further research will need to identify the right strategies.

Future work. Future work can use the survey and taxonomies presented in this work to understand how student attitudes toward and uses of GenAI evolve—both at our own as well as other institutions. Our snapshot can serve as a point of reference for future studies, especially as university policies and classroom experiences respond to the still-changing capabilities of GenAI. Future, larger surveys could reveal either broad trends or contextual idiosyncrasies in adoption related to experience level, type or size of institution, geography, or other dimensions. Future work should also examine how the student-provided use cases relate to learning outcomes and objectives of current and future curricula.

ACKNOWLEDGMENTS

We thank all participants, as well as additional research assistant volunteers, James Frishkoff, James Sprow, and Laila Barrett. Thanks to our colleagues Tracy Camp, Kathyrn Stolee, Christine Liebe, and Iris Bahar for feedback and departmental support. Finally, we thank members of the Trefny Innovation Instruction Center, especially Deb Jordan, Megan Sanders, and Carter Moulton.

REFERENCES

[1] John R. Anderson, C. Franklin Boyle, and Brian J. Reiser. 1985. Intelligent Tutoring Systems. *Science* 228, 4698 (April 1985), 456–462. https://doi.org/10.1126/science.228.4698.456

[2] John R. Anderson, Albert T. Corbett, Kenneth R. Koedinger, and Ray. Pelletier. 1995. Cognitive Tutors: Lessons Learned. *Journal of the Learning Sciences* 4, 2 (April 1995), 167–207. https://doi.org/10.1207/s15327809jls0402_2 Publisher: Routledge _eprint: https://doi.org/10.1207/s15327809jls0402_2.

[3] Miguel A Cardona, Roberto J Rodríguez, and Kristina Ishmael. 2023. *Artificial Intelligence and the Future of Teaching and Learning.* Technical Report. U.S. Department of Education, Office of Educational Technology, Washington, DC. 71 pages. https://www2.ed.gov/documents/ai-report/ai-report.pdf

[4] Lorena Casal-Otero, Alejandro Catala, Carmen Fernández-Morante, Maria Taboada, Beatriz Cebreiro, and Senén Barro. 2023. AI literacy in K-12: a systematic literature review. *International Journal of STEM Education* 10, 1 (April 2023), 29. https://doi.org/10.1186/s40594-023-00418-7

[5] Jack Caulfield. 2023. University Policies on AI Writing Tools | Overview & List. https://www.scribbr.com/ai-tools/chatgpt-university-policies/

[6] Ismail Celik. 2023. Towards Intelligent-TPACK: An empirical study on teachers' professional knowledge to ethically integrate artificial intelligence (AI)-based tools into education. *Computers in Human Behavior* 138 (Jan. 2023), 107468. https://doi.org/10.1016/j.chb.2022.107468

[7] Cecilia Ka Yuk Chan. 2023. A comprehensive AI policy education framework for university teaching and learning. *International Journal of Educational Technology in Higher Education* 20, 1 (July 2023), 38. https://doi.org/10.1186/s41239-023-00408-3

[8] Jie Chao, William Finzer, Carolyn Rose, Shiyan Jiang, Rebecca Ellis, Kenia Wiedemann, Cansu Tatar, and James Fiacco. 2023. Teach Artificial Intelligence with StoryQ, A Web-Based Machine Learning and Text Mining Tool for K-12 Students. In *Proceedings of the 54th ACM Technical Symposium on Computer Science Education V. 2 (SIGCSE 2023).* Association for Computing Machinery, New York, NY, USA, 1262. https://doi.org/10.1145/3545947.3573236

[9] Thaddieus W. Conner and Thomas M. Rabovsky. 2011. Accountability, Affordability, Access: A Review of the Recent Trends in Higher Education Policy Research. *Policy Studies Journal* 39, s1 (2011), 93–112. https://doi.org/10.1111/j.1541-0072.2010.00389_7.x _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.1541-0072.2010.00389_7.x.

[10] Grant Cooper. 2023. Examining Science Education in ChatGPT: An Exploratory Study of Generative AI. *Journal of Science Education and Technology* 32, 3 (June 2023), 444–452. https://doi.org/10.1007/s10956-023-10039-y

[11] Paul Denny, James Prather, Brett A. Becker, James Finnie-Ansley, Arto Hellas, Juho Leinonen, Andrew Luxton-Reilly, Brent N. Reeves, Eddie Antonio Santos, and Sami Sarsa. 2023. Computing Education in the Era of Generative AI. http://arxiv.org/abs/2306.02608 arXiv:2306.02608 [cs].

[12] Donn Emmanuel Gonda and Beatrice Chu. 2019. Chatbot as a learning resource? Creating conversational bots as a supplement for teaching assistant training course. In *2019 IEEE International Conference on Engineering, Technology and Education (TALE).* 1–5. https://doi.org/10.1109/TALE48000.2019.9225974 ISSN: 2470-6698.

[13] Arthur C. Graesser, Xiangen Hu, Benjamin D. Nye, Kurt VanLehn, Rohit Kumar, Cristina Heffernan, Neil Heffernan, Beverly Woolf, Andrew M. Olney, Vasile Rus, Frank Andrasik, Philip Pavlik, Zhiqiang Cai, Jon Wetzel, Brent Morgan, Andrew J. Hampton, Anne M. Lippert, Lijia Wang, Qinyu Cheng, Joseph E. Vinson, Craig N. Kelly, Cadarrius McGlown, Charvi A. Majmudar, Bashir Morshed, and Whitney Baer. 2018. ElectronixTutor: an intelligent tutoring system with multiple learning resources for electronics. *International Journal of STEM Education* 5, 1 (April 2018), 15. https://doi.org/10.1186/s40594-018-0110-y

[14] Philip J. Guo. 2013. Online python tutor: embeddable web-based program visualization for cs education. In *Proceeding of the 44th ACM technical symposium on Computer science education (SIGCSE '13).* Association for Computing Machinery, New York, NY, USA, 579–584. https://doi.org/10.1145/2445196.2445368

[15] Hsiu-Fang Hsieh and Sarah E. Shannon. 2005. Three approaches to qualitative content analysis. *Qualitative Health Research* 15, 9 (Nov. 2005), 1277–1288. https://doi.org/10.1177/1049732305276687

[16] Celeste Kidd and Abeba Birhane. 2023. How AI can distort human beliefs. *Science* 380, 6651 (June 2023), 1222–1223. https://doi.org/10.1126/science.adi0248 Publisher: American Association for the Advancement of Science.

[17] 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 *ICER.* ICER, Chicago, IL.

[18] Weng Marc Lim, Asanka Gunasekara, Jessica Leigh Pallant, Jason Ian Pallant, and Ekaterina Pechenkina. 2023. Generative AI and the future of education: Ragnarök or reformation? A paradoxical perspective from management educators. *The International Journal of Management Education* 21, 2 (July 2023), 100790. https://doi.org/10.1016/j.ijme.2023.100790

[19] Sathiamoorthy Manoharan and Ulrich Speidel. 2020. Contract Cheating in Computer Science: A Case Study. In *2020 IEEE International Conference on Teaching, Assessment, and Learning for Engineering (TALE).* 91–98. https://doi.org/10.1109/TALE48869.2020.9368454 ISSN: 2470-6698.

[20] Douglas C. Merrill, Brian J. Reiser, Michael Ranney, and J. Gregory Trafton. 1992. Effective Tutoring Techniques: A Comparison of Human Tutors and Intelligent Tutoring Systems. *Journal of the Learning Sciences* 2, 3 (July 1992), 277–305. https://doi.org/10.1207/s15327809jls0203_2

[21] Chinedu Wilfred Okonkwo and Abejide Ade-Ibijola. 2020. Python-Bot: A Chatbot for Teaching Python Programming. *Engineering Letters* 29, 1 (March 2020), 25–34.

[22] OpenAI. 2023. GPT-4 Technical Report. https://doi.org/10.48550/arXiv.2303.08774 arXiv:2303.08774 [cs].

[23] Jimmy Pezzone. 2023. Harvard is bringing its own brand of generative AI to the classroom. https://www.techspot.com/news/99185-harvard-bringing-own-brand-generative-ai-classroom.html

[24] James Prather, Paul Denny, Juho Leinonen, Brett A. Becker, Ibrahim Albluwi, Michelle Craig, Hieke Keuning, Natalie Kiesler, Tobias Kohn, Andrew Luxton-Reilly, Stephen MacNeil, Andrew Petersen, Raymond Pettit, Brent N. Reeves, and Jaromir Savelka. 2023. The Robots Are Here: Navigating the Generative AI Revolution in Computing Education. In *Proceedings of the 2023 Working Group Reports on Innovation and Technology in Computer Science Education (ITiCSE-WGR '23).* Association for Computing Machinery, New York, NY, USA, 108–159. https://doi.org/10.1145/3623762.3633499

[25] Thomas Price, Rui Zhi, and Tiffany Barnes. 2017. *Evaluation of a Data-Driven Feedback Algorithm for Open-Ended Programming.* Technical Report. International Educational Data Mining Society. https://eric.ed.gov/?id=ED596568 Publication Title: International Educational Data Mining Society ERIC Number: ED596568.

[26] José Quiroga Pérez, Thanasis Daradoumis, and Joan Manuel Marquès Puig. 2020. Rediscovering the use of chatbots in education: A systematic literature review. *Computer Applications in Engineering Education* 28, 6 (2020), 1549–1565. https://doi.org/10.1002/cae.22326 _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/cae.22326.

[27] Zak Risha, Jordan Barria-Pineda, Kamil Akhuseyinoglu, and Peter Brusilovsky. 2021. Stepwise Help and Scaffolding for Java Code Tracing Problems With an Interactive Trace Table. In *Proceedings of the 21st Koli Calling International Conference on Computing Education Research (Koli Calling '21).* Association for Computing Machinery, New York, NY, USA, 1–10. https://doi.org/10.1145/3488042.3490508

[28] Henry W. Robbins, Samuel C. Gutekunst, David B. Shmoys, and David P. Williamson. 2023. GILP: An Interactive Tool for Visualizing the Simplex Algorithm. In *Proceedings of the 54th ACM Technical Symposium on Computer Science Education V. 1 (SIGCSE 2023).* Association for Computing Machinery, New York, NY, USA, 108–114. https://doi.org/10.1145/3545945.3569815

[29] Michael P. Rogers, Hannah Miller Hillberg, and Christopher L. Groves. 2024. Attitudes Towards the Use (and Misuse) of ChatGPT: A Preliminary Study. In *Proceedings of the 55th ACM Technical Symposium on Computer Science Education V. 1.* ACM, Portland OR USA, 1147–1153. https://doi.org/10.1145/3626252.3630784

[30] Kyoungwon Seo, Joice Tang, Ido Roll, Sidney Fels, and Dongwook Yoon. 2021. The impact of artificial intelligence on learner–instructor interaction in online learning. *International Journal of Educational Technology in Higher Education* 18, 1 (Oct. 2021), 54. https://doi.org/10.1186/s41239-021-00292-9

[31] Donggil Song, Eun Young Oh, and Marilyn Rice. 2017. Interacting with a conversational agent system for educational purposes in online courses. In *2017 10th International Conference on Human System Interactions (HSI).* 78–82. https://doi.org/10.1109/HSI.2017.8005002

[32] Jiao Sun, Q. Vera Liao, Michael Muller, Mayank Agarwal, Stephanie Houde, Kartik Talamadupula, and Justin D. Weisz. 2022. Investigating Explainability of Generative AI for Code through Scenario-based Design. In *27th International Conference on Intelligent User Interfaces (IUI '22).* Association for Computing Machinery, New York, NY, USA, 212–228. https://doi.org/10.1145/3490099.3511119

[33] Kamilla Tenório, Viktoriya Olari, Margarita Chikobava, and Ralf Romeike. 2023. Artificial Intelligence Literacy Research Field: A Bibliometric Analysis from 1989 to 2021. In *Proceedings of the 54th ACM Technical Symposium on Computer Science Education V. 1 (SIGCSE 2023).* Association for Computing Machinery, New York, NY, USA, 1083–1089. https://doi.org/10.1145/3545945.3569874

[34] Kyle Wiggers. 2024. OpenAI signs up its first higher education customer, Arizona State. https://techcrunch.com/2024/01/19/openai-signs-up-its-first-higher-education-customer-arizona-state/

[35] Michael V. Yudelson, Kenneth R. Koedinger, and Geoffrey J. Gordon. 2013. Individualized Bayesian Knowledge Tracing Models. In *Artificial Intelligence in Education.* Vol. 7926. Springer Berlin Heidelberg, Berlin, Heidelberg, 171–180. http://link.springer.com/10.1007/978-3-642-39112-5_18 Series Title: Lecture Notes in Computer Science.

[36] Xuesong Zhai, Xiaoyan Chu, Ching Sing Chai, Morris Siu Yung Jong, Andreja Istenic, Michael Spector, Jia-Bao Liu, Jing Yuan, and Yan Li. 2021. A Review of Artificial Intelligence (AI) in Education from 2010 to 2020. *Complexity* 2021 (April 2021), e8812542. https://doi.org/10.1155/2021/8812542 Publisher: Hindawi.