Gaining Insight into Effective Teaching of AI Problem-Solving Through CSEDM: A Case Study

Spencer Yoder
North Carolina State University
smyoder@ncsu.edu

Cansu Tatar
North Carolina State University
ctatar@ncsu.edu

Ifeoluwa Aderemi
North Carolina State University
iwaderem@ncsu.edu

Sankalp Boorugu
North Carolina State University
sboorug@ncsu.edu

Shiyan Jiang
North Carolina State University
sjiang24@ncsu.edu

Bita Akram
North Carolina State University
bakram@ncsu.edu

ABSTRACT

CS education for all has been established as an important area of focus for both researchers and practitioners in recent years. At the same time, due to the increasing prevalence of AI technology in humans' everyday lives, artificial intelligence (AI) education for K-12 is gaining special attention among CS educators. AI literacy, even more than general CS competencies, requires evidence-based research to be effectively integrated in our schools. The common learning environments utilized for CS education enable us to go beyond conventional educational research approaches by providing a platform where detailed data can be collected from students' interaction with CS-education-related activities. Thus, conventional educational research approaches coupled with insights gained from pattern recognition and student modeling approaches enable us to effectively improve our instruction and to provide students with adaptive scaffolding. In this work, we present our first AI curriculum module that is designed to teach a fundamental AI search algorithm, Breadth-First Search (BFS), through a series of progressively scaffolded activities. Data is collected from a preliminary pilot of this activity with a high-school student in the form of a think-aloud protocol, screen capture, submitted block-based programming artifacts, and interview questions. Our results demonstrate that our activities have been successful in increasing the student 's knowledge about the BFS algorithm and more importantly, how this particular AI algorithm can be utilized to solve real world problems. Based on the results of this pilot study, we propose designing a comprehensive AI curriculum contextualized within a learning environment that collects detailed data from students' progress to inform instructional design and facilitate adaptive scaffolding for students.

Keywords

K-12 AI Education, Breadth-first-search AI algorithm CSEDM for AI Education

1. INTRODUCTION

It has been almost half a decade since Simon Papert introduced the idea of computer science (CS) for all [14]. Since then, many researchers[15], educational institutes (e.g. [13]) and organizations [18] have picked up the effort to create accessible CS curricula for K-12 classrooms. Meanwhile, artificial intelligence, a sub-field of computer science, is becoming a domineering aspect of people's everyday lives [17]. Daily technologies and decisions are becoming more and more dependent on AI. Under these circumstances, it is imperative for our new generation to gain a fundamental understanding of AI mechanisms and also its potential to introduce biases and unfairness through automated decision-making. Furthermore, the advances in AI systems are substituting many of the old jobs with ones that require the ability to do problem-solving with AI. For the aforementioned reasons, efforts for integrating AI into K-12 curriculum as an important part of CS education is gaining momentum [11], [7].

Integrating AI education into the K-12 curriculum poses significant challenges for educators since teachers often lack prior education in CS related fields [11]. Furthermore, the skills, knowledge, and abilities required for getting engaged in AI-related activities are novel to students and do not align with their conventional ways of learning [cite]. Thus, effective instructional approaches should be identified through evidence-based research. This is especially important since strong stereotypes and biases about who can learn this field prevent some of the students from getting engaged in the devised curriculum in the first place [12]. This can be facilitated by taking advantage of one of the main affordances of CS learning environments that is their inherent capability to collect fine-grained data from students' interactions with the learning activities. The collected log data can then be utilized in data-driven analysis that can inform instructors about the curriculum effectiveness and students' learning.

In this work we present the preliminary design of an AI curricular module that teaches one of the most fundamental AI search algorithms, breadth-first search (BFS). The module

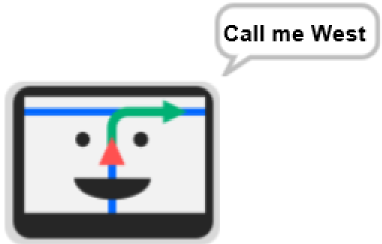

Figure 1: West, the character to guide students through the exercises.

consists of a series of activities that aims to teach learners about the BFS algorithm and ends with a coding activity where students have to incorporate the BFS algorithm in a real-world problem scenario to build a contact tracing application for COVID-19. While many devised AI curricula focus on the machine learning (ML) aspect of AI, we aim to provide K-12 students with a comprehensive view of AI that includes ML as an integrated aspect of AI. We present a case study [1] where this curricular module is piloted with a high-school student and data is collected through think aloud protocol [3], screen recording, and clinical interviews [8]. This data is then analyzed to make inferences about the effectiveness of the curriculum, required improvements and implications for design for a comprehensive AI curriculum integrated within a learning environment that collects learners' interaction data to provide both educators and students with adaptive assessment and feedback.

Section 2 discusses related work. The details of the devised curriculum are presented in section 3. The case study is presented in section 4, and the results of the pilot study are discussed in section 5. Section 6 concludes the work, and section 7 discusses implications for design and future work.

2. RELATED WORKS

Emerging research is exploring the design of learning experiences to foster youths' Artificial Intelligence (AI) literacy so that they are prepared to enter and engage with an AI-filled future [10]. In particular, efforts have been made to engage youth in understanding AI through the development of machine learning (ML) models. For instance, a work by Zimmermann-Niefield and colleagues (2019) provided an embodied learning experience for youth to create ML models for recognizing their own physical activities [19]. They found that youth developed an understanding of how ML models learned patterns of body movements and this could contribute to the understanding of the iterative process of ML. In addition, Google developed web-based tools (e.g., Teachable Machine) to make ML accessible to the public, including youth [3]. These studies stressed the cultivation of data literacy among youth as modeling data as a core concept in ML. While K-12 AI education is gaining momentum, the curricula being devised for K-12 students group tend to focus on basic ML approaches. However, these curricula fail to demonstrate the connection between ML techniques and the broader schema of automated problem-solving [5, 9, 6]. As AI technology is being more thoroughly integrated into our everyday lives, it is imperative for students to become familiar with a holistic view of problem solving with AI.

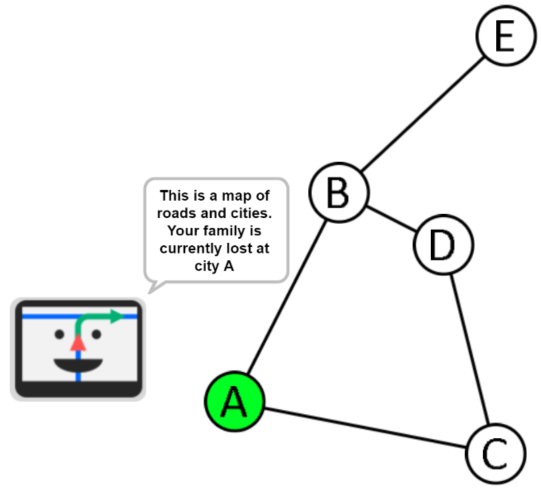

Figure 2: The graph representation of a city map.

In this paper, we present a preliminary study that investigates how students learn in a series of carefully-designed activities that introduce them to automated problem-solving approaches and their ability to leverage AI techniques to improve their decision-making process.

3. ACTIVITY DESIGN

We developed 5 exercises in UC Berkeley's Snap block-based programming language that introduce and use a simple graphical interface for teaching the breadth-first search algorithm (BFS). The first four exercises serve as an introduction to BFS, and feature a character, West, pictured in Figure 1, who guides student through the steps of the algorithm by telling them how to act as a GPS. The last activity is formatted as a programming exercise, where students are given an incomplete implementation of BFS, and are required to fill in the gaps. The algorithm is used for contact tracing, similar to the COVID contact tracing apps, to showcase a real-world application of BFS.

3.1 Exercise 1

Exercise 1 is designed to introduce graph search as a problem in the context of a GPS navigating between cities, as well as to show how the methodologies for performing this search differ between computers and humans. West shows the student a city map as a graph, with nodes representing cities and transitions representing roads between them (Figure 2). The student is then instructed to click through the cities to find the shortest path from a starting city to a goal. The student does this first with the simple map shown in figure 2, then with a more complicated map, and then finally with a map which has cities hidden from the student, which can only be shown once they click an adjacent city. This last example is meant to describe the perspective of the algorithm to the students, to contrast it with the human perspective of seeing the whole state space at once.

3.2 Exercise 2

Exercise 2 is designed to introduce BFS as an algorithm for performing graph search. BFS is introduced to the students via a series of animations and they are instructed to perform

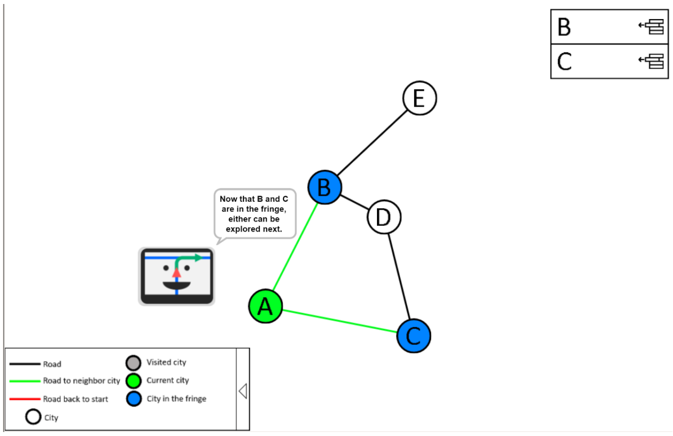

Figure 3: The graphical interface students interact with to perform BFS. (Bottom-left) The legend, which explains to students how the different states for nodes and transitions are displayed. (Middle) The graph representing a city map. (Right) The fringe.

each of the steps of BFS by interacting with our graphical interface, shown in Figure 3. This interface provides two ways to perform the actions of the algorithm. The first is to add nodes to the fringe by clicking them on the displayed graph. The second is to remove nodes from the fringe. The legend in Figure 3, which shows the graphical representations of each state for the nodes and transitions, is displayed and the interface updates accordingly as students interact with it.

3.3 Exercise 3

Exercise 3 consists of a completely guided run-through of BFS for a medium-sized graph with eight nodes and ten transitions. At every step of the algorithm, students are told what to do next and upon making a mistake, they are informed about what the mistake is and re-prompted until choosing the correct option.

3.4 Exercise 4

Exercise 4 is composed of a partially guided run-through of BFS. To connect this exercise with a real-world application, we show the graph superimposed over a map of the east coast of the U.S., with the nodes named for their corresponding cities. For the duration of the exercise, students complete the steps of the algorithm without feedback unless they make a mistake. Upon making a mistake, they are informed about the nature of their mistake and re-prompted until they perform the correct next step.

3.5 Exercise 5

Exercise 5 challenges students to implement their understanding of BFS as Snap code. For this exercise, we recontextualized the state space for BFS from a map of cities to a social network of people. When the exercise is initialized, one to three random people are marked as infected with COVID-19. The objective for BFS was changed from finding the shortest path from a starting city to a goal city to finding the shortest number of social connections from a user-selected person to an infected person. This scenario was inspired by the COVID contact tracing apps that have recently been developed and implemented in order to show students a real-world application of BFS. Figure 4, right shows the graph of people with the infected people in red.

Students were given a partial implementation of the algorithm along with missing blocks, with some instructions on how to snap the blocks into their proper place (Figure 4, left). Upon running the algorithm, if it is correctly implemented, students see the path from the user-selected person to the nearest infected person.

4. PRELIMINARY CASE STUDY

4.1 Research Design

This study employed the case study method [16]. This method has been seen as a framework to determine the problems that need to be studied in a bounded system [16]. Furthermore, Creswell and Poth [4] elaborate that the case study approach is a research methodology that aims to identify specific cases and to explore in-depth content related to these cases. Since this study aims to understand problems, challenges, and opportunities while piloting the AI curriculum module within a bounded system, this methodology was followed for the research design.

4.2 Participant and Context

The case study participant was Selim (pseudonym). He was a student in the 11th grade at a public high school in the United States. He had an interest in Artificial Intelligence and was actively looking for opportunities to learn about how AI worked and how AI could be used in the field of his interest, Biomedical Engineering. He had some general knowledge about AI and Python programming. The study took place in the context of two informal virtual sessions that were designed to introduce Artificial Intelligence with the BFS search algorithm. The sessions were conducted remotely, a week apart, via Zoom, a video conferencing tool. The first session took approximately one hour. In this session, we first described the goal of the session and emphasized that the student was expected to experience the first four learning activities as a user, complete a post-activity questionnaire, and participate in an interview. The second session was mainly focused on the fifth activity and took 1.5 hours. In this session, Selim answered questions related to the previous session, followed a written Snap tutorial, completed the activity, and participated in the interview.

4.3 Data Collection Process

In this study, we collected data in the form of a think-aloud protocol with approximately thirty minutes of a semi-structured interview, post-questionnaire, screen capture, and submitted block-based programming artifacts. In the first session, we asked the participant to answer warm-up questions (e.g., Have you ever taken any class on AI?) and complete an activity that aimed to extract his interests and general knowledge about AI. In addition to the warm-up activity, at the end of each learning activity, we also asked “What do you think you have learned? What else would you like to explore?”. With these questions, the participant reflected upon his own learning. Also, after he completed the activities, we asked him to complete the post-questionnaire that included seven multiple-choice questions related to the

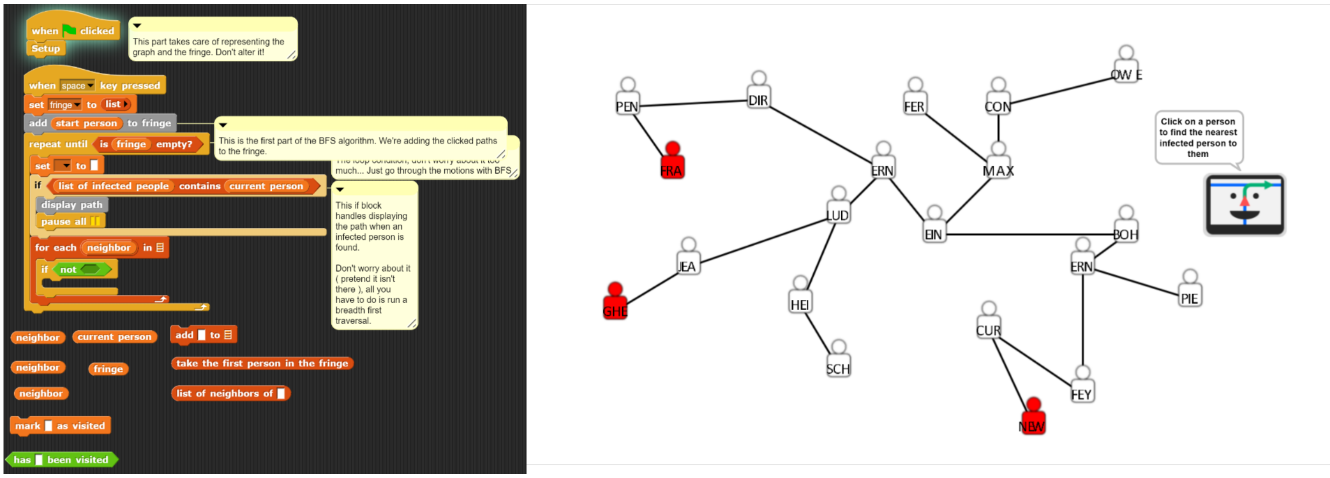

Figure 4: Exercise 5. (Left) The incomplete BFS implementation with the missing blocks located under the code. (Right) the graph representing the social network from which the student selects an arbitrary person to trace to the nearest infected person.

BFS learning activities. At the end of the first session, we conducted semi-structured interviews for approximately 10 minutes to learn more about his experience. In the second session, we followed a similar structure. We started the session with the warm-up questions to recall the student's understanding from the previous session and to correct misconceptions. Before the activity, we guided the student through the creation of pseudocode which reflected his understanding of the algorithm. We then provided a written Snap tutorial to allow the student to get familiar with the learning environment. After he skimmed through the tutorial document, he moved on to the fifth activity. At the end of the activity, we conducted an interview to learn more about his experiences.

4.4 Data Analysis

To analyze the transcripts of interviews, we followed the thematic coding method [2], which is a widely used method to analyze qualitative data for identifying, organizing, and describing themes. In the first round of the analysis, one researcher analyzed the data to explore learning challenges and opportunities. After the first-round coding, a peer-debriefing meeting was held with three other researchers to ensure the trustworthiness of the findings. Additionally, we analyzed the student's process of building the block-based programming artifacts by following the thematic coding strategies. We open-coded the students' actions in the activities to identify learning challenges and opportunities. After that, we discussed these codes and came up with the general themes discussed in section 5.3.

5. RESULTS AND DISCUSSION

Results are organized around three overarching themes: teaching AI as an interdisciplinary field, creating opportunities for students to discuss who benefits from AI and is left out, and promoting effective AI learning through developing a humanized curriculum.

5.1 Teaching AI as an interdisciplinary field

We should provide opportunities for students to understand the integration of AI in application areas and teach AI as an interdisciplinary field. Selim was interested in the field of biomedical engineering and believed that AI would contribute to innovations in every field, including biomedical engineering. Thus, he was motivated to learn more about the application of AI in the field of his interest. During the first session, he shared, "If you are trying to determine, for tissue engineering, maybe you could use a model to determine what's the best material to use based on different situations." At the end of the first session, Selim asked the research team about learning experiences in biomedical engineering and computer science in higher education since he was at the critical stage of choosing future careers. His question highlighted the need of supporting students to develop interdisciplinary learning and collaboration skills and understand the integration of AI in fields of their interest.

5.2 Creating opportunities for students to discuss who benefits from AI and is left out

Our analysis indicates that topics related to AI ethics might serve as a catalyst for students to have in-depth discussions about who benefits from AI and who is left out. For instance, in the excerpt below (Excerpt 1), guided by the research team, Selim emphasized that AI ethics could be an engaging topic of discussion. Excerpt 1.

1. **Research team**: Have you talked about AI with your family and friends?

2. **Selim**: Not really my friends. With family, yeah. We've had a few discussions. Um, it's probably a really common discussion. I think one of my parents heard it on a podcast or something like the thing about a self-driving car having an unavoidable accident.

3. **Research team**: Yeah, I remember this one.

4. **Selim**: Like, who it (referring to the self-driving car) picked to crash into or something. Just like ethics of AI is probably the most interesting conversation to have with other people.

In Turn 4, he highlighted that a model would fall short in addressing the moral question of who a self-driving car should kill in an unavoidable accident. However, he did not recognize that essentially, it's the car maker or designer who would make the decision.

1. **Research team**:Do you find AI ethics interesting?
2. **Selim**: Oh, yeah.
3. **Research team**: In what ways? Can you give an example?
4. **Selim**: So another thing that I found interesting is the hiring process. I think it's Amazon. It has implicit bias against hiring applicants that had women in their application. That's obviously an important issue that needs to be fixed. So there are really complex issues when you talk about AI ethics.

In turn 8, he raised the concern about gender bias using AI recruiting or resume screening tools. Overall, we can see that AI ethics and AI decision making that would have an impact on who benefits and is left is a promising topic for students to think critically about the impacts of AI technologies.

5.3 Promoting effective AI learning through developing a humanized curriculum

As described in section 3, the student is presented with a series of five activities. The first four activities are designed to introduce the BFS algorithm to the student while the fifth requires him to implement the BFS algorithm in a block-based programming environment in the context of a contact tracing app. Selim progressed though the first four activities rather smoothly. After each activity he indicated a deeper understanding of the BFS algorithm. He then successfully answered content knowledge questions presented to him at the end of the first session

In the beginning of the second session, the student started by reflecting on what he recalled from the previous session and was then asked to devise pseudocode for the BFS algorithm. Through this process he demonstrated a good understanding of the BFS algorithm but a lack of familiarity with writing pseudocode. With the guidance of researchers he managed to turn his knowledge of the BFS algorithm into a pseudocode format. He was then presented with an incomplete contact tracing app that utilizes BFS to find the shortest path between an arbitrary person and an infected person in his social circle.The BFS algorithm was partially implemented as a Parson's problem, and the student snapped the readily available blocks on the screen to complete the algorithm. Here again, the student showed an adequate understanding of the BFS algorithm. However, he faced some challenges when trying to implement it within the block-based programming environment. We identified three main themes of challenges faced by the student while conducting the programming. First, he was not familiar with some of the blocks, list-related blocks in particular, and was confused about what they represent in the incomplete program. Secondly, he had trouble understanding the functionality of some of the custom blocks provided for him. Finally, the student was unclear about how to map from steps of his pseudocode onto the different sections of the partial code.

While the first challenge can be addressed by providing students with a more extensive block-based programming tutorial beforehand, the second two challenges require improvements in the design of the activity. For example, a better description of the functionality of custom blocks and a clearer distinction between custom blocks and contextually named variables can alleviate some of the confusions that students might encounter when analyzing the partial code presented to them. Additionally, having students implement a simple case of BFS following the same context as the previous activities might facilitate the transition to implementing the BFS algorithm in an arbitrary real-world context.

6. CONCLUSION

With the advances of AI technology and its rapid integration within our society, it is imperative that AI education is included as part of the K-12 standard curriculum. In this study, we presented the preliminary design of a curricular module that is designed to teach a foundational AI algorithm, breadth-first search (BFS), to high school students. We presented the results of a pilot implementation of this curriculum with a high-school student as a case study. Our results demonstrated that this intervention helped the student gain a better perception about the interdisciplinary nature of AI and how it can be incorporated to enrich other fields. Furthermore, the discussions fostered curiosity around AI ethics which is an important aspect of AI education. Finally, our analysis of the student's interactions with the designed activities and his reflections showed that the activities were successful in improving the student's understanding of the BFS algorithm while posing some challenges when the student tried to map from the algorithm to a block-based programming implementation.

7. FUTURE WORK

In the future, we aim to utilize insights obtained from this case study to not only improve the design of the current module but also to inform design of the future AI-related curricular modules. While this study provided a qualitative analysis of observed interactions between the student and activities, we plan to automate this process in the future through data-driven approaches to obtain more insights into the effectiveness of our curriculum and students' learning. We propose creating a series of curricular modules that teach students a holistic view of AI and equip them with knowledge, skills, and abilities that prepare them for conducting problem-solving with AI in humanized real-world scenarios. We further propose situating our curriculum within a learning environment where data can be collected and analyzed for informing instruction and providing students with adaptive scaffolding.

8. REFERENCES

[1] M. Bassey. Case study research. *Research methods in educational leadership and management*, pages 108–121, 2002.

[2] V. Braun and V. Clarke. Using thematic analysis in psychology. *Qualitative research in psychology*, 3(2):77–101, 2006.

[3] E. Charters. The use of think-aloud methods in qualitative research an introduction to think-aloud methods. *Brock Education: A Journal of Educational Research and Practice*, 12(2), 2003.

[4] J. W. Creswell and C. N. Poth. Qualitative inquiry and research method: Choosing among five approaches, 2007.

[5] J. Estevez, G. Garate, and M. Graña. Gentle introduction to artificial intelligence for high-school students using scratch. *IEEE Access*, 7:179027–179036, 2019.

[6] T. Hitron, Y. Orlev, I. Wald, A. Shamir, H. Erel, and O. Zuckerman. Can children understand machine learning concepts? the effect of uncovering black boxes. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, pages 1–11, 2019.

[7] S. Isotani, E. Millán, A. Ogan, P. Hastings, B. McLaren, and R. Luckin. *Artificial Intelligence in Education*. Springer, 2019.

[8] V. R. Lee and D. A. Fields. A clinical interview for assessing student learning in a university-level craft technology course. 2013.

[9] P. Lin, J. Van Brummelen, G. Lukin, R. Williams, and C. Breazeal. Zhorai: Designing a conversational agent for children to explore machine learning concepts. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 13381–13388, 2020.

[10] D. Long and B. Magerko. What is ai literacy? competencies and design considerations. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, pages 1–16, 2020.

[11] P. S. Mohammed et al. Towards inclusive education in the age of artificial intelligence: Perspectives, challenges, and opportunities. In *Artificial intelligence and inclusive education*, pages 17–37. Springer, 2019.

[12] D. J. Moran. The need for evidence-based educational methods. In *Evidence-based educational methods*, pages 3–7. Elsevier, 2004.

[13] S. Nikiforos, C. Kontomaris, and K. Chorianopoulos. Mit scratch: A powerful tool for improving teaching of programming. *Conferrence on Informatics in Education*, pages 1–5, 2013.

[14] S. Papert. *The children's machine: Rethinking school in the age of the computer*. ERIC, 1993.

[15] D. Passey. Computer science (cs) in the compulsory education curriculum: Implications for future research. *Education and Information Technologies*, 22(2):421–443, 2017.

[16] R. E. Stake. Qualitative case studies. 2005.

[17] C. Tatar, M. Yoder, M. Coven, K. Wiedemann, J. Chao, W. Finzer, S. Jiang, and C. Rosé. Modeling unstructured data: Teachers as learners and designers of technology-enhanced artificial intelligence curriculum. In *International Society of the Learning Sciences Annual Meeting 2021*, 2021.

[18] A. Tucker, F. Deek, J. Jones, D. McCowan, C. Stephenson, and A. Verno. *A Model Curriculum for K–12 Computer Science*. ACM/Association for Computing Machinery, 2003.

[19] A. Zimmermann-Niefield, M. Turner, B. Murphy, S. K. Kane, and R. B. Shapiro. Youth learning machine learning through building models of athletic moves. In *Proceedings of the 18th ACM International Conference on Interaction Design and Children*, pages 121–132, 2019.