Creative AI Literacies for Families

Stefania Druga

A dissertation
submitted in partial fulfillment of the
requirements for the degree of

Doctor of Philosophy

University of Washington

2023

Reading Committee:

Amy J. Ko, Chair

Katie Davis

Ben Shapiro

Program Authorized to Offer Degree:
Information School

ACKNOWLEDGMENTS

Thank you to all the families who participated in my studies, my mentees, collaborators, and friends who provided feedback and support. Also very grateful to my family and my partner for being there for me, especially during these challenging times. Lastly, I want to thank my committee and advisor for their guidance and patience.

University of Washington

Abstract

Creative AI Literacies for Families

Stefania Druga

Chair of the Supervisory Committee:
Professor Amy J. Ko
Information School

Many families engage daily with artificial intelligence (AI) applications, from conversations with a voice assistant to mobile navigation searches. Unfortunately, existing intelligent technologies in the home are prone to algorithmic bias and cyber-security attacks. To ensure the new generations of children growing up with AI can develop a critical understanding of AI technologies, we must explore parents' roles in helping their children develop AI literacies and identify how best to support families in engaging in creative learning activities with and about AI while proposing recommendations for future family-centered AI literacies resources. To guarantee that diverse families can realize their dignity and potential to develop AI literacies, we must enable stakeholders (e.g., children, parents, technology designers) to make informed, timely, and equitable action.

While AI technologies often perpetuate and exacerbate inequities in many contexts, they could also support family learning goals if properly contextualized for use by stakeholders. This work explores this idea in informing youth about how to train and program smart games in self-directed learning experiences and informing curriculum & technology designers' domain expertise with empirical evidence on family AI literacies practices. I investigate how to design novel programming and AI learning interfaces for families to develop literacies for creating and being creative with AI. This involves the development of Cognimates, a family AI programming tool.

This dissertation demonstrates the following thesis: Family joint engagement in creative AI literacy activities enables children to: (1) discover the core concepts of AI technologies and the power they can bring, (2) foster critical reflection on the uses of AI in the home and beyond, and (3) learn creative coding with AI as a way to enable self-expression.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1

INTRODUCTION

Figure 1.1: Image generated with DALL·E model using the prompt "abstract art of parents and kids learning with creative AI"

AI devices entered the lives of young people in 2013 with the launch of voice assistants, which became part of their homes, giving birth to the first generation to grow up with AI. Nevertheless, the convenient smart devices became Trojan horses for a new set of paradigms and dilemmas that educators and parents must resolve. In this context, there is an urgent need and opportunity to prepare families to make meaningful use of AI technologies in the home.

Before we move further, let's define what I mean by *Artificial Intelligence*. In the context of this dissertation, I define AI as any computer program that teaches computers human abilities like walking, talking, thinking, and listening. In recent years, AI has made many advances in replicating specific human abilities, such as speech or vision, primarily using large data sets created through manual human classification and using that data to build machine-learned prediction models. However, it is essential to note that these prediction models perpetuate and amplify the values and biases encoded in their algorithms and data. Therefore, educators, families, and communities must develop *AI literacy* to request and defend their rights to algorithmic justice to avoid the risk of discriminating against or oppressing minoritized groups. I define AI literacy as the ability to read, author and analyze with AI and in chapter 5 I dive deeper into each of these dimensions. In this context, I see a unique opportunity to prepare families to understand different AI concepts and become critical thinkers with and about AI.

Moreover, the increasing use of large language models and ChatGPT technology in applications has the potential to impact families and children significantly. These applications will offer AI-driven conversations and automated tasks that can support and help children learn and provide parents with ways to monitor their children's online activities. This technology could also provide more accurate and timely feedback to parents about their children's behavior. In addition, large language models and ChatGPT technology can be used to create personalized experiences for children, such as providing them with tailored advice or recommendations. However, it is crucial to consider the potential privacy and security implications of using this technology and the potential for misuse or exploitation.

In order to adequately address the implications and risks associated with AI and large language models such as ChatGPT, it is essential that families, educators, and parents gain the skills and knowledge to assess the benefits and risks of adopting AI-driven applications. Furthermore, it is necessary to create awareness and understanding of AI technology's ethical and legal implications and its use in our society. In order to ensure that families can make informed decisions, we must provide resources and education to help them comprehend and consider the potential risks and benefits of AI technology.

Additionally, it is essential to create a culture of dialogue and understanding around the ethical and legal implications of AI technology, its use in society, and the potential consequences of using AI without proper oversight. This dialogue should involve parents and educators, create trust, and build relationships between families, educators, and the technology industry.

Through collaboration between families, researchers, technology designers, and educators, we can empower families and children to make well-informed decisions about Artificial Intelligence (AI) by increasing their awareness of the associated risks and benefits. It is also necessary to create a culture of open dialogue and mutual understanding between the technology industry, families, and educators to ensure trust and comfort in using AI technology. To facilitate this, it is essential to provide families and educators with resources and education to make informed decisions about AI technology.

Several initiatives provide AI educational resources for youth [304, 98, 198]. However, few resources currently help parents mediate AI technologies, despite growing parental concerns about their children's in-home use of AI. Pediatricians, policymakers, and parents' associations struggle to provide family guidance for appropriate AI use, and their recommendations are influenced by the affordances and limitations of existing commercial AI products [282, 151, 325, 3]. Further, AI products such as voice assistants or smart mobile apps are only sometimes developed for youth despite increasing usage [151]. These products pose additional concerns in terms of (1) *inclusivity* for families of different ethnicities, family structures, general technological literacies, and diverse socioeconomic backgrounds [27] and (2) *algorithmic fairness*, or subtle ways AI technologies can amplify bias, sexism, racism, and other forms of discrimination [60, 30].

By understanding the potential implications of algorithmic bias, families can begin to think more critically about how AI technology is used in their homes and communities. It is crucial to examine how AI models are developed, trained, and evaluated to ensure that they are correctly calibrated and that potential biases are not perpetuated. Additionally, families should ensure that AI technologies are not used to discriminate against certain groups of people. Families should also be aware of the potential for AI technology to be used maliciously, such as for surveillance or exploitation purposes. It is essential to understand the implications of algorithmic bias and to take steps to ensure that AI technology is being used responsibly and ethically. Finally, families should seek out opportunities to engage with AI technology in a meaningful way, such as by participating in AI research projects or engaging in dialogue with AI experts.

Prior studies have described the benefits of families jointly learning about technology or engaging in technology co-design. For example, Barron et al. showed that parents could play various supporting roles, such as collaborator and learning broker [41]. More recent work by Michelson et al. emphasized the importance of balanced partnerships in family technology co-design activities [212], and Yu et al. showed that parents primarily act as spectators, scaffolders, and teachers when supporting children interact with coding kits [344]. Though these studies underline the importance of family engagement in children's technology learning, we need to be more open about best practices supporting joint family AI learning and co-design.

To understand joint AI learning, I explore how families can best develop multiple *AI literacies* in the home. Our work builds on the notion of multiple literacies [63], which emphasizes how negotiating multiple linguistic and cultural differences in our society is central to the lives of young people and their families. For our purposes, *AI literacies include the ability to read, work with, analyze and author with AI* [103, 106, 98]. Our framing of multiple AI literacies also borrows from Freire's assertion that literacy is about the acquisition of technical skills and the emancipation achieved through the literacy process [122].

For this research, I conducted a series of studies with families to understand the range of AI literacies in families. I asked participants to share their experiences of using AI and how they have developed an understanding of this technology. Additionally, I asked them to discuss the challenges they have faced in learning and using AI in their homes. This research revealed a range of AI literacies that families are developing and the challenges they face in learning and using AI. I discovered that families are finding ways to use AI in their lives, from using it to perform tasks such as playing music and shopping to more complex activities such as building robots and coding. I also found that families face challenges in learning and using AI, particularly in understanding and making sense of the technology. Finally, I identified a few key themes that emerged from our studies, such as the need for more accessible and inclusive approaches to learning AI and the importance of developing skills such as critical thinking and creativity when it comes to AI.

This dissertation explores how to design learning experiences that enable stakeholders (family members, technology & curriculum designers) to understand and use AI to support meaningful and creative family learning experiences. I explore this within three domains:

- **1. Curriculum Design:** Existing efforts in AI education for K12 fail to consider families and developmental considerations for youth.

- **2. Family AI Literacies:** How families co-design and jointly engage with AI learning activities and applications.

- **3. Creative Coding with AI:** How children and parents engage in collaborative creative coding supported by an AI friend.

To build on these research findings, this dissertation will focus on developing a better understanding of how families can develop AI literacies, how AI technologies can be used in the home, and how AI technologies can be used to support meaningful family learning experiences. First, I will examine the implications of AI technologies for family relationships and the home environment and explore how AI technologies can foster creative, meaningful, and equitable learning experiences for families. Additionally, I will investigate how we can support families to work together and create AI-enabled learning experiences that are inclusive and accessible to all and that promote ethical and responsible AI use. Finally, I will explore best practices for family AI co-design and joint AI learning activities, aiming to develop practical, meaningful, and equitable approaches.

This dissertation demonstrates the following thesis statement: *Family joint engagement in creative AI literacy activities enables children to: (1) discover the core concepts of AI technologies and the power they can bring, (2) foster critical reflection on the uses of AI in the home and beyond, and (3) learn creative coding with AI as a way to enable self-expression.*

Chapter 2
FAMILIES AND CREATIVE AI LITERACIES

Figure 2.1: Kids and parents playing with Cozmo Robot in our AI Literacy workshop in Berlin, Germany, 2019

This chapter presents relevant related work from the following categories: Parenting in a new Media & Technology Ecology, Families Interactions and Learning about AI, Bias, Power and Critical Understanding of AI, Role of Family Joint-Engagement in AI Literacies, Support Tools for Creative AI Literacies.

2.1 Parenting in a New Media & Technology Ecology

Home technology and media environments reflect families' values and aspirations and their beliefs about the impact of new media and technology on their children's learning and communication. With the arrival of the first home computers, the influence of computing and the media created with this new medium became intertwined with family life. This also

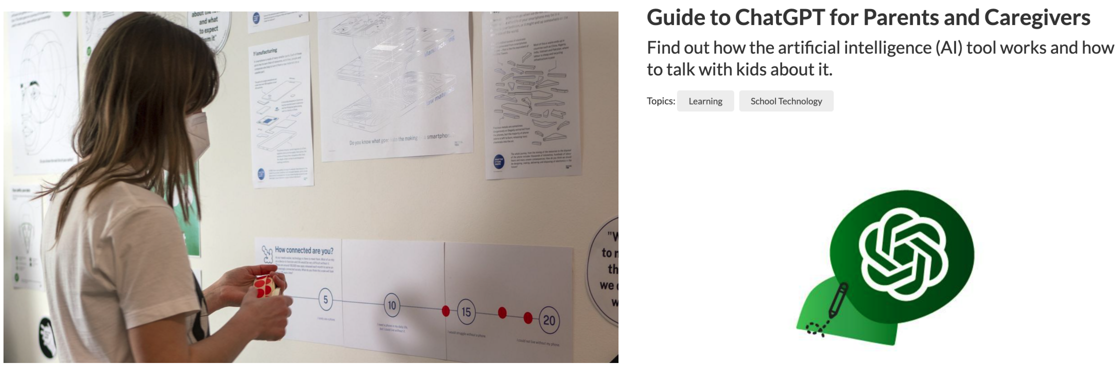

Figure 2.2: Examples of platforms for Media & Digital Literacy:(left) Tactical Tech Activity from Digital Literacy for Youth Initiative [299]; (right) Common Sense Guide on ChatGPT for Parents [108]

brought about parents' anxiety about integrating new media in the home [280]. This anxiety persists today as technology advances rapidly and parents either lack the knowledge to support their children or the information to make informed decisions (e.g., understanding what data is collected by devices and how it is used [210]). Our previous work demonstrated how parental attitudes and values shape children's perception of and attribution of intelligence to smart toys, and robots [101]. In this context, it is essential to understand families' position on technology adoption and to inform their decisions better.

With the rapid advancement of AI technologies and Large Language Models (LLMs) applications, companies are cutting corners on trust and safety efforts, negatively affecting young people's digital ecosystems. This can be due to social media platforms having fewer guardrails or AI chatbots needing to have filters for unsafe content. This lack of incentive to protect users leaves parents and educators responsible for guiding kids to develop mindful habits as digital citizens [318]. Organizations such as Common Sense Education [108] and Tactical Tech [299] are providing dedicated curricula and activities for media and digital literacy (see fig.2.2) to support parents and educators.

Despite the progress in digital literacy initiatives, there is still a need to focus specifically on youth in the AI ecosystem. It is essential to consider the cultural differences that may arise when working with international families [255]. Our previous research revealed that family perceptions and attitudes towards AI devices vary greatly between countries [98]. For instance, children and parents in Denmark and Germany are less trusting of smart toys and devices than in the USA. More recent studies on machine learning education in the African context confirms this [270].

A more recent series of international AI literacy workshops we organized in Berlin closed with a big round of discussion with both children and parents. The topics covered in depth included robots in everyday life, AI in the future, fantasies and concerns, and potential applications. Many children and parents expressed concerns that such devices could take the place of human contacts and friends, and they added that such developments would contradict the basic purposes of technology. All such concerns notwithstanding, the parents repeatedly emphasized that they felt schools were still giving too little attention to AI. In addition, some of the families expressed their desire to learn how to program and "teach" the robots or the computer in their native language [28] (see Fig. 2.1).

In this thesis, I investigate how discourses and parenting approaches influence parents' strategies to manage AI media use in the family. In the following section, I examine the implications of bias, power, and critical understanding of AI in relation to family life.

2.2 Bias, Power and Critical Understanding of AI

Unlike humans, machines acquire intelligence through algorithmic techniques inspired by domains such as statistics, mathematical optimization, and cognitive science and powered by computer processing power and a large amount of data [190]. AI systems have great potential to help children and families through improved online search quality, increased accessibility via advances in digital voice assistants, and AI-supported learning [136, 262, 260]. However, AI systems can also amplify bias, sexism, racism, and other forms of discrimination, particularly for those in marginalized communities [60, 30]. It is, therefore, essential to promote a critical understanding of AI among children and families in this context.

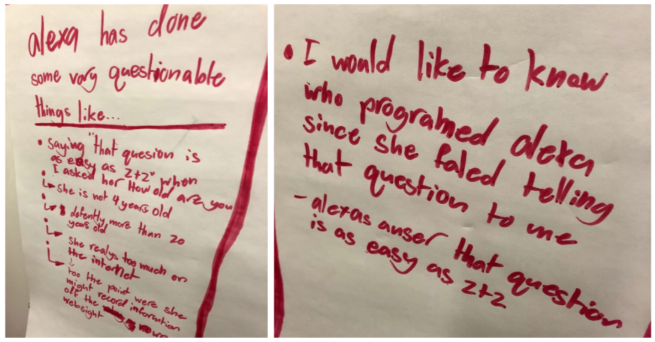

Figure 2.3: The poster made by B., an 11 years old girl, for describing her process of determining how old the voice assistant is in our study on algorithmic bias [103]

Without AI literacy, families, particularly those from historically marginalized groups, are at risk of being misled, scared, and missing out on potential learning opportunities in the future [113, 127, 229]. In addition, many AI devices have proven to be easy to compromise [313, 329, 26], and some companies designing voice assistants, for example, engage in questionable practices [3]. Families and children need to work together to learn about AI systems and to think critically about how this technology impacts their lives [98, 69, 200].

Previous studies on family engagement with digital technologies have highlighted the importance of considering the variations between families and parenting styles [294, 74]. Therefore, to promote algorithmic justice in families, we need to consider how various families can access these skills [339, 84].

Necessary technological infrastructure determines access to AI and AI literacy. For instance, a 2019 Pew study shows that in the USA, data caps and speed limit access to broadband [29]. AI systems often rely on large-scale technological infrastructures, so families without broadband may be left disengaged [253]. It is essential for minority groups to be able to both "read" and "write" AI. Smart technologies do much of their computing in the cloud, and without access to high-speed broadband, marginalized families may have difficulty understanding and accessing AI systems [39]. Families must be able to engage with

AI systems in their homes to develop a deeper understanding of AI. When designing AI education tools and resources, designers should consider how lack of access to stable broadband could lead to an AI literacy divide [314]. Research on families' comprehension of algorithmic equity has focused on children's views on AI and agent interactions. Reviews of the area [197] revealed several studies that suggest children often overestimate agent knowledge [102] and trust agents too easily [100, 146, 295, 333]. Skinner [285] observed that children linked kindness with fairness in AI agents, using polite correspondence with people to defend the fairness attribute. Coenraad [71] found that, without guidance, youth were aware of the visible bad effects of technology, not just AI, and could give examples of this prejudice in their lives. As educators and researchers intensify their efforts to teach children essential computing literacies [50, 65, 221], the studies presented in chapters 4 and 5 of this thesis provide a blueprint for how to leverage children's understanding and backgrounds to create a stronger moral understanding of the complexities of algorithmic fairness or lack thereof.

In our first study on the family AI Literacy framework, presented in Chapter 4, we showed that after watching videos of algorithmic prejudice examples, children could relate them to their own lives. They identified instances of unfair treatment from AI based on race, ethnicity, age, and gender [103](see fig.2.3). In chapter 5, I identify the most common roles parents play in helping their children develop a critical understanding of AI [92]. We also explore how the home can become a "third place" [142], where family members can engage in reflection activities that allow them to view AI literacies through the lens of culture and power [322]. This allows them to envision and imagine meaningful AI designs for the future.

2.3 Families Interactions and Learning about AI

Beneteau et al. demonstrated that parents play an essential role in helping their children communicate better with voice assistants [47] or recognize the assumptions these assistants make about children's questions [45]. In our earlier work, we revealed that parental models of machine intelligence also affect how children ascribe intelligence to machines [101] and that children and parents can collaborate in AI learning activities [103]. Recently, Long

Figure 2.4: Examples of family training a custom model with Teachable Machine in our study [103]

et al. demonstrated that parents and children could learn more about how AI works by co-designing interactive AI museum exhibits [197, 200].

Unequal access to AI technologies exacerbates digital divides, with only some children learning how to interact with smart toys and devices [43, 85]. In addition, our prior research has demonstrated that parental attitudes, socioeconomic status, and cultural differences significantly impact how children attribute agency, intelligence, and socio-emotional traits to smart devices [102, 89, 98].

Previous studies also show that youth can influence their parents' digital media use [72] and suggest the importance of parent and peer contexts for children's moral reasoning development [326]. As parents are still unfamiliar with some aspects of AI literacies, children can share their knowledge and perspectives [316, 192, 100, 315]. Nevertheless, parental guidance and scaffolding are still essential when considering the ethics of AI [238, 239] and algorithmic bias [103, 30].

Other studies have shown that children often misunderstand agents or overestimate their abilities. This may be due to a lack of understanding of how these agents work or because artifacts like toys and phones can talk, express emotions, and interact with youth in persuasive and charismatic ways [337, 115, 243].

In our study on machine intelligence perception, which we present in chapter 6, we show that children's perception of AI abilities changes after engaging in AI programming and training activities [94].

We recognize the need for inclusive AI literacies to prepare the next generation for life with AI. Our approach builds on the theory of "multiple literacies" [63]. This theory has been used to propose a transversal approach to computing education for youth [211], define critical literacies in a digital age [286], conceptualize digital games literacies for youth [33], and propose new computational literacies [319]. It has also been used to frame family literacy as a third space [142] between home and school [231], and to observe family environments that foster kids' curiosity [173]. Our approach to AI literacies involves four practices: multimodal and embodied situated practices, AI conceptual learning, critical framing of AI, and design for future meaningful use.

We view the family and home as a third space [142] where children can develop AI literacies [92]. Thus, we seek to investigate how to design family-centric learning activities that create zones of possibilities [220] (see an example of activity in fig.2.4). This combines family social contexts for learning, and their collective zone of proximal development [323].

We aim for our AI literacies interventions to teach new skills, such as training AI models, programming smart games, and testing smart devices, and new practices, such as situated reflection and re-design for meaningful family use. In the following sections, I will discuss the benefits of family-joint engagement when engaging in AI literacies learning activities.

2.4 Role of Family Joint-Engagement in AI Literacies

Stevens et al. conducted a review of research on *Joint-Media Engagement* (JME), which they define as "spontaneous and designed experiences of people using media together" [291]. They designed activities for children and parents to work together and engage with various forms of media. Their analysis focused on the six ideals of productive JME outlined in the paper: (1) mutual engagement, (2) dialogic inquiry, (3) co-creation, (4) boundary-crossing, (5) intention to develop, and (6) focus on content, not control [291]. This joint media

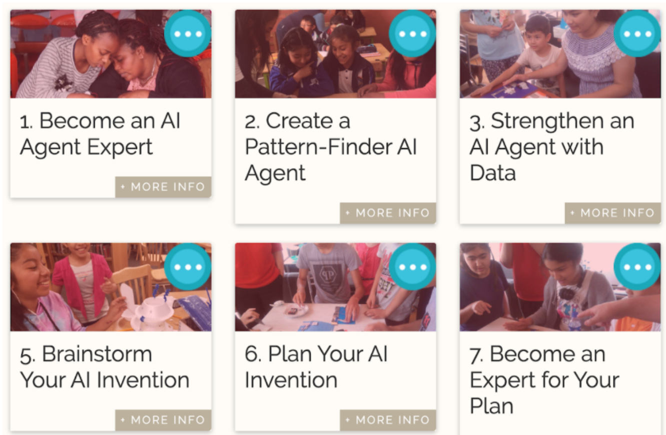

Figure 2.5: example of curriculum modules created by Technovation for the international Curiosity Machine Competition for families [300]

engagement framework serves as a guide for our analysis of family interactions in the studies presented in this thesis.

AI is a unique form of media that elicits assumptions and interactions different from traditional technological media forms, such as television. By engaging with it through the Joint Media Engagement (JME) framework, we can explore how it intersects with established JME parent-child dynamics and where it differs from or extends them.

Building on the third research case study presented by Stevens et al. [291], this thesis researches "ways that parents can be supported to engage in joint media engagement-creation (JME-C), even when they do not have expertise." It also carries out "micro-interactional studies to better theorize cognitive, relational, and affective components of JME-C." The JME-C framework is of particular interest, as exploring AI literacies applications in families is challenging due to the unfamiliar mechanisms and opportunities of AI to most people outside computer science.

For numerous reasons, the inter-generational structure of families is critical to understanding AI Literacies for youth. Prior work has found that parents, peers, and caregivers can play a dynamic role in youth learning. They can act as facilitators or guides [41], learners, or lead youth to see themselves as experts [213, 91]. Families can also bridge formal learning at school, and informal student-driven learning outside of school [219]. Other studies have demonstrated that parental experience in technology fields significantly impacts how they support their children's learning [85].

Family-oriented programs, such as Family Creative Learning (FCL) [257, 256], are important for families lacking "preparatory privilege" [207] to get involved in their children's creative coding activities. In addition, studies on AI perception have found that programmability is a key factor influencing youth understanding of the technology.

Research on family use and perception of coding revealed that parents' main concern is their limited programming knowledge [344]. Designers have explored text-free programming platforms to support parents better, finding that families can create together successfully [36, 133]. Further understanding AI programming in family contexts may uncover new opportunities to link youth interests in AI with interest-driven programming [75], family relationships [223], and formal computing education [38].

Our qualitative findings in chapter 6 offer new interpretations of prior research on program understanding. Previous work has mainly focused on individual cognitive accounts (e.g., [174, 17]). However, our investigation from a constructionist [233] and social sense-making [81] perspective reveals that children employ more than just cognitive strategies to understand AI behavior. These assets include social strategies such as observing and discussing with peers and introspective, egocentric strategies for inferring models of agent behavior.

Studies have shown that parental involvement in learning at home has a significant impact on school performance [40, 49], and is critical for children's future success. For example, the AI Family Challenge (AIFC) was a 15-week program implemented with 3rd-8th grade students (n = 7,500) and their families in under-resourced communities across 13 countries (see curriculum example in fig.2.5). The program aimed to teach families how to develop AI-based prototypes to solve problems in their communities. Pre- and post-surveys, as well as interviews with participants in the US, Bolivia, and Cameroon [69], was conducted. After the AIFC, 92% of parents reported that their child was more capable of explaining AI to others, and 89% believed their child could create an AI application. The findings suggest the need to improve parent training materials, connect technical mentors to local sites, and enhance the curriculum to make it more hands-on, engaging, and better illustrative of machine learning concepts. A recent study on family mediation of preschool children's digital media practices at home revealed that family members are often unaware of how much they are aiding children in developing competencies concerning media texts and devices [273]. To address this, we aim to involve parents and children in joint AI literacies activities. This will help clarify family members' roles in supporting each other. Building on our findings from workshops with groups of children engaging in collaborative sense-making of AI games on different platforms such as Cognimates(chapter 6), we studied how kids and parents engage in joint programming of rule-based games using the TileCode platform[90], or procedural smart games on Cognimates with the support of an AI assistant (chapter 7). Our studies on families developing AI literacies together at home, both with and without assistance, aim to identify the language and scaffolding strategies that parents use to explain AI and programming concepts to their children. This offers an opportunity to identify potential future interventions that address this family's joint engagement with AI literacies. In the last section, I explore ways to support family AI literacy and how future AI assistants for creative coding should be co-designed by children, parents, and researchers. This will ensure that everyone has an equal opportunity to express their creativity.

2.5 Supports for Family Creative AI Literacies

Creativity is driven by imagination and play for youth and is both practical and conceptual. What young people create today heavily depends on the tools and materials they have access to and what they make with them. From Froebel gifts, [5] to LEGO Mindstorms [8] and creative learning tools such as Scratch [206], notable efforts have been made to foster creative learning and coding for youth as a counter-culture to the instructionist approach to education that has been dominant since the industrial revolution. These initiatives flourished primarily outside of traditional educational institutions, leveraging two critical aspects of creativity for

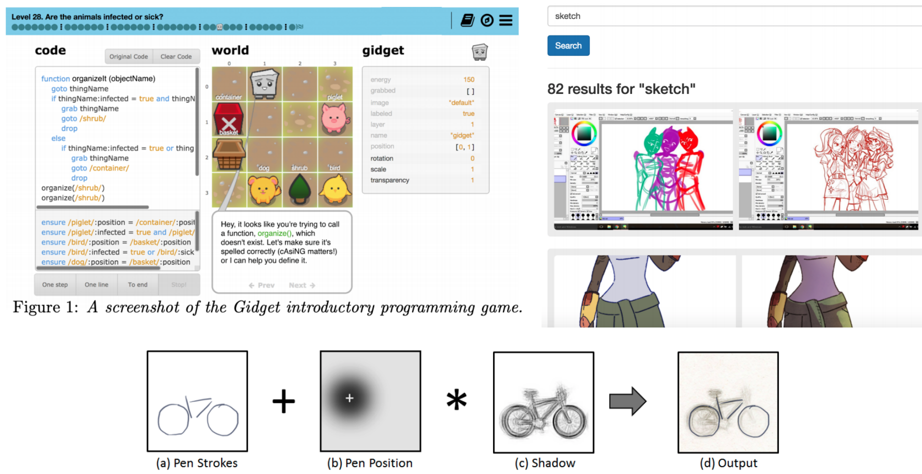

Figure 2.6: Creativity Support Tools Examples: Gidget for programming [186] (left), Mosaic for design [170] (right) and ShadowDraw for drawing [188] (bottom)

children: allowing them to tinker, construct, debug, test, and modify ideas and encouraging them to collaborate in person or digital communities.

The success of these projects has also driven change in the way creative thinking and coding are taught in schools, with more initiatives focused on project-based learning and coding. Nevertheless, questions remain as to how best to balance structure and agency in programming for youth [54].

As youth grow up with AI, our framing, understanding, and intelligence development are again under discussion, including the approach to creative thinking and coding. Recognizing that every child is born with immense natural talents [126] and innate creative potential [324], how can we design new learning opportunities and tools for creative thinking that allow families to flourish in an era of constant technology consumption?

Although a growing body of work suggests that technology-enabled tools could effectively scaffold parent-child activities, most have focused on supporting remote parent-child communication. For example, numerous projects have analyzed how technology-enabled systems can provide a virtual space for parents and children to interact [156, 293, 340]. Other studies have explored how to support remote parent-child activities, such as facilitating gameplay [116, 150] or reading together [246]. Recent work on parent-child interactions in co-located contexts has studied multi-touch tabletop applications [336], sensor-based exergames [268], and technology-enhanced storytelling activities [302, 70]. Although this work informs design, it speaks to something other than learning and AI literacies.

A recent systematic literature review study sheds light on the diversity of approaches to designing Creative Support Technologies (CSTs) [124] (see examples in fig.2.6). The study classification of CSTs found six significant categories of support in the creative process: pre-ideation, idea generation, evaluation, implementation, iteration, and reflection (see Table 9.1). As pointed out by Frich et al., much of CSTs are disconnected from the creators' daily practices [124]. In the context of computing education, youth want their programming learning to be authentic [275]. Authenticity in creative coding could involve providing the proper media support, like in the case of the danceON project [240], and the opportunity to work on microworlds with curated programming activities, such as fashion or music [307].

In the field of Human-Robot Interaction (HRI), several studies have shown that having embodied agents, such as robots or connected toys, could lead to increased engagement and learning effects for youth [235, 23]. In digital sketching [165] and game design [202], Karimi and Lucas show different ways for creators to ask for help when interacting with AI-powered CSTs. They provide slider controls where creators can vary the degree to which they want the system to support them (i.e., visual similarity 10%). These two CSTs also control the various aspects of the creative process support (i.e., visually vs. conceptually). These examples suggest different ways in which children might work together with their CSTs and engage in co-creativity with intelligent systems [164].

When looking into the world of CSTs for youth, we recognize several tensions around providing support without locking in creators. Therefore, we identified opportunities for designing CSTs that provide an optimal fit for creators' skills and needs, integrate creators' interests and media preferences, and make it easy for creators to ask for and receive help from the system.

Our current creativity support designs fail to support creative expression in the wild. Nevertheless, we can still learn many valuable lessons from existing efforts in creativity support. First, it is essential to acknowledge how little we know and the limitations of our current understanding of creativity. A future agenda on creativity with and for families will need to actively and genuinely respect and listen to children's and parents' voices in co-designing new forms of creative coding in digital, physical, and mixed mediums. Third, beyond the metaphor of ”support,” we have an opportunity to embrace the metaphor of ”collaboration” that positions both family members and creativity support agents on equal footing in the act of creative expression (with code and beyond) [79].

In our study on family creative coding supported by AI friends, presented in chapter 7, we illustrate how family joint engagement enables children to learn creative coding with AI as a way to enable self-expression and provide insights for the design of future AI systems that support family joint-creative coding.

2.6 Summary Related Work

While prior work on parenting in a new media and technology ecology found that children and parents like to learn about home technologies together [41] and joint family engagement with media supports youth digital literacy, we have yet to discover how families can jointly discover the core concepts of AI technologies and the power they can bring. Studies on the bias, power, and critical understanding of AI have highlighted the disparate impact of algorithmic injustice on minoritized communities [?, ?]. However, we have yet to discover how families might foster critical reflection on the uses of AI in the home and beyond. Scholars found that parents play an essential role in supporting children's interaction with AI [47] and can sometimes support their learning about AI [200]. However, we need a clearer understanding of parents' role when supporting children to engage in multiple forms of creative AI literacies. Lastly, prior studies showed that families could positively impact youth creative coding [344], and creative learning is important for families lacking the “preparatory privilege” in computing [258]. In this context, we have yet to explore how families could learn

AI-supported creative coding.

Chapter 3
LANDSCAPE OF AI LEARNING RESOURCES

Figure 3.1: Examples of AI curriculum focused on social impact from ai-4-all.org

Artificial Intelligence (AI) educational resources such as training tools, interactive demos, and dedicated curriculum are increasingly popular among educators and learners. While prior work has examined pedagogies for promoting AI literacy, it has yet to examine how well technology resources support these pedagogies. To address this gap, we conducted a systematic analysis of existing online resources for AI education [1], investigating what learning and teaching affordances these resources have to support AI education. We used the Technological Pedagogical Content Knowledge (TPACK) framework to analyze a final corpus of 50 AI resources (see fig. 3.1). We found that most resources support active learning, have digital or physical dependencies, do not include all the five big ideas defined by AI4K12 guidelines, and do not offer built-in support for assessment or feedback. Teaching guides are hard to find or require technical knowledge. Based on our findings, we propose that future AI curricula move from singular activities and demos to more holistic designs that include

[1] This study was done in collaboration with Nancy Otero and Amy J. Ko and was published in ITICSE '22: ACM conference on Innovation and Technology in Computer Science Education 2022 [96]

support, guidance, and flexibility for how AI technology, concepts, and pedagogy play out in the classroom.

We conducted this study in order to discover how existing efforts in the space of AI education for K12]consider families and developmental needs for youth and uncover opportunities to consider the needs of families for adapting and customizing existing curricula, demos, and tools within their learning ecosystems. This study provided a baseline of comparison for the family Creative AI literacies activities I present in subsequent chapters.

3.1 Study motivation and contributions

Modern computing is rapidly embracing artificial intelligence (AI) for it's great promise in improving our lives via advances in digital voice assistants, AI supported learning and increased accessibility [136, 262, 260]. However, AI systems can also amplify bias, sexism, racism, and other forms of discrimination, particularly for those in marginalized communities [60, 30]. In this context, promoting both technical and sociotechnical literacy of AI in primary and secondary education is critical [98, 199, 305, 247].

How to achieve this, however, is still an open question. Explorations of AI applications in education are challenging since the mechanisms and opportunities of AI are unfamiliar to most people outside of computer science. AI education is considered a vital part of computational thinking [305, 80], and there are arguments to include AI literacy in the primary and secondary education CS curricula [208, 238, 99]. Some works have begun to systematize competencies and skills for AI literacy [199].

One part of achieving AI literacy is the creation of technology resources to facilitate learning and teaching. For example, dedicated coding platforms such as Cognimates[2] and Machine Learning for Kids[3] have emerged to enable AI learning. Organizations like AI4All[4] have also created a free AI curriculum for secondary students. These technologies and their

[2]http://www.cognimates.me

[3]https://machinelearningforkids.co.uk

[4]https://ai-4-all.org/

designs matter [177] as they shape and constrain what content knowledge can be taught. Educators must understand and appropriate AI resources to integrate them into their practice [267].

Despite the proliferation of AI education, prior work has only begun to examine its efficacy and appropriateness for primary and secondary teaching and learning. For example, studies have recently found that whether data is personal can influence student learning [247], that AI curriculum needs to be adapted to different cultural references and languages to become more inclusive [99, 317], that children become more skeptical of machine intelligence if they engage in active training and coding with AI [95], that carefully designed scaffolding is key to learning and transfer of knowledge [148], that gaps in access to technological resources and appropriate infrastructure, especially in the global south, can prevent learning from happening at all [317], and that teaching machine learning differs from teaching computer science as it is not "rule-based" [301].

While prior work has begun to reveal the pedagogies necessary for AI literacy, no prior work has examined the technological resources necessary to support these pedagogies. Prior studies have focused on more narrow aspects of machine learning learning ressources, either by analyzing visual tools for teaching machine learning in K-12 [321] or by doing a systematic review of research efforts on AI education[348]. For our analysis we choose to analyse how existing AI resources support pedagogical efforts and teachers. Therefore, we asked: *What learning and teaching affordances do existing AI resources have for supporting teaching AI?* To answer this, we conducted a systematic analysis of 50 AI resources curated from the most popular AI Education communities in North America: the AI4K12 repository[5], the CSTA repository[6], the MIT AI Education repository[7].

Building on the Technological Pedagogical Content Knowledge (TPACK) framework [177], we formulated a series of questions and criteria to identify the extent to which current

---

[5]https://ai4k12.org/resources/list-of-resources/

[6]https://www.csteachers.org/page/resources-for-virtual-teaching

[7]https://raise.mit.edu

AI learning resources offer the support that educators might need. Overall, we found that AI resources broadly do not consider educators' needs to adapt and customize them for pedagogical use. In the rest of this paper, we elaborate on these findings in detail and discuss implications for design.

3.2 Study procedure

To answer our question, we analyze a corpus of resources that could be used for AI learning. This mirrored prior corpus of studies of learning technologies, such as those considering coding tutorials [169] and programming environments for novice programmers more broadly [167]. Our focus is on resources that explicitly engage AI concepts relevant to AI literacy, including those not necessarily designed to be learning technologies.

3.2.1 Inclusion and Exclusion Criteria

To obtain a corpus of AI resources, we focused on curated lists of resources recommended for primary and secondary educators in North America: the AI4K12 repository[8], the CSTA repository[9], the MIT AI Education repository[10]. From these lists we considered only: curriculum materials, demos, list of links, online course, and software packages.

Based on these lists, the first two authors gathered an initial corpus of 100 resources. They then identified a subset of resources that were still available and functional and removed all duplicated entries, reducing the set to a total of 50 demos, interactive activities, tools, and curricula. The final corpus of 50 AI Education resources together with our final analysis is available here tinyurl.com/aiedk12.

---

[8]https://ai4k12.org/resources/list-of-resources/

[9]https://www.csteachers.org/page/resources-for-virtual-teaching

[10]https://raise.mit.edu

3.2.2 Theoretical Framework

Since our research question focused specifically on teaching and learning concerns, we developed our framing based on theories that would make salient varying levels of support for teaching and learning. Our primary frame was the Technological Pedagogical Content Knowledge framework (TPACK) [177]. Building upon Shulman's Pedagogical Content Knowledge framework (PCK) [278], which posited the existence of knowledge of how to teach particular content knowledge, TPACK makes a similar claim. TPACK analyzes the existence of teacher knowledge of how to use technology (TK), how to use technology to teach (TPK), how technology and content influence and constrain each other (TCK), and how to use technology to teach particular content (TPACK).

We specifically used the TPACK definition proposed by Cox for our investigation, which synthesizes 89 other definitions. Her definition describes TPACK as five connected facets of teacher knowledge: "(1) the use of appropriate technology (2) in a particular content area (3) as part of a pedagogical strategy (4) within a given educational context (5) to develop students' knowledge of a particular topic or meet an educational objective or student need" (p.65) [73]. Each facet describes what a teacher needs to know about technology to use it for teaching and learning.

For the content knowledge dimension of our TPACK framework, we used the AI4K12 guidelines[11], which at the time of this writing defined five "big ideas" about artificial intelligence: 1) Perception: *computers perceive the world with sensors*, 2) Representation & Reasoning: *agents maintain representations of the world and use them for Reasoning*, 3) Learning: *computers can learn from data*, 4) Natural Interaction: *intelligent agents require many kinds of knowledge to interact naturally with humans*, and 5) Social Impact: *AI can impact society in both positive and negative ways.* These ideas provide structure for analyzing the kinds of content knowledge that resources can feasibly help students learn.

[11] https://ai4k12.org

While the above TPACK framework is not necessarily theoretical, it derives from particular theoretical traditions that view teachers as pedagogical experts who develop content and technological knowledge to facilitate student learning [264]. While we acknowledge other more sociocultural [269] and sociopolitical teaching theories [121], our specific focus here is on educators' cognitive and pedagogical needs in their AI teaching practice.

3.2.3 Analysis

Our analysis built on the definition by Cox [73] by devising guiding analysis questions for each of its five facets, leading to 20 questions that structured our systematic evaluation of each resource. For example, one of our questions was "What types of pedagogical strategies does the tool support?" with fixed potential answers (i.e., "interactive learning", "direct instruction", and "hybrid between direct instruction and interactive learning"). The complete listing of these questions is available at tinyurl.com/aiedk12. Both first two authors collaborated on answering these 20 questions for each resource, resulting in a large spreadsheet with labels for each of the five facets of existing teacher support. Any disagreements in answering the questions were discussed until consensus was reached.

3.3 Findings

Overall, there were many distinct genres of resources by various creators: 39% were curriculum collections, 27% were single activities, 18% were demos, and 16% were tools. Only 20% were behind a paywall, though some of the more extended curricula offerings had a prohibitive price (i.e., ReadyAI charged more than USD 2.5k, TeensinAI charged more than 2.5k€). In this section, we evaluate the different genres of existing AI resources concerning how well they support teaching AI.

3.3.1 Communication of Intended Use

We considered the first facet of TPACK educators' need to know what technology is "appropriate" for a given student and learning goal. Therefore, we examined what kinds of

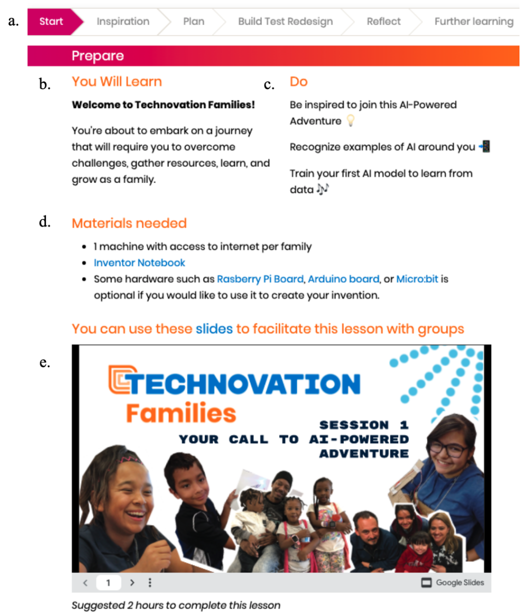

Figure 3.2: Curiosity Machine offered clear guidance to teachers about appropriate use, including: a) clear curriculum progression, b) learning goals, c) activity overview, d) materials description, e) and teaching materials.

information educators might need to judge the appropriateness of analyzing resources. A critical piece of information was the *intended use* of a resource, which illustrates the resource designers' assumptions about users' prior knowledge and context. To analyze resources' intended use, we asked questions such as: "does the resource provide teaching guides?" and "does it provide explanations of the AI concepts it demonstrates?"

Teaching guides were one way to articulate intended use. Overall, we found that 59% of resources offered them. However, some teaching guides were minimal; for example, Zhorai[12] provided brief descriptions of "moderator" and "student" roles without grounding AI concepts and activities in existing curricular standards and practices. In contrast, platforms such as AI4ALL and Curiosity Machine[13] (shown in Figure 3.2) offered clear guidance for

[12]http://zhorai.csail.mit.edu

[13]https://www.curiositymachine.org

educators across several pedagogical dimensions, including learning objectives, pedagogical demonstrations, and materials required.

Another indicator of appropriate use was prior knowledge required to engage a resource. For example, 36% of the resources required users to perform an initial setup before testing or using the AI activity. Many of these setup requirements implicitly assumed particular content knowledge (i.e., terminal use, version control knowledge), with no guidance on how to acquire it. Similarly, while many resources were framed as learning materials—69% offered some written explanations of AI concepts—many explanations were not on the main page of the activity. Still, they were found in other locations like GitHub repositories, further obscuring whether the resource was intended for teaching and learning.

Trends in the clarity of intended use were primarily shaped by the genre of the resources. **Demos**, were often designed to emphasize one or more components of AI functionality, not to teach a comprehensive understanding of AI. None of the demos had teaching guides, only 50% of them explained the AI concepts they were addressing, and just 20% of them allowed participants to change the demonstration's output by modifying either the input data or the parameters. For example, TensorFlow Neural Network Playground[14] (Figure 3.2c) demonstrated how modifying different neural network parameters could lead to different outcomes. This resource offered a separate blog post explaining neural networks but did not integrate the explanation into the experience.

Activities were similar to demos, but often applied AI without a particular teaching goal. Only 30% of activities included teaching guides, only 50% of them explained how a part of AI works, and 62% allowed users to customize their creations. For example, Doodle Bot[15] was an activity for building a bot that uses speech commands to tell a bot what to draw. The activity listed instructions for building the bot and training the AI model with just one paragraph of AI explanation which mentions the pre-trained models used by the system (i.e., "ml5.soundClassifier()").

---

[14]https://playground.tensorflow.org/

[15]https://mitmedialab.github.io/doodlebot

Tools gave even less direction for use. They offered platforms for creating new artifacts. Just two of the tools had teaching guides, but 75% included explanations of how AI works. Cognimates is an example of a tool that could be used to program interactive games using AI by training models to recognize specific images or text. It provided explanations of what algorithms were used to train the models.

Curricula were the clearest about their intended use offering explicit learning progressions for learners. All curricula included teaching guides, AI explanations, and 63% of them included a fixed progressive trajectory. For example, AppsForGood [16] had 14 sequential teaching sessions covering topics from what machine learning is to highlighting careers in machine learning. Of the curricula, 84% used both active learning and direct instruction. AI+Ethics curriculum included several activities that explored ethical questions and AI by doing projects as well as slides that teachers can use to explain AI concepts such as supervised machine learning.

3.3.2 Big Ideas Coverage

The second facet of TPACK is content-specificity: teachers' knowledge of technology must be linked to the specific content knowledge they are teaching. Therefore, we examined the extent to which each resource covered the five AI4K12 big ideas [305].

Resources varied widely in their coverage. Most covered more than one big idea (88%), and most (72%) covered *Perception*. Some, typically curricula, covered all five (24%). The second most prevalent combination of coverage were resources that covered *Perception*, *Representation & Reason*, and *Learning* (18%). These resources were creative tools that typically allowed participants to input sound, images, or video, change the model's parameter, and get an output that showcases how a specific AI algorithm works. These resources typically covered supervised learning and training (28%), neural networks (20%), GANs (12%), image classification (8%), and word embeddings (4%). *Social Impact* was the least common,

[16] https://www.appsforgood.org/courses/machine-learning

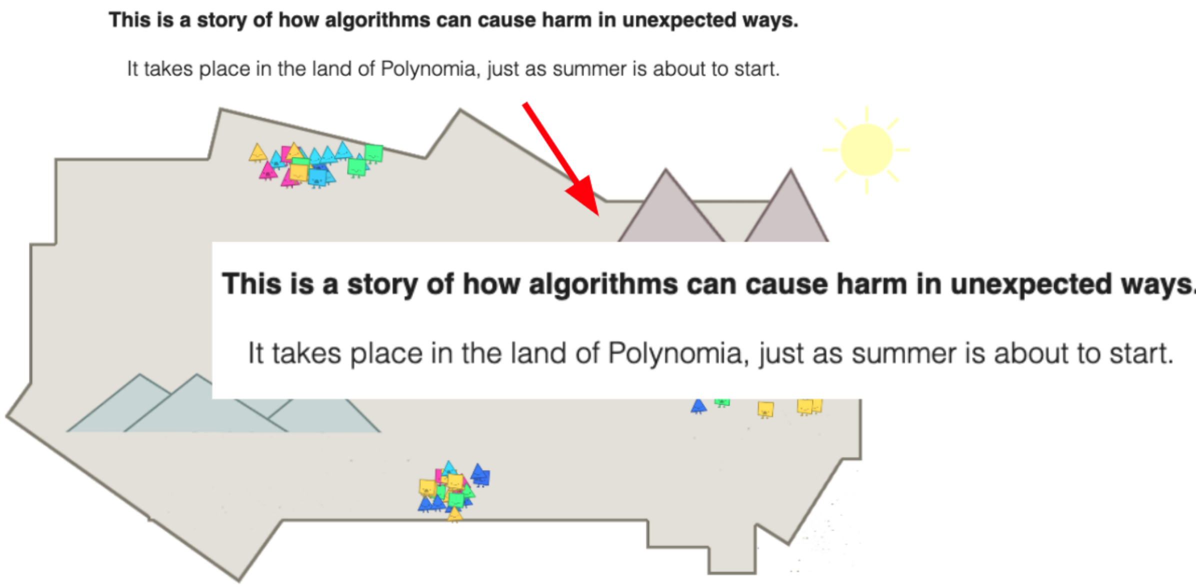

Figure 3.3: The Supervised Polygon activity creatively demonstrated unintended consequences of machine perception.

present only 2% of resources, typically in full curriculum or specialized activities on that topic.

AI big idea coverage varied by genre. For example, **demos** varied substantially in their coverage: 80% covered *Perception*, none covered *Social Impact*, half of the demos covered two ideas, and 30% had just one idea (*Perception* or *Learning*). One example was Pix2Pix which was a website that modifies a picture in real time based on drawings made by the learner[17]. Half of the demos covered two ideas, for example Scrooby, a website that enables participants train a cartoon based on movement perceived on the webcam. For one of the demos, Art Climate Change[18], it was not clear which AI big idea was present. Half of the demos had an explanation of the big ideas they covered.

[17]https://www.tensorflow.org/tutorials/generative/pix2pix

[18]https://experiments.withgoogle.com/cold-flux

Most **activities** focused on *Perception*. For example, Supervised Polygons[19], as shown in Figure 3.3, creatively used data on polygons' shapes (*Perception*) to illustrate AI concepts with unintended consequences (*Social Impact*). Most (84%) also focused on *Learning*; for example, PlushPal used data from the movement of a microbit to train a sound model. Half (53%) explained concepts; for instance, in FarmBeats, learners could use AI to optimize their farms and directly referenced AI4K12 big ideas.

Tools tended to cover at least three of the big ideas, most often *Perception*, *Learning*, and *Representation & Reasoning*. For example, the Personal Image Classification from App Inventor[20], where users could create, train, and test their image classifier and use it to create a game. Most tools (75%) had an explanation of the big idea; for example, Wekinator[21] offered detailed descriptions of algorithms used to train models.

Curricula such as AI4ALL and CuriosityMachine (Figure 3.2) were the most comprehension, with 63% covered all the "big ideas". Some curricula covered the ideas in narrow ways, focusing on a particular technology. For example, Embeducation[22] focused specifically on word embeddings. Nearly all (90%) curriculum had explanations of at least one of the five big ideas.

3.3.3 Pedagogical Strategies

The third facet of TPACK is how teacher knowledge of technologies is tied to particular pedagogical strategies. To examine these resources from this perspective, we analyzed the types of teaching methods resources engaged (active learning, direct instruction, or both) and the extent to which a resource accounted for learner prior knowledge.

Overall, we found that all the resources use either exclusively active learning or integrate active learning and direct instruction. Every resource had some interactive component,

[19] https://supervised-polygons.github.io

[20] https://appinventor.mit.edu/explore/resources/ai/personal-image-classifier

[21] http://www.wekinator.org/

[22] https://embeducation.github.io

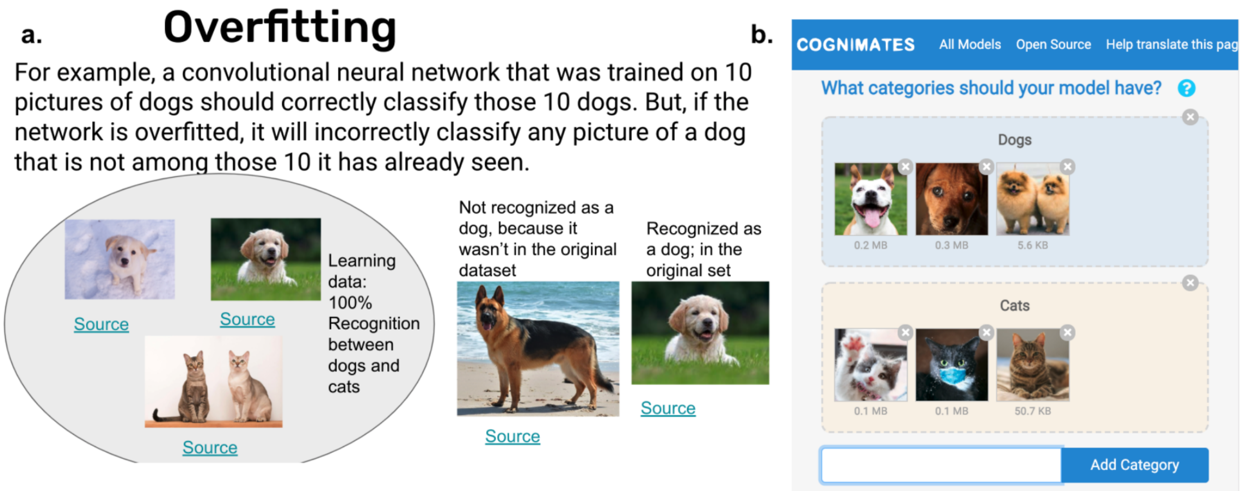

Figure 3.4: Examples of pedagogy integration from AI4All providing both direct instruction a) and active learning using Cognimates.

whether support for creating projects, training a model, or changing the model's parameters and seeing the outcome. We did not find any resources that were designed for purely direct instruction with no opportunity for practice or tinkering.

Despite this consistency in pedagogy, resource genres varied in their implementation. **Demos**, for example, primarily focused on self-contained interactive activities with limited opportunities for tinkering. Moreover, none offered any direct instruction, so it would be up to teachers to integrate them into a broader pedagogical strategy. InferKit[23], for example, was a demo that uses a neural network to generate text; it could support a range of pedagogical strategies involving active learning but offered no detailed guidance on how to do so.

Whereas demos offered unrestrained opportunities for tinkering, **activities** offered more structured active learning experiences with lightweight guidance. For example, Doodle Bot enabled participants to create a robot trained to draw based on speech commands, offering direct step-by-step instruction in tutorial form. About half of these resources offered multiple

[23]https://app.inferkit.com/demo

activities, with 27% giving learners the option to choose their activity and 28% offering fixed sequences of activities. For example, Code.org's AI for Oceans structured multiple activities around training a model to identify fish from garbage, unlocking activities as a learner makes progress.

Tools offered the most learner agency but also offered little scaffolding. Most (62%) gave learners the choice of what activity to do next. An example is RunwayML[24], a tool for creating a video with AI. Its environment offered several opportunities to build knowledge in arbitrary sequences of tutorials.

Whereas all of the other genres generally offered relatively little scaffolding, **curricula** offered the most structure and pedagogical support. The majority (63%) had a fixed sequence of activities. For example, STEM UK[25] was a curriculum with four sequential challenges, starting with an introduction of AI to later centering on the role of AI in making transportation safer, cleaner, and better connected. However, 26% allowed learners to make some choices in their progression. For example, Machine Learning for Kids[26] lets learners select activities based on project types, difficulty, and program environment. Most curricula (84%) used both direct instruction and active learning methods. For example, Figure 3.4 shows how AI4All combined direct instruction about overfitting with opportunities to tinker with overfitting a model by training a dog classifier.

3.3.4 Educational context

The fourth facet of TPACK is the particular educational context in which teacher knowledge is bound. To address this in our analysis, we considered the kinds of educational contexts the AI resources could support, asking: 1) what equipment they required, 2) if teachers might need to prepare a particular technical setup to use the resource, 3) if the resources were designed for a particular level, age, or grade, and 4) if the resources were accessible on

---

[24] https://app.runwayml.com/

[25] https://www.stem.org.uk/resources/collection/447030/grand-challenges

[26] https://machinelearningforkids.co.uk

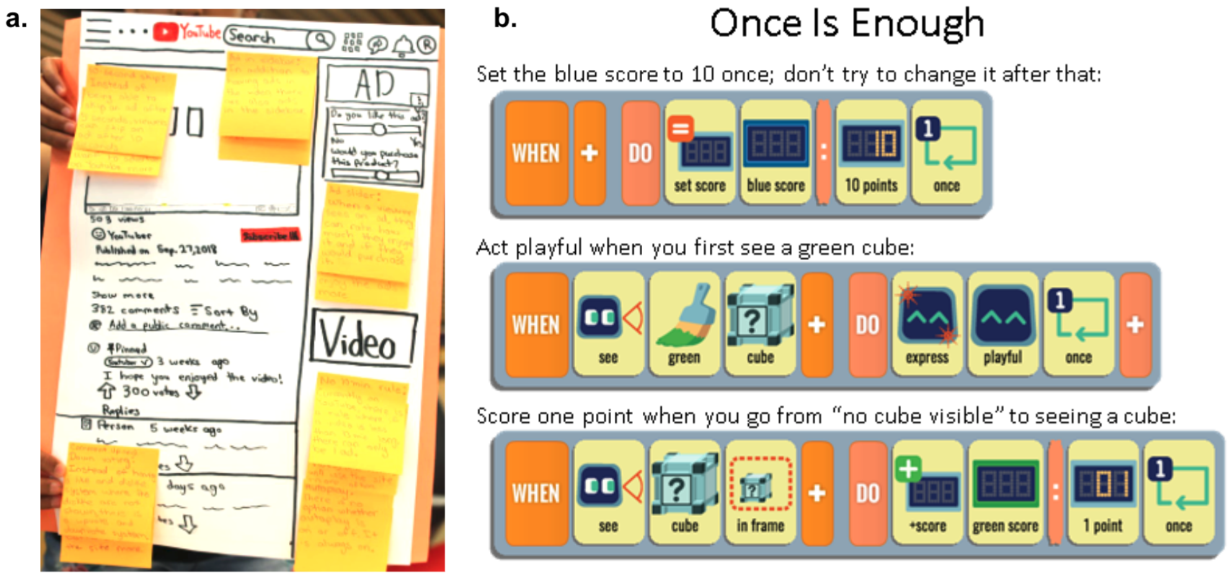

Figure 3.5: Some resources offered unplugged activities requiring no device, including AI Ethics and Calypso's activity sheet.

a tangible or digital medium.

Overall, we found that 36% of the resources required some form of setup either because of their use of hardware, specific technical requirements such as libraries, or the creation of accounts. Most of the resources (62%) were digital-only, but 30% required a physical component, such as an unplugged learning activity or hardware integration. Only 8% of the resources were exclusively non-digital.

Only 59% of resources explicitly noted age or grade level. Of those 59%, most did not have implicit assumptions about either educator or students' prior AI and technical knowledge. For example, Scroobly[27], ModelZoo[28], and Ml5 Tool[29] required prior knowledge of both CS and AI, despite being framed as learning resources.

[27]https://www.scroobly.com/

[28]https://modelzoo.co/

[29]https://ml5.js

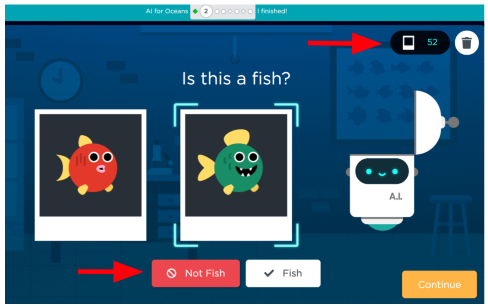

Figure 3.6: The AI Ocean Activity failed to provide any feedback, even when the learner mislabeled fish images.

Each genre had distinct context assumptions. **Demos**, for example, all required computers with sufficient memory and compute power as some of the AI models they used were RAM intensive, but none required a technical setup beyond a web browser. Of the **activities**, 58% required some additional technical setup, and 25% had instructions for age and grade levels. Those that involved hardware, such as AIY kits for vision and sound [30], required significant familiarity with hardware components and technical setup. More than half (57%) of **tools** required a technical setup; all required computers or mobile apps. Fewer than half (43%) offered specific instructions regarding the age and grade levels of users. **Curricula** had the fewest technical requirements, with only 33% requiring configuration. However, all but two curriculum resources required the use of computers; the exceptions, shown in Figure 3.5, included AI Ethics[31] and Calypso[32], both of which involved activities that used paper and writing utensils instead of computers. Most curricula (77%) had age- and grade-based guidance, though several left the intended audience unstated.

---

[30]https://aiyprojects.withgoogle.com/vision

[31]https://www.media.mit.edu/projects/ai-ethics-for-middle-school/

[32]https://calypso.software/

3.3.5 Support for practice and assessment

The fifth and last facet of our TPACK analysis concerns how knowledge is deployed to develop students' knowledge. We, therefore, focused our analysis on how resources could support teachers in facilitating practice and assessment, analyzing if each resource: 1) provided support for practice and assessment, 2) provided opportunities for personalizing the learning experience, and 3) supported collaborative learning. Overall, we found that 68% of the resources supported practice and assessment, 64% provided opportunities for customizing the learning experience by allowing teachers to either change the parameters of the resources or change the training data. In total, 40% of the resources supported collaborative learning.

Demos offered the fewest support for practice and assessment: only 33% supported repeated practice, only 22% allowed teachers to customize the configuration for learning, and only 11% allowed collaborative learning. None offered explicit support for assessment.

Activities tended to support practice (58%), often by allowing users to engage more in customizing either the input for the AI demo (i.e., record specific gestures like in the case of Plushpal[33]) or by customizing the output of the demo by changing how the demo output is displayed (i.e., Teachable Machine, allowing users to choose animations, text or sound). In total, 66% of activities supported the customization of the AI experience parameters, 58% had support for the practice, and 33% supported collaborative learning. Most activities offered no form of feedback on learners' actions; for example, the AI Oceans activity shown in Figure 3.6 allowed learners to label fish however they wanted and offered no explanation of how that might affect training.

Most **tools** (87%) offered substantial opportunities for practice. For example, iNaturalist[34] was a tool that used AI to support citizen scientists in classifying organisms. It had a path to practice adding IDs of an organism, comments, and observations before creating a project. On this platform, participants could post as many projects as they want. Most of

---

[33]https://www.plushpal.app/

[34]https://www.inaturalist.org/

the tools (75%) also allowed participants to personalize and customize their creations. One of the tools that do not allow it was Jukebox[35], a neural net that generated music. Jukebox let learners play with the creations of the model but unless participants could run the model on their computer they could not create their music. Another tool, AI Playground[36], allowed users to go more in-depth in modifying the AI parameters by controlling the number of training cycles (epochs). In some cases, tools tried to scaffold practice with activity sheets. Many sheets might be confusing because they introduced many new terms and references. For example, the activity sheet from Calypso (shown in Figure 3.5b) was meant to support users to learn how to program a robot but it could be difficult to grasp because it introduces a new programming language together with a series of new icons and terms.

All of the **curricula** we could access had activities for participants to practice AI concepts. For example, the AI and Machine Learning Module at Code.org[37] taught AI concepts at several different levels. Most curricula (89%) had the option to input customized data and personalized the outcome of the activities. Another example in this group is AI Ethics. The last module in this curricula is about YouTube re-design. Participants in this activity learn how YouTube uses AI, select what features they want to re-design, and have the option to present their mock-ups.

3.4 Discussion

Overall, our analysis found the following:

- *Intended use.* Most resources, even those not designed for teacher use, had guidance that conveyed intended use. But the direction was often hard to find or required obscure technical knowledge to find and comprehend.

[35]https://openai.com/blog/jukebox/

[36]https://theaiplayground.com/

[37]https://studio.code.org/s/aiml-2021

- *Content.* While most of the resources covered many of the AI4K12 big ideas, most did not cover all five, in most cases overlooking *Social Impact.* Curricula were the most likely to cover all five.

- *Pedagogy.* Most resources supported direct instruction and active learning combinations, though few were responsive to learners' prior knowledge.

- *Educational Context.* Most resources had some form of device dependency, constraining the learning and IT contexts in which they were compatible. Demo hardware requirements could be quite prohibitive for schools that do not have access to updated computers [317].

- *Student Learning.* While most resources offered substantial opportunities for individual and collaborative practice with AI concepts and skills, few offered assessment support or learner feedback.

In some ways, these findings reflect prior work on other classes of CS educational technologies. For example, Kim and Ko's evaluation of coding tutorials found a similar focus on active learning, a similar lack of communication about intended audience and context use, a lack of responsiveness to the student prior knowledge, and a disregard for formative and summative assessment [169]. Our results also mirror Kelleher's review of novice programming environments, showing a bias toward tinkering over direct instruction [167]. Our results also mirror the experience of educators who are currently designing their AI curriculum and directly expressed the need for support to combine the various AI resources and create a friendly learners interface [267, 317].

Our evaluation adds to these prior works in two ways. First, our results suggest that AI learning resources repeat some of the same mistakes of non-AI CS educational resources. Second, our results expand upon this, showing that many of the needs educators might have in developing TPACK to use AI resources aren't yet supported. Most resources do not clarify their assumptions about learner prior knowledge, required classroom resources context, alignment with pedagogical strategies, or even intended use. Even many of the curricula we analyzed were vague on these points. Some of the resources were consistent with implications from recent studies (e.g., leveraging personal data to an extent [247], embracing emerging student skepticism about AI [99], and leveraging embodiment [316]). But most resources did not meet basic pedagogical design principles, let alone offer the information teachers need to develop TPACK appropriate for successfully using the resources.

These findings have several implications for research. Future work might explore creating design principles for CS educational technology designers and understanding the barriers designers face in meeting those principles. In some cases, research is needed to achieve these principles. We see an opportunity for educators and designers to develop a common language based on a common set of guidelines, similar to the five big ideas [305]. For example, "features" could be described as "observable detail of object", "training" as "machines learning from data", and "model" as "application of what the machine has learned".

In terms of practice, our results suggest that until resource designers are more explicit about the various dimensions of TPACK in resource content, metadata, and design, teachers will have to make complex judgment regarding what resources might be appropriate for their students' learning. The curricula in our corpus generally fared best from a TPACK perspective (though not all were equal), with only two at the time of this writing—Curiosity Machine and AI4ALL—offering a clear path to adoption for teachers. Perhaps with time, resource designers and educators will find better ways of partnering, ensuring that all AI education resources can empower teachers to better facilitate AI education for all.

Based on our findings in this study, I proposed that future AI curricula move from singular activities and demos to more holistic designs that include support, guidance, and flexibility for how AI technology, concepts, and pedagogy play out in different learning scenarios. Moreover, most of these AI learning resources need to consider the needs of families for adapting and customizing existing curricula, demos, and tools within their learning ecosystems. This motivated me to explore how future AI learning resources for K12 could consider families and the developmental needs of youth. In the next chapter, I will present how such resources could be co-designed with kids and parents.

Chapter 4

CO-DESIGN ACTIVITIES FOR AI LITERACIES WITH FAMILIES

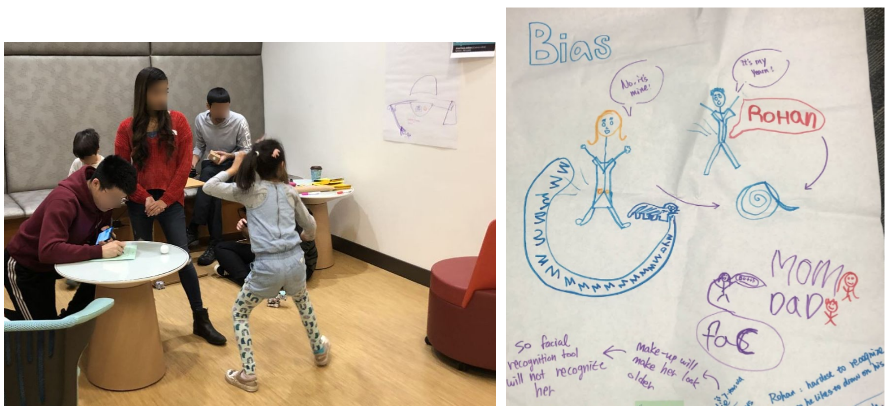

Figure 4.1: (Left) Examples of families engaging with the Smart Toys activity during our co-design sessions;(Right) Examples of bias instances identified by children in the first and second sessions.

Families' interactions with AI technologies have recently gained attention. However, these technologies do not provide developmentally adaptable, and family-friendly interactions [48, 102]. Therefore, I propose a framework to support family AI literacies, composed of four main dimensions (4As): ask, adapt, author, and analyze. Families can use this framework to develop a critical understanding of smart technologies and ensure algorithmic fairness [1].

We define our AI literacies dimensions based on prior work and through co-design and AI learning sessions with families. This study reveals how children perceive algorithmic bias differently from adults and how families engage in collaborative sense-making by probing,

[1]This study was done in collaboration with Jason Yip, Michael Preston, and Devin Dillon and was published in MIT Press Journal for Algorithmic Rights and Protections for Children 2021 [103]

tricking, and authoring AI applications in playful ways. As a result, parents should be included in future designs of AI education for children.

We discuss the implications of family AI literacies from the broader perspective of technology development, public policy, and algorithmic justice. Finally, we argue that AI literacies is a fundamental right for families and propose a series of learning activities and guidelines to support and protect this right.

4.1 Study motivation and contributions

Without AI literacies, families, mainly from historically marginalized groups, risk falling prey to misinformation, fear, and missing opportunities for future potential for learning [113, 127, 229]. Families and children must work together to learn about AI systems and to think critically about how this technology impacts their lives [98]. Prior research on family engagement with digital technologies stressed the importance of considering variation between families and parenting styles [294, 74]. Therefore, to support algorithmic justice in families, we need to consider how many families can access these skills [339, 84].

AI literacies do not occur in a vacuum but are influenced by social, cultural, institutional, and techno-infrastructural contexts. Therefore, we need to consider the ecological and situational issues surrounding families and how macro and micro-factors influence AI literacies in the modern family. Therefore it is crucial to address the socio-ecological conditions that influence how families may adopt AI literacies and to create guidelines that integrate human-centered design into practice. An analysis of ecological systems [58] can explain how families could succeed with AI literacies and unveil the broader implications of such an intervention.

A survey of 1,500 parents of elementary and middle school students, commissioned by Iridescent [Technovation, 2018], found that 80% of parents in the United States believe AI will replace the majority of jobs (not just low-skilled jobs), less than 20% understand where and how AI technologies are currently used, 60% of low-income parents have no interest in learning about AI. Furthermore, less than 25% of children from low-income families have access to technology programs [69].

Little knowledge exists on how parents or guardians learn with their children using tools for AI literacies. To address this gap, we pose the following research questions:

- RQ1: How do children and parents from different countries and diverse socioeconomic status (SES) perceive and interact with AI?

- RQ2: How can we best support parents to scaffold their children's use of AI technologies in the home?

- RQ3: How can we design future technologies to best support families' AI literacies?

Our goal is to understand how to facilitate AI literacies in families. We investigate this from two perspectives: an ecological evaluation of current AI systems and designing new systems for AI literacies. Our research provides a conceptual and empirical understanding of how families engage with AI literacies activities, which can inform the design of culturally-tailored tools and resources.

We contribute new insights on family AI practices to address critical AI literacies needs in families. Additionally, we develop a foundation to encourage innovations that take advantage of family dynamics for AI literacies learning. Finally, we analyze and compare prior data sets to propose a novel research-based family-facing framework for thinking with and about AI.

We begin by reviewing ecological systems as they pertain to supporting AI literacies [57]. Ecological systems theory refers to the multiple nested systems (i.e., exosystems, macrosystems, mesosystem, microsystems) that influence people's learning development. Through a review of the literature, we consider how current technological systems support or not the development of AI literacies. From our evaluation of ecological systems in AI literacies, we develop a design framework for supporting critical understanding and use of AI for families. This study proposes a framework which considers four dimensions of AI literacies (Ask, Adapt, Author, and Analyze). We prototype and refine learning activities such as detecting bias, testing a voice assistant, coding a smart game, and drawing what is inside the smart devices to explain how they work.

Through family co-design sessions, we found that children perceive bias in smart technologies differently than adults and care less about technological shortcomings and failures as long as they have fun interacting with the devices. In addition, family members supported each other in various collaborative sense-making practices during the sessions by building on each other's questions, suggesting repairs for communication breakdowns with the voice assistants, coming up with new and creative ways to trick the AI devices, and explaining or demonstrating newly discovered features.

We demonstrate how our framework supports the development of AI literacies through play, balanced partnership, and joint-family engagement with AI learning activities. We provide a set of guidelines for families and engage in a broader discussion that connects the ecological systems theory with our AI literacies framework to draw implications for the broader perspective of practice, program design, public policy, and algorithmic justice.

4.2 Study procedure

Through our analysis of the ecological perspective on the current state of AI understanding for families and building on theories of parental mediation and joint-media engagement [294], we propose a new framework for defining family AI literacies (see Table 4.1). To examine our framework in action, we adhere to the standards and practices of Participatory Design (P.D.), precisely the method of Cooperative Inquiry [104, 137]. Under Cooperative Inquiry in P.D., adults and children work closely together as design partners, emphasizing relationship building, co-facilitation, design-by-doing together, and idea generation [342].

Cooperative Inquiry works well for understanding AI systems and literacy because children already work closely with adults and are more likely to express their perceptions around childhood [335]. In addition, in design partnerships, there is a strong emphasis on relationship building, which allows children to be more open to experimentation and open dialogue.

AI literacies Dimension	Family Activity	AI literacies Question
Ask	Interact fluently with an existing AI application or technology	*How do you make it do...? Do you? Are you?*
Adapt	Modify or customize an AI application to serve their needs	*How do I modify it?*
Author	Create a new AI application	*How do I make a new one?*
Analyze	Analyze the data and the architecture of their AI application and modify it to test different hypotheses	*How does it work? What if*

Table 4.1: The 4 A's: Proposed Framework for Families AI literacies Dimensions

Our co-design sessions focused on designing and eliciting responses from children and families around their perceptions of different aspects of AI systems. We conducted three 90-minute sessions from October to November 2019 with 8 - 11 children. We also worked with families in co-design sessions in December 2019 to understand children's engagement with AI together with their parents.

4.2.1 Study participants

An inter-generational co-design group of adult design researchers (undergraduates, masters, and doctoral students) and child participants (n = 11, ages 7 - 11) participated in the four design sessions. The team is called KidsTeam UW (all names are initials). At the time of the study, children typically participated from 1 - 4 years (2016 - 2019). In the fourth session, three KidsTeam UW children and their families (e.g., parents and siblings) came on a weekend co-design session to engage and discuss their perceptions of AI technologies.

4.2.2 Design sessions

Each design session (child and families) at KidsTeam UW consisted of snack time (15 minutes), where the children gathered to eat, share, and develop relationships through play. In Circle Time (15 minutes), we provided children with a "question of the day" to make them think about the design session. We also provided the instructions for engagement (verbal facilitation and activity printouts). The majority of the time was spent designing together (45 minutes), in which children engage in some design techniques with an adult partner(s) [327]. Children break into smaller teams or remain together in a single design activity. Finally, the group comes back together in discussion time (15 minutes) to reflect on the design experience. We organized the sessions in the following way to investigate how the family AI literacies framework could be utilized as a series of design activities:

- **Design Session 1** (October 2019): We showed the children different video clips of "algorithmic bias." Video clips included AI not being able to recognize darker skin tones, voice assistants stuck in an infinite loop, and a very young child unable to get an Alexa Echo device to start. We used Big Paper [328], a technique that allows children to draw on large sheets of paper to reflect and consider what "bias" means.

- **Design Session 2** (October 2019): We provided children with different technology activities with three kinds of AI devices: Anki Cozmo (AI toy robot), Alexa Echo voice assistant, and Google Quickdraw (AI recognizes sketches). Each inter-generational team went through the stations and documented what was "surprising" about the technology and if they could "trick" the AI system into doing something unexpected.

- **Design Session 3** (November 2019): Using Big Paper, we asked children and adults to draw out how they thought a voice assistant (Amazon Alexa) worked.

- **Design Session 4** (December 2019): Finally, five KidsTeam UW families came together on a weekend morning workshop to engage in multiple AI technologies stations.

Stations included Amazon Alexa, Google QuickDraw, and the Teachable Machine. One station, in particular, used Cognimates [97] and BlockStudio [36] to show models of how computers made decisions. Families spent, on average, 15 minutes per activity trying out the different technologies and wrote down their ideas and reflections on the technologies

4.2.3 Data Collection and Analysis

The first two authors used an inductive process to analyze the audio capture family AI interaction themes [66]. We began with memoing and open coding during the initial transcriptions of the video files. Through memoing and open coding, we noticed emerging themes related to family AI literacies practices and joint family engagement. We then began coding literacy practices and joint engagement from transcripts of each of the five families, developing and revising codes as we found additional examples of AI-joint engagement, reviewing a total of 17 hours of video capture. We continued this process until codes were stable (no new codes were identified) and applicable to multiple families. Disagreements were discussed until consensus was reached. Once the codes were stable, we reviewed transcripts from each of the five families for AI literacies practices and joint family engagement again. We included AI literacies practices from each participant in our corpus of 350 AI family-AI interactions, systematically going through each family's transcript and pulling out each code (when present). For our final analysis of the family's AI interaction, a total of 180 AI interactions falling under the broad themes of AI literacies practices were deeply analyzed by two researchers. AI literacies practices were defined as interactions between family members and the various AI technologies, as defined in table 4.1. We drew on the human-computer interaction conversational analysis approach to analyze family interactions set in an informal learning environment, focusing on the participants' experiences.

4.3 Findings: The 4A Framework In Action

4.3.1 Ask Dimension - identify AI bias

When we initially asked children to describe what bias means and give examples of bias, we found ourselves at a crossroads as we realized none of our participants understood what this term meant. However, we quickly noticed that children understood the notions of discrimination and preferential treatment and knew how to identify situations where technology was treating unfairly specific groups of people.

"Bias? It means bias" - L. 7, years old boy. During the initial discussion in the first study session, we tried to identify examples of bias that children could relate to, such as cookies or pet preferences. For example, when talking about cat people versus dog people, D., a nine years old girl, said, "Everything they own is a cat! cat's food, cat's wall, and cat(...)". We then asked the kids to describe dog people. A., an 8-year-old boy, answered: "Everything is a dog! The house is shaped like a dog, bed shapes like a dog". After the children shared these two perspectives, we again discussed the concept of bias, referring to the assumptions they made about cat and dog people.

Race and Ethnicity Bias. In the final discussion of the first session, children could connect their daily life examples with the algorithmic justice videos they had just watched. "It is about a camera lens that cannot detect people with dark skin," said A. while referring to other biased examples. We asked A. why he thinks the camera fails this way, and he answered: "It could see this face, but it could not see that face(...) until she puts on the mask." B., an 11 years old girl, added, "it can only recognize white people." These initial observations from the video discussions were later reflected in the children's drawings. When drawing how the devices work (see fig. 4.2), some children depicted how smart assistants separate people based on race. "Bias is making voice assistants horrible; they only see white people," - said A. in a later session while interacting with smart devices.

Age Bias When children watched the video of a little girl having trouble communicating with a voice assistant because she could not pronounce the wake word correctly, they quickly noticed the age bias. “Alexa cannot understand baby’s command because she said *Lexa*,”- said M., a 7-year-old girl; she then added: “When I was young, I did not know how to pronounce *Google*,” empathizing with the little girl in the video. Another boy, A., jumped in, saying: “Maybe it could only hear different kinds of voices,” and shared that he does not know Alexa well because “it only talks to his dad.” Other kids agreed that adults use voice assistants more.

Gender bias After watching the video of the gender-neutral assistant and interacting with the voice assistants we had in the space, M. asked: “Why do AI all sound like girls?” She then concluded that “mini Alexa has a girl inside and home Alexa has a boy inside” and said that the mini-Alexa is a copy of her: “I think she is just a copy of me!” While many of the girls were not happy that all voice assistants have female voices, they recognized that “the voice of a *neutral gender* voice assistant does not sound right” -B., 11 years old. These findings are consistent with the UNESCO report on implications of gendering the voice assistants, which shows that having female voices for voice assistants by default is a way to reflect, reinforce, and spread gender bias [4].

4.3.2 Adapt Dimension - Trick the AI

In the second design session, we invited participants to engage directly with the smart technologies and see if they could trick them. We wanted to provide the children with concrete ways to test the device’s limitations and bias, and we learned from our prior studies that children enjoy finding glitches and ways to make a program or a device fail [89]. Such prompts not only give them a sense of agency but also provide valuable opportunities for debugging and for them to test their hypotheses about how the technology works. During our workshop, children imagined and tested various scenarios for tricking smart devices and algorithmic prediction systems. For example, when playing with Anki’s social robot Cozmo, they disguised themselves with makeup, masks, glasses, or other props so the robot could no longer recognize them. They also decided to disguise other robots and make them look like humans and see it would trick the robots’ computer vision algorithm (see fig. 4.3). Children

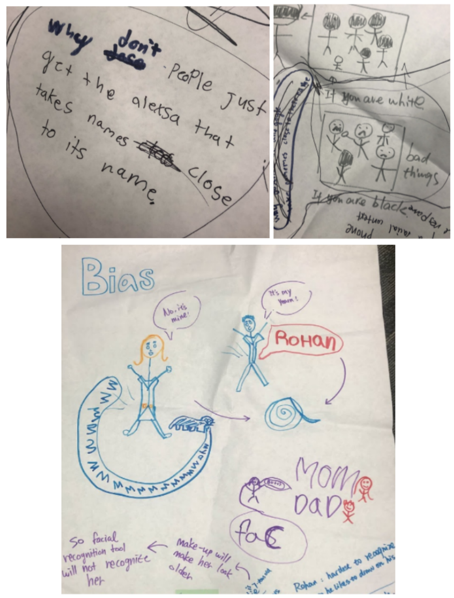

Figure 4.2: Examples of bias instances identified by children in sessions 1 and 2.

also used this strategy in our prior AI literacies workshops for families in Germany, and it is a fun activity that could easily be replicated at home.

When playing with the Quick Draw app, children were amazed at how quick and efficient the program was in guessing their drawings, so they decided to deploy many different strategies to confuse the program. They first tried to draw nonsensical drawings and see if they would still get object predictions after they decided that multiple children should try to draw on the same device at the same time so that the program would have a hard time keeping up with their drawing speed. When interacting with Amazon's voice assistant, Alexa participants found various ways to probe if it is biased. In essence, they tried to speak Spanish and see if the device would recognize a new language; they used different names for calling the device "Lexa" to see if it could deal with more informal language, and they asked "silly" questions to see if the device can engage in child play (i.e., "Call me princess"), they also tried to see if it can sing songs from different locations such as the North Pole or the Indian Ocean. Very often, children build on each other's questions during the interaction and

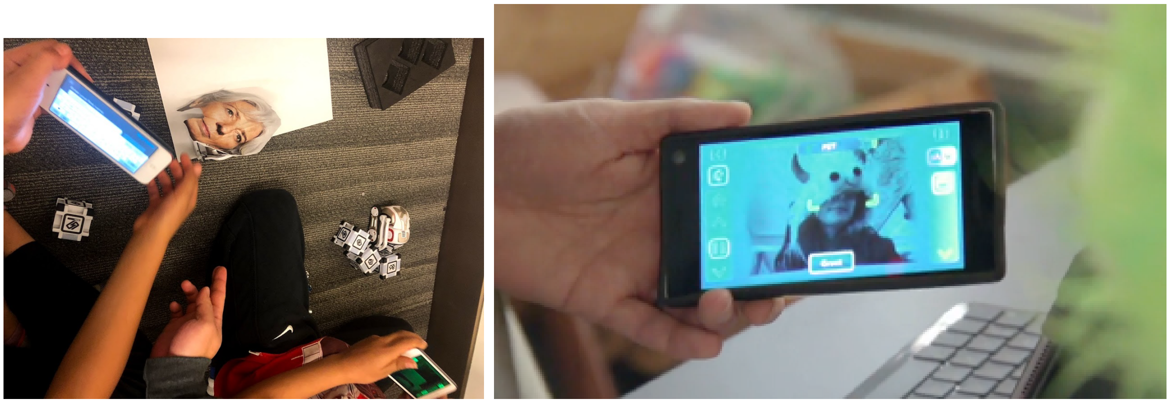

Figure 4.3: Examples of ways in which children are trying to trick Cozmo robot

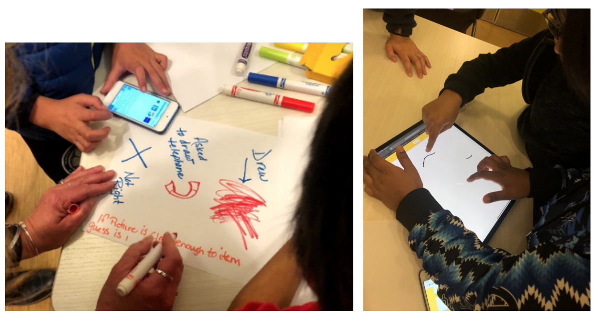

Figure 4.4: Examples of ways in which children were trying to trick the AI

help each other reformulate a question when needed. This finding is consistent with prior work done in this field, where we learn how much peers or family members can help repair communication breakdowns when interacting with voice assistants [100, 48]. While trying to probe and trick the voice assistant, children voiced several privacy concerns: "Amazon can hear everything users have said to their Alexas," said A. he then added, "Alexa buys data, takes data, and gives it to people who build Alexa." D. was worried that "the tiny dots on Alexa are tiny eyes where people can see users," so she decided to cover the device with post-its. From these examples, we see how children's privacy concerns can vary widely based on their naive theories [153], prior experiences with these technologies, and conversations they had with or heard from their parents.

Figure 4.5: Examples of children coding a game with BlockStudio

4.3.3 Author Dimension - design, code, teach the AI

The democratization of current AI technologies allows children to communicate with machines not only via code but also via natural language and computer vision technologies. These new interfaces make it easier for a child to control and even "program" an agent via voice, but it is harder for a child to debug when the machine does not behave the way he expects. During our design sessions, children could individually discover a series of AI programming applications and use them with their parents. Sometimes families would start by playing with example games that would recognize their gestures or objects. We would then ask them to make the games more or less intelligent. Other times families would come up with their project ideas and start a program from scratch. We would ask the children to explain specific concepts from their project. "What does the loop mean?" asked one of the researchers. M. answered by drawing a circle in the air. We also asked both children and parents to reflect on how they can make the technology suitable and meaningful for their families. D.'s older sister said they could program the Sphero ball robot for "maybe dog chasing."

In all the authoring activities, families tried to test their programs in various ways, moving their bodies together, standing up, and sitting down. Meanwhile, one of the family members was going back and forth to modify the code blocks or the parameters of the smart games to see what would happen. Children and parents engaged in a balanced partnership, especially when using the applications where it was straightforward for multiple people to interact with the program (i.e., Quickdraw, Cognimates motion games, Teachable machine vision training). Similar to prior studies, parents helped scaffold their children's behavior when interacting with robots or interactive devices together [64, 120].

When M. and her dad played with the Teachable Machine Platform (ref fig. 4.5), the dad would always probe his daughter with helping questions. "So I put in 150 pictures, and you put in 25, so that model knows me better because I put more pictures in it. The more pictures I put in, the more the model will learn. How would you fix it?" asked M.'s dad. M. replied, "add more," and proceeded to add more pictures of herself. When she realized she could not add more pictures after a model was trained, she would say, "No, we have to redo it. Daddy goes first this time." After training their model for a second time, M. and her dad tried to trick it, and both faced the camera simultaneously to see which one would be recognized. M. noted that the machine looked very similar, but she had a pink bow, and she thinks that is why the machine could recognize her. She thought of another way of tricking the machine into giving her pink bow to her dad.

We observed the same behavior when families interacted with voice assistants. All family members helped each other to repair various communication breakdowns, similar to prior studies [48]. For example, R.'s dad tried to get the voice assistant to act like a cat. He said "meow" when talking to the device. "Oh, you have to say something," replied R., his 11 years old son, then R. added, "if you wanna wake her up, you should say something like *Alexa*." The device turned blue, and R. said, "meow." After, the voice assistant started to meow.

From these examples, we see how children build on experiences and skills developed in the prior study sessions for probing the technology as they are designing it, either by asking it questions, trying to trick their games, debugging collaboratively, or by teaching and supporting each other. In this way, our **Ask**, **Adapt**, and **Author** framework dimensions become intertwined in practice and serve as a support in helping families gain a more in-depth understanding and control of AI technologies.

4.3.4 Analyze Dimension - How does it work? How do we make it better

The last step in our design sessions with families was critically analyzing the technologies discussed, used, or created in all the other study sessions. This critical analysis was done as part of a group discussion at the end of the study in which children, parents, and researchers participated in a circle. The analysis was also done throughout the other sessions when we asked participants to draw and explain how the devices work and what they have inside. With these prompts, we aimed to discover the families' mental models of AI technologies and observe how these explanations draw on or influence their direct interaction with smart devices. The purpose of **Analyze** discussion was also to elicit systematic reframing for families to reflect on how they might make better use of AI systems in the future and think about when and if they should use such technologies.

What is inside? In order to help uncover how children conceptualize smart devices, we asked them to draw what is inside the device and explain how it works. Children resorted to various representations and explanations: by saying there is a computer inside, a series of apps, a robot, a phone, or a search engine. "There is a search engine inside the Alexa, but I do not know what it looks like," said L., a 10-year-old boy.

Y. and S., two 9-year-old girls, said that an army of people sits at their computers inside the "Company of Alexa" and replies to all the questions after they research the answers online. "There is a bunch of cords and a speaker inside the Alexa. It would connect to a computer and link it to Amazon people. So, for example, if the question is what is the weather, it [the person] would search the weather and type it up and let Alexa say it," said Y., a 9-year-old girl.

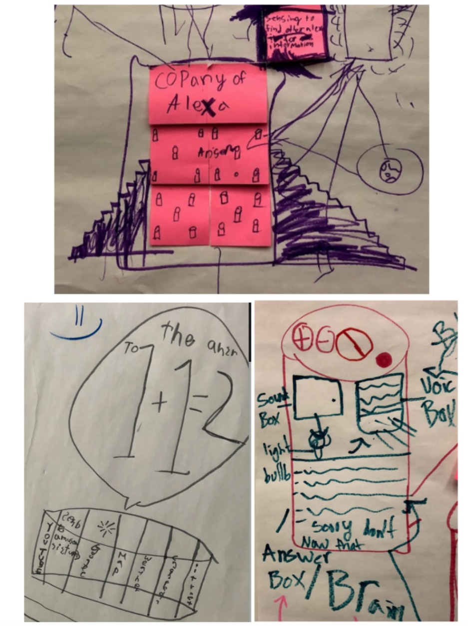

Figure 4.6: Examples of drawings from children explaining what is inside the voice assistant

The most common analogy children made was that of the mobile apps they were familiar with. Children imagined how the voice assistant would use different mobile apps depending on the user's question (see fig. 4.6). D., another 9-year-old girl, also imagined how the different devices are linked to each other: "if Alexa does not know an answer, it asks other Alexa before asking Amazon; once one Alexa gets the answers...every single Alexa in the world will get that answers." The younger children (6-7 years old) provided more vitalistic explanations, consistent with prior studies [153]. "There is a brain inside Alexa, and there is a part that connects to a computer with a speaker. The speaker will shout out the answer," said M., a 7-year-old girl. The older children (8-11 years old) had a very different explanation, which was primarily related to other technologies or applications they are currently using:"Alexa looks at every place it can search for an answer: Amazon, YouTube, Internet, Weather, Map, any place" said A., an 8-year-old boy. "The database is a box with stuff in it. The stuff is statements you tell Alexa," added R., an 11-year-old boy.

It is as simple as 2+2. During the design sessions, children tried to validate their mental models by probing the different devices with questions. Children also tried to determine the age of the devices to determine how much they could trust them. When they asked how old it was, children were disappointed by the answer Alexa gave: "it is as simple as 2+2." They described this answer as "questionable," as they would find it hard to believe a voice assistant could possess so much information at age 4. B. said the assistant must be at least 20 years old. When children find bugs or limitations in the device's answers, they think the errors happen because the device "relies too much on the internet." Children requested to know who programmed the voice assistant to understand why the device was lying about its age. From this example, we see how our participants were able to draw on prior workshop experiences and not only understand how the device behavior is linked to the way it was programmed but also figure out what questions to ask in order to test the device.

4.4 Discussion

Today's modern world is now governed by the decisions made through AI and algorithms. While these tools show incredible promise in healthcare, education, and other fields, there is also a need to support ways in which people (mainly from vulnerable and marginalized populations) can carefully critique the ways AI could amplify racism, sexism, and other forms of discrimination. For people to start considering algorithmic justice early, we must find ways in which they develop forms of literacy around AI. We argue that AI justice and AI literacies begin in early interactions, inquiries, and investigations in the family unit.

However, AI literacies are not a form of knowledge that can be taught in a didactic and lecture-based form [89]. Instead, designers need to consider how to promote sense-making, collaboration, questioning, and critical thinking. How to design future AI systems for families tapping into the idea of "children as scientists" and leveraging their curiosity and both the explore/exploit paradigms? Prior work shows that children are developmentally primed for this type of exploration [131], and we believe it is a missed opportunity to not provide AI literacies opportunities through the design of future smart technologies and via parenting.

Based on our prior research and the findings of this study, we propose a novel AI literacies framework for designers and educators to consider to support critical understanding and use of AI systems for families. Furthermore, we believe it is important to consider this design framework in the context of our current analysis of nested ecological systems [58].

In **Asking** sessions, children and families can inquire and interact with AI agents through various means, such as calling out with voice interactions, drawing, and playing. However, embedded in these interactions with Asking is the notion of privacy policies that must be transparent for families (exosystem). Families have several questions about privacy, technology, policies, and their children [346]. Therefore, how do we support families to ask and interact with AI agents in a way that deems their information safe and confidential? Designers must also consider how at-home interactions happen between children and families (microsystems). In this context, can families collaborate and ask AI agents? How do prior relationships in families mediate how comfortable family members are to engage with AI at home?

With **Adaptation** sessions, families are shifting and mitigating their perceptions and engagements around AI to fit their contexts. However, in adapting to AI, there remain questions of negotiation and power [39]. AI systems cannot code switch and recognize children and adults [48]. How are more substantial cultural capital and social contexts (macrosystems) of families thought about with AI? For instance, bilingual families can switch and merge languages (e.g., Spanglish). For AI voice assistants, this means having to adopt a single language. Similarly, AI systems have difficulty recognizing different languages and accents (macrosystems). In this case, families who may have grown together in specific social and cultural norms now face systems that cannot adapt to these larger macrosystems.

For the **Author** dimension, families need a chance to build and create to develop AI literacies. We ask, though, who has a chance and opportunity to build? Even if designers create authoring systems for AI engagement, this can solely depend on technology infrastructure at home (exosystems)[253]. Authoring may also mean learning how to build, which may privilege individual families in communities, libraries, schools, and networks that can teach and build knowledge capacity.

Finally, under **Analyze**, the design of AI learning tools can be situated towards collaboration, and sense-making [237, 31]. However, this approach assumes that different family units work together (microsystem). Therefore, how is a careful reflection on AI designed to deal with real family constraints, like working families, families with limited time, and families that always move (i.e., children living between households)? How might designers create activities and technologies that support diverse families to generate and test various hypotheses about how smart technologies work and engage in the systematic reframing of how AI should work to support meaningful and inclusive family activities [79]?

While complex ecological systems need to be considered within design frameworks, there are still takeaways for families with AI literacies and justice. Our study shows that with the Ask, Adapt, Author, and Analyze dimensions, parental roles and relationships still matter when families learn about AI together. Aarsand (2007) describes "asymmetrical relations" between parents and children concerning assumptions about expertise with computers and video games as both a challenge and an opportunity for joint engagement with these media. The so-called "digital divide" through which children are considered to be experts with digital media, while adults are positioned as novices, becomes a "resource for both children and adults to enter and sustain participation in activities" [14]. Children can teach parents about AI technologies, but it is also parents' responsibility to teach children about the values in their community that matter and how AI tools and systems align with these values [125].

4.4.1 Design Features that Encourage AI literacies for families

We can use our findings to examine the conditions and processes that our family AI literacies framework could support. We use our findings to show how the Ask, Adapt, Author, and Analyze dimensions can lead to a critical understanding of AI for families [89, 99] through a balanced engagement with these new technologies [287, 294, 342].

- **Mutual engagement** (i.e., multiple family members should be equally motivated to participate). Families in this study were able to participate in different ways, whether they were asking several questions to voice assistants, playing and authoring together with new AI systems, or trying to analyze how bias is introduced into smart technologies.

- **Dialogic inquiry** (i.e., inspiring collaboration and meaning-making): Families can try to analyze the AI system and figure out how it works. They can also determine how the AI systems need to adapt to their families' culture, rules, and background.

- **Co-creation** (i.e., through co-usage, people create shared understanding): Parents and children can come together to ask, adapt, author, and analyze AI systems in order to find out what they all currently know and what they would like to know more about.

- **Boundary crossing** (i.e., spans time and space): Families can consider how AI systems are pervasive in multiple technologies. Whether in Internet searches, YouTube recommendation systems, and voice assistants of multiple forms, recognizing how pervasive AI is becoming on many platforms can shape how AI is crossing boundaries.

- **Intention to develop** (i.e., gain experience and development): Families can consider how they are adapting to AI systems. For instance, are the questions they are asking voice assistants changing? Are families noticing when AI systems may be present? Interestingly, families can develop as they understand how AI systems adapt to different people and contexts.

- **Focus on content, not control** (i.e., the interface does not distract from interaction): With some AI systems, families can engage via multiple straightforward means of engagement. Through asking voice assistants questions, seeing if AI systems can recognize drawings and sketches, and engaging with computer vision models, many different and simple mechanics allow families to question and critique AI systems.

4.5 Conclusion

Our aim in designing is to ensure we support families in raising a generation of children who are not just passive consumers of AI technologies but active creators and shapers of its future. With our AI Literacies Framework, we aim to encourage and enable families to learn how to develop a critical understanding of AI.

We propose this framework from an ecological systems theory perspective and provide examples of implications for supporting family AI literacies across various nested layers of our society. As designers of technologies, we strive to support a diverse population of children and adults and provide significant inspiration and guidance for future designs of more inclusive human-machine interactions. We hope that democratizing access to AI literacies through tinkering and play will enable families to decide when and how they wish to invite AI into their homes and lives.

In the following chapter, I describe how the insights gained from our family co-design sessions enabled us to create a new series of AI literacy activities for families to use at home. These activities are designed to help families learn how to use AI technologies, comprehend the implications of these technologies, and develop a critical understanding of the algorithms that power them.

Chapter 5

AI LITERACIES AT HOME: HOW DO CHILDREN AND PARENTS LEARN ABOUT AI TOGETHER?

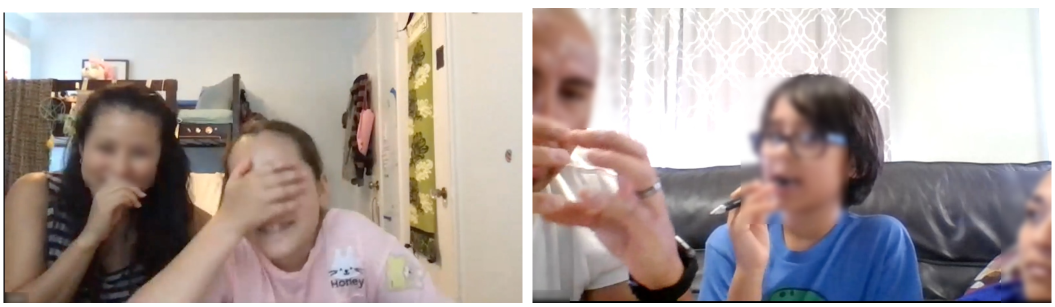

Figure 5.1: (left) A family laughs when they accidentally call Siri "Alexa" during the "AI Bingo Game". (right) A father suggests adding sign language support to their "Design AI" project.

The previous chapter showed that families successfully engage in the co-design of AI literacies. However, a question remains: what happens when children and parents learn about AI together at home? To answer this question, together with my collaborators, I used our prior co-design sessions with families [103] to create 11 learning activities for family AI literacy [1]. First, we invited 15 families from all over the USA to participate in a 5-week in-home study. Families spoke more than ten languages other than English and had a variety of backgrounds and levels of exposure to smart devices. Each family got to learn more about image classification, machine learning, and voice assistants, and they also got to design and analyze their own AI assistants [91].

[1] This study was done in collaboration with Fee Cristoph and Amy J. Ko and was published in CHI '22: ACM Conference on Computer-Human Interaction 2022 [92]

We found that our learning activities enabled families with different perceptions, attitudes, and knowledge about AI to engage in the following learning processes successfully: exploring multi-modal and embodied situated practices with AI, developing AI conceptual learning, engaging in critical framing of AI, and reflecting on future meaningful uses of AI at home. In addition, embodied and tangible activities best-supported families to engage in all these learning processes (in particular, the "Train AI" and the "AI BingoGame"). The study materials are available at aiplayground.me.

5.1 Study motivation and contributions

As discussed in Chapter 1, several initiatives provide AI educational resources for youth [304, 98, 199]. However, few resources currently exist to help parents mediate their children's in-home use of AI, despite growing parental concerns [282, 325, 3]. Furthermore, AI products such as voice assistants or smart mobile apps are only sometimes developed for youth, despite increasing usage [152]. These products raise additional inclusivity issues for families of different ethnicities, familial structures, technological literacies, and socioeconomic backgrounds [27].

Previous studies have described the benefits of families learning about technology together or engaging in co-design. For example, Barron et al. showed that parents could play various supporting roles, such as collaborator and learning broker [41]. More recently, Michelson et al. emphasized the importance of balanced partnerships in family technology co-design activities [212], and Yu et al. showed that parents primarily act as spectators, scaffolders, and teachers when supporting children's interactions with coding kits [344]. Although these studies highlight the importance of family engagement in children's technology learning, there still needs to be more knowledge about best practices for joint family AI learning at home.

To understand joint AI learning, we explore how families can best develop multiple *AI literacies* in the home. Our work builds on the notion of multiple literacies [63], which emphasizes how negotiating multiple linguistic and cultural differences is central to the lives of young people. By using the lens of multi-literacies, we aim to help families achieve two goals for AI learning: (1) creating access to the language of AI technologies and the power and community it can bring, and (2) fostering the critical engagement necessary to design social futures and meaningful use of AI in the home. For our purposes, *AI literacies* include the ability to read, work with, analyze, and author with AI [103, 106, 99]. Our framing of multiple AI literacies also borrows from Freire's assertion that literacy is about the acquisition of technical skills and the emancipation achieved through the literacy process [122].

Parents have experienced learning designers, routinely improvising learning experiences for their children. Suppose parents had a basic understanding of how AI works and valued applications of AI for their families. In that case, they could translate and explain AI terminology and concepts to their children and thereby guide meaningful adoption and use of this technology in the home, as was the case for video games [283].

To understand how families of different ethnicities, structures, technology exposure levels, and socioeconomic backgrounds interact with and learn about AI literacies, we pose the following research questions:

- RQ1: How do children and parents learn about AI together?
- RQ2: How can we design learning supports for family AI literacies?

We conducted a 5-week longitudinal study of 15 families with varying prior knowledge about technology and AI to answer these questions. We designed four learning sessions comprising 11 learning activities based on the four dimensions for multiple literacies, a framework proposed by the New London Group (NLG) [63] that we adapted to the field of AI learning for families by building on prior work [199, 103]. In the fifth session, we gathered feedback from families on the study learning activities.

We recorded and transcribed all study sessions to identify how family members supported each other in developing multiple AI literacies when engaging in our learning activities. Through thematic analysis of our codes, we identified eight parents' roles in supporting children's AI literacies practices. We then showed how our different activities supported parental roles in each session and proposed design recommendations for future family-centered AI literacies resources.

Our findings provide a roadmap for understanding family learning pathways to early AI literacies and contribute guidelines for supporting a constellation of family practices [255] and interests. Situating family AI literacies within the larger context of critical computational literacies [158, 175] and family as a third space for socio-critical literacy [142, 284], this paper highlights the benefits of partnerships between children and parents when reflecting on how to make use of AI for their family meaningfully. Finally, our study conceptualizes AI as a socio-material knowledge with social and societal histories and consequences.

5.2 Study procedure

Our study consisted of five sessions: (1) an image classification session, (2) an object recognition session, (3) a voice assistants session, (4) unplugged AI learning and co-design activities session, and (5) a reflection on study activities. The study took place online, and we used a free video conference application to connect with the families and guide them through the activities. In addition, detailed instruction playbooks, sent to each family one week before each study session, described the learning activity and provided links to tools, apps, or printed documents they needed to use during the activity (detailed descriptions of all study materials are included in the appendix).

5.2.1 Study Participants

We recruited 15 families for our study, consisting of 18 children and 16 parent participants. We posted an announcement on several family forums, social media groups, and Slack channels to recruit. Forty-four families applied to participate in the study. Our inclusion criteria for the study were to select families that were as diverse as possible along the following dimensions: family structure, ethnicity, geographical location, socio-economical background, children's ages, and gender. We selected 15 families. Of the 15 chosen, only 11 attended all the sessions. One family attended only one session, and three attended only two. The

Family ID	Parent(s)	Language(s)	Child(ren) and Age(s)	Joint Time
F1	Mom (S.), Dad (J.)	English, Spanish	Son, 7 (G.)	2 hrs, 57 mins
F2	Mom (C.)	English	Son, 9 (Et.) & Son, 9 (E.)	2 hrs, 49 mins
F3	Mom (D.)	English, Gujarati	Son, 11 (R.)	2 hrs, 34 mins
F4	Dad (E.)	English	Daughter, 10 (Sb.) & Daughter, 6 (Sm.)	3 hrs, 21 mins
F5	Mom (K.)	English	Daughter, 9 (L.)	1 hr, 5 mins
F6	Mom (T.)	English, Spanish	Daughter, 10 (H.)	2 hrs, 44 mins
F7	Mom (G.)	English, Chinese	Son, 7 (R.)	1 hr, 9 mins
F8	Mom (L.)	English	Son, 9 (E.)	2 hrs, 14 mins
F9	Mom (J.)	English, Spanish	Daughter, 10 (C.)	2 hrs, 7 mins
F10	Mom (I.)	English	Son, 10 (S.) & Daughter, 8 (K.)	0 hrs, 29 mins
F11	Mom (R.)	English	Son, 11 (A.)	2 hrs, 25 mins
F12	Mom (N.)	English, French	Daughter, 9 (C.)	3 hrs, 19 mins
F13	Dad (N.)	English, Hindi, Marathi	Daughter, 7 (M.)	2 hrs, 56 mins
F14	Dad (N.)	English, Hindi, Malayalam, Gujarati	Daughter, 8 (M.)	3 hrs, 5 mins
F15	Dad (A.)	English, Tagalog	Daughter, 5 (L.)	1 hr, 49 mins

Table 5.1: List of families that participated in the study

families unable to attend sessions cited extraordinary family circumstances as the reason or skipped sessions they deemed inappropriate for the young age of their child.

Children's ages ranged from 5 to 11 years old, with an average age of 8.5 years old. Ten children were female, and 8 were male. Of the 16 parents, 11 were female, and 5 were male. Of the 15 families, 5 were Asian American and Pacific Islander, 5 were White, 3 identified as multi-ethnic, and 2 were Hispanic or Latin. Families were located in 10 US states distributed evenly across the country. In terms of languages spoken, 10 families reported speaking languages other than English at home; these included 10 distinct languages and dialects, such as Spanish, Chinese, Hindi, Tagalog, Gujarati, and Malayalam. Regarding technology literacy, 6 parents had professional experience with technology design, 3 had programming experience, and the remaining 7 had no programming experience. In addition, families reported in-home use of a wide range of smart technologies: 15 used a computer and smartphone, 9 used a voice assistant, five used coding kits, and 4 had robots.

	Activity Name	Activity Description	MSP	ESP	ACL	CFA	DFMU
Session 1	Classification Game	Sort a set of 12 images of marine life into groups and name each group.	X		X		
	Anchor Game	Select the most important part of each image from a set of 12 marine life images.	X		X		
	Reflection	Reflect on how to use the image games to make something useful for society.				X	X
Session 2	Object Recognition	Identify home objects with an object recognition phone app.		X			
	Train AI	Train an interactive game to recognize different images and produce animations.		X	X		
	Prediction Game	Predict how the Train AI game would recognize specific edge case image examples.			X	X	
Session 3	Compare with Voice Assistant	Compare answers to specific questions between a voice assistant and a family member.		X		X	
	Draw What is Inside	Draw what is inside a voice assistant and how it works.	X		X		X
Session 4	AI Bingo Game	Complete a set of prompts by getting a voice assistant to say or do specific things.	X				
	Analyze AI	Analyze different characteristics of voice assistant along continuums (i.e., friendly to unfriendly).	X			X	
	Design AI	Design a custom AI device by selecting from a list of common AI toolkit features.	X				X

Table 5.2: Activities completed during the four sessions with corresponding AI literacies dimensions: Multimodal Situated Practice (MSP), Embodied Situated Practice (ESP), AI Conceptual Learning (ACL), Critical Framing of AI (CFA), Design Future Meaningful Use (DFMU).

All parents and children older than age 7 signed digital consent forms reviewed by an institutional review board agreeing to participate in our study explained to them by the first author of this paper. A list of family demographics is presented in Table 5.1.

5.2.2 Study Sessions

Session 1: Image classification. In this initial activity, families learned how to classify images of various marine objects ("Classification Game"). They then learned how to pick a representative segment of each image (anchor) such that an algorithm could only guess what the image was about by examining this smaller segment ("Anchor Game"). Both activities were conducted on a dedicated digital platform we designed and built. After these activities, families reflected on using them for good ("Reflection").

Session 2: Object recognition. Each family got to experiment with and learn about automatic object recognition in this activity. This session had 3 parts. The families (1) used a free smartphone app that recognized objects in their house and tried to tag them ("Object Recognition"), (2) trained their models for object recognition using a free public web app on their computers ("Train AI"), and (3) took a quiz that prompted them to guess what the computer model would predict for similar-looking objects ("Prediction Game").

Session 3: Voice assistants. For the third session, families engaged with voice assistants. This activity had 2 parts. (1) The families played a game with a voice assistant of their choice, comparing the assistant's answers with one of the family members' answers ("Compare with Voice Assistant"). If the families did not have a voice assistant, they were instructed to use Siri or download the Alexa mobile app. (2) The participants were asked to draw what is inside the voice assistant and how it works ("Draw What is Inside").

Session 4: Unplugged AI games and co-design. This last interactive session consisted of 3 parts. Family members (1) completed a set of prompts by getting their voice assistant to say or do specific things ("AI Bingo Game"), (2) compared humans, robots, and voice assistants on a printed scale that assessed dimensions of intelligence and socio-emotional attributes ("Analyze AI"), and (3) designed their smart assistant using different components from an AI toolkit we provided ("Design AI").

Session 5: Reflection on study and learning activities. In this final session, participants reflected on each activity. They were asked to describe how much fun they had doing the activity, how easy it was to do it, and how much they learned. We also asked for suggestions about improving the activity and describing what they liked the most. The first author then prompted the families to reflect on whether and how they would change their current uses of AI technologies and asked them to describe future AI learning activities they would like to use.

5.2.3 Data Collection and Analysis

Our study produced video recordings of all online sessions with individual families that participated in the study. A total of 35 hours of footage was collected from all sessions. The average duration for a family session was 33 minutes (see details of sessions duration for each family in Table 5.1).

For the qualitative analyses, the first and second author transcribed the videos and noted comments on children's body language and non-verbal interactions. The final corpus included 1,704 pages of transcripts (368,159 words). Once all transcriptions were finished, the first two authors each reviewed half of the data independently, separately analyzing each transcript using a combination of etic codes developed from our theoretical frameworks of joint-media engagement [291]. Parental scaffolding [111], and emic codes that emerged from the interviews themselves [216, 236]. In addition, we listed all joint-media and parental technology scaffolding practices that we found in prior studies of families interacting with home technologies, mobile tablets, or coding kits [41, 226, 343] and identified connections with a series of themes that emerged from our study.

After a final coding frame was developed, all transcripts were independently coded by the first two authors. To ensure the validity of the analysis, the two authors regularly met to discuss and reach an agreement on any newly emerging codes, any discrepancies in the analyses, and any refinement to the codes [172, 195]. Finally, the coding frame was changed, and the transcripts were reread according to the new structure. This process was used to develop categories, which were then conceptualized into broad themes after further discussion. Towards the end of the study, no new themes emerged, which suggested that all major themes had been identified [53].

Once the parental roles were identified, both authors looked at the transcripts for each activity with each family and marked roles as present or not present. Together with the second co-author, we discussed discrepancies until we reached an agreement. Each time a role was present for pairing a family and activity, we counted it as an *instance* of that role. We used the counted instances to address RQ2.

5.3 Findings: AI Literacies at Home

In this section, we summarize our perceptions of children's experiences and then discuss our results concerning RQ1 (how do children and parents learn together about AI) and RQ2 (how to design activities to support family AI literacies).

5.3.1 RQ1: How do children and parents learn about AI together?

We now turn to a more granular analysis of families' joint learning of AI literacies. Our qualitative analysis revealed a clear set of roles parents play when supporting their children's development of AI literacies. The way parents took on these roles for the different study activities varied. To illustrate this variation, we present examples of prominent parental roles for each study session.

What were parents' attitudes towards AI?

Our participant families reported varied use of technologies at home. All 15 of our families reported using computers and smartphones daily. Of these 15 families, 13 reported using mobile tablets, 11 reported using gaming devices, 9 used voice assistants, and 5 used coding kits.

Convenience. Some families enjoyed using smart devices in their homes, sometimes reporting having multiple voice assistants in different rooms (F4) or using voice assistants to control other connected appliances in their homes, such as smart lights (F11). However, some families were concerned about *privacy issues* with voice assistants or other AI technologies. For example, the father in F14 said he feels uncomfortable using Google Home, although they own the device. Parents echoed these privacy concerns in F3, F8, F9, and F11, with some parents recognizing that sometimes they do not know what information access they consented to when setting up their smart devices.

Control. Parents from families F1, F2, and F11 expressed the desire for more personalized answers from their voice assistants. However, they said they would like to control what information the voice assistants and other AI applications get access to:

> *"I would like an app where you can add personal information. It would be nice if they [AI devices] do not know unless you give them that information. Otherwise, it seems creepy"* — R., mom F11.

These findings are consistent with recent studies showing that parents often need to be made aware of the privacy settings of their smart devices [184, **?**] or smart toys [210]. Prior work has also found that parents would like to have more control of smart devices and decide what information they choose to disclose or not [21].

Quality. Many families recognized the utility of voice assistants in providing answers to factual questions (F1, F4, F9, F11, F12), and some described the voice assistants as knowledgeable (F1, F11, F4) and confident (F6).

Accuracy. While recognizing a voice assistant's abilities to answer factual questions, some of the parents (F13, F14) encouraged their children to recognize what assumptions the device is making before answering the questions, similar to parental roles observed by Beneteau et al. [46]:

> *"You assume [talking to his daughter] that the egg that we are talking about is from a chicken. Alexa had no such assumptions."* — N., dad F13.

Human element. In other cases, it was the children that would point out the device's limitations when it comes to answering questions that require human reasoning and opinions (F3):

> *"Nowadays, AI is supposed to have intelligence, but it does not think like a brain that can have opinions(...). Computers do not have opinions; they look at the facts."* — R., son F4.

Families sometimes perceived the voice assistants as "chatty"(mom F2) and not good at engaging in conversation (i.e., "I think we are more personal than Alexa," said mom F1). Parents' recognition of the voice assistants as not always fit for engaging in conversations led to them actively trying to scaffold the device's conversations with children, either by helping children reformulate their questions or by helping them make sense of the device's answers. This parental role is consistent with other studies that explored how parents mediate child interaction with voice assistants [46, 47, 100].

Transparency and Intelligence attribution. The level to which parents and children saw the voice assistants as knowledgeable and trustworthy was influenced by how smart they thought the devices were. We noticed that children and parents would influence each other regarding intelligence attribution to the voice assistants.

Inclusive design. Several of the multilingual families complained that voice assistants had trouble recognizing their voices or accents:

> *"Siri has trouble recognizing my voice, which annoys me."* — J., mom F9, who speaks Spanish as a first language.

Cultural relevance. As our study population comprises diverse families of ethnicity and spoken languages, several family members raised issues concerning the cultural relevance of some of the interactions with the smart devices. For example, C. (mom F2) complained that "some of her favorite songs are not there."

We identified nine concerns parents considered necessary when evaluating AI technologies at home: convenience, quality, accuracy, the human element, privacy, control of settings, transparency, intelligence attribution, inclusive design, and cultural relevance. In addition, we noticed that parents and children's initial concerns would determine if, when, and how they chose to engage with AI technologies at home.

These findings are consistent with a large scale pediatric study on parental attitudes towards AI medical support for their children's treatment which found that parental openness was positively associated with quality, convenience, and cost, as well as faith in technology and trust in health information systems [282]. Families with different perceptions and concerns towards AI could still find important, value-affirming discussion material in our study sessions. For example, F15's dad was against voice assistants and would use the interactions with AI to show his daughter their limitations. Meanwhile, F11's mom, who embraced smart devices in her home, would use the study sessions to geek out with her son about how excellent the assistants are.

How did families learn image classification together?

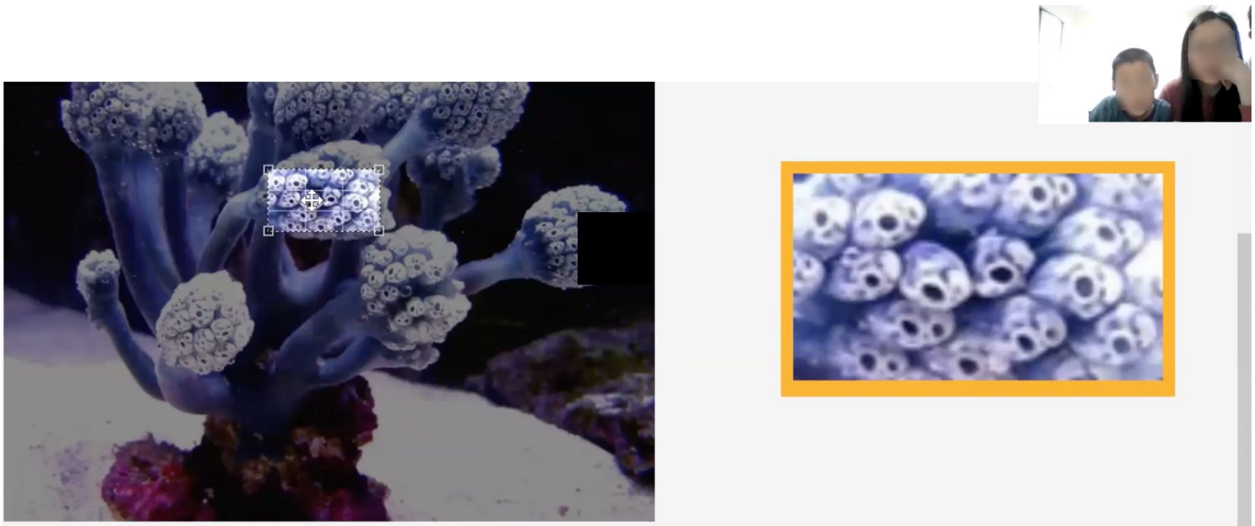

Figure 5.2: Example of the family engaging in the Anchor Game from the first session.

Fourteen families participated in this initial study session where they got to play two games classifying and summarizing various ocean images and then reflect on their process. Children primarily drove the activities during the image classification and image summarizing and created rules for categorizing the corals. Their categories ranged from grouping corals by color, size, or texture (i.e., "bumpy" vs. "sticky") to creating stories about the corals (i.e., "with fish" or "no fish"). Parents acted primarily as collaborators (31 instances), mentors (22 instances), mediators (17 instances), and teachers (17 instances) in this session (see Fig. 5.7a). There were also three families with older children where parents learned from their children's logic and image classification reasoning.

When acting as *collaborators*, parents would primarily support their children with scaffolding questions to help them identify unique features in the images. Parents would also try to support children's flexibility in changing their classification groups or image sections. The collaborative aspect of the family interaction in this activity was particularly useful in identifying and discussing various image classification and summarizing strategies.

The more difficult pictures had several different corals or showed a zoomed-in version of a coral. The images often caused children to pause and look to their parents for help. This happened in 12 of the 14 families that participated in this activity. Complex images also sometimes led families to consider renaming their image groupings or grouping images differently; however, renaming of groups only happened in 6 of the 14 families, as children were more reluctant to change their initial decisions. Sometimes the role of *collaborator* would shift into the role of *mentor* for parents, as they would prompt children to reflect on how a computer would make sense or be able to distinguish their examples.

Parents also played the critical role of *mediator*. This manifested when parents would help children understand the instructions or the activity's goal or help them recall the decisions they made in previous activities. In addition, if the family had multiple children participating in the study, the parents would help mediate the collaboration between the siblings.

Parents played the role of *teacher* in multiple ways throughout the 3 parts of his first session activity. During the image classification and anchoring games, parents taught children by providing cognitive or affective scaffolding [111]. For younger children, parents also provided support with domain knowledge (i.e., "what is a coral?") during the two games. During the reflection activity, parents acted as *teachers* by helping children link the current activity with other prior relevant experiences. Sometimes parents had to come up with elaborate stories and examples in order for children to understand how we could use applications of computer vision technologies in order to make something good for the planet:

> *" Maybe the computer can group it by where in the world it was taken. Kind of like if we go to SeaWorld. Then we take pictures; then people are going to be like, oh, where did you take this in SeaWorld?"* — J., dad F1.

Other parents (F4, F13) also prompted their children to think about algorithmic bias and consider what happens if the people who give examples of images to the computer make mistakes.

Parents also played the role of *student* in this activity. This either happened when children were older and had prior programming experience (this was present in 4 families participating in this first session) or when children would come up with scenarios for future AI applications that parents had not considered, such as involving scientists and experts in the process of crowd-sourcing image classification games.

> *"A computer would make mistakes because everything makes mistakes. Because computers, they are just people programming something new."* — L., daughter F8.

Both children and parents proposed various ideas when thinking about future potential applications for image classification and image anchor detection games. However, children were more likely to propose fun things, such as recognizing different types of dogs (F11) or recognizing children's drawings (F13). Some of the older children went much further in their reflections for future computer vision applications, imagining either how people could collaborate in the future with machines by playing games or imagining how computers could learn rules from the current image classification and image anchor detection games to program themselves:

> *"So when you make a program, you create some rules. So for the anchors, you could think of a rule that a computer could follow to know where to put the anchor [...] most likely where the most colors change."* — R., son F3.

How did families learn object recognition together?

Fourteen families participated in the second study session, which focused on object recognition. First, families looked for objects that would confuse a mobile recognition app ("Object Recognition" activity). Then, they trained and tested *Teachable Machine* application with three objects ("Train AI" activity). Finally, they predicted what the computer would choose when trained on two objects and tested with a different type of object ("Prediction Game"

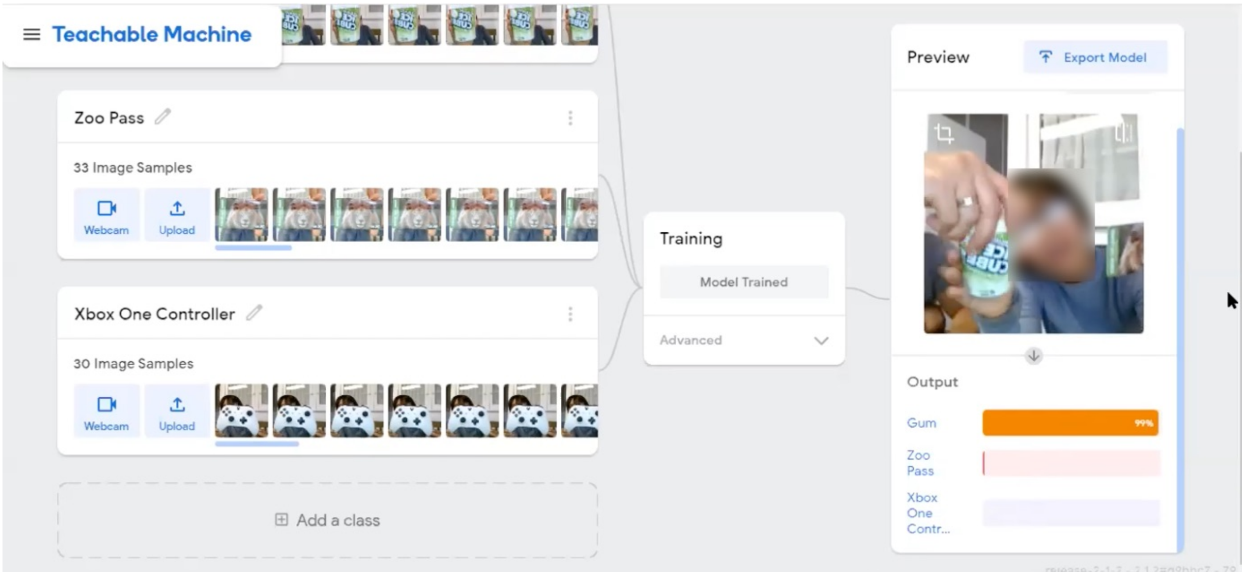

Figure 5.3: Example of a father using hand-on-hand scaffolding to help his son position the object correctly during the "Train AI" activity.

activity). Across all three activities in this session, parents acted primarily as collaborators (37 instances), mentors (30 instances), cheerleaders (25 instances), teachers (20 instances), and tinkerers (19 instances) (see Fig. 5.7b). There were also five instances of parents learning from their children either how to do object recognition, discover niche terms related to their children's interests, or learn about their children's past experiences with similar apps.

When acting as *collaborators*, parents would display their enthusiasm, actively make suggestions, and help children with the tasks. One source of enthusiasm from both children and parents was the act of "tricking the AI," first introduced in the object recognition app testing but carried into the *Train AI* activity by some families. Children and parents collaborated at two main points during the prediction activity: (1) when determining what the computer would predict, (2) when learning their initial prediction was incorrect. When the machine defied their expectations, family members tried to determine why their prediction did not work. In addition, parents and children sometimes collaborated to work through technical challenges:

> *"We should probably aim it at the ceiling, cause we have a bunch of pillows [in the background]."* — A., son F11, suggests how to fix the noisy background when training the AI.

In the "Train AI" activity, parents engaged as *mentors* when younger children would sometimes choose unusual objects to train their AI with (e.g., their pet), to which parents sometimes had to set ethical and safety boundaries (e.g., telling them they were not going to train it on their dog).

When acting as *teachers*, parents provided explanations for (1) what the object recognition application was doing, (2) what companies and other technologies supported object recognition, and (3) how the computer's behavior was similar to or different from the child's. When parents took on the *tinkerer* role, their interventions varied between the three activities. In the first activity, they would suggest different objects for the child to test with the object recognition app. Then, they would point to objects, pass the child objects, or suggest that a child looks for a certain kind of confusing object. In the "Train AI" activity, families got to "fix" some recognition issues because they trained the AI. Parents would suggest different edge cases for the child to test their AI with by picking different objects with similar shapes (F14), picking objects of the same color (F15) or rotating initial objects (F1) (see Fig. 5.3).

Though the number of instances of parents taking on the *student* role was low (only five instances), some children taught their parents how to use the *Teachable Machine* platform (daughter F12), while others taught them specific terms or gave them new insights into their previous experiences with object recognition applications (son F11).

How did families mediate learning with voice assistants?

Twelve families participated in the third study session, where they engaged in two activities related to voice assistants. During the "Compare with Voice Assistant", children or parents answered the game questions. Families chose different assistants to compare themselves to

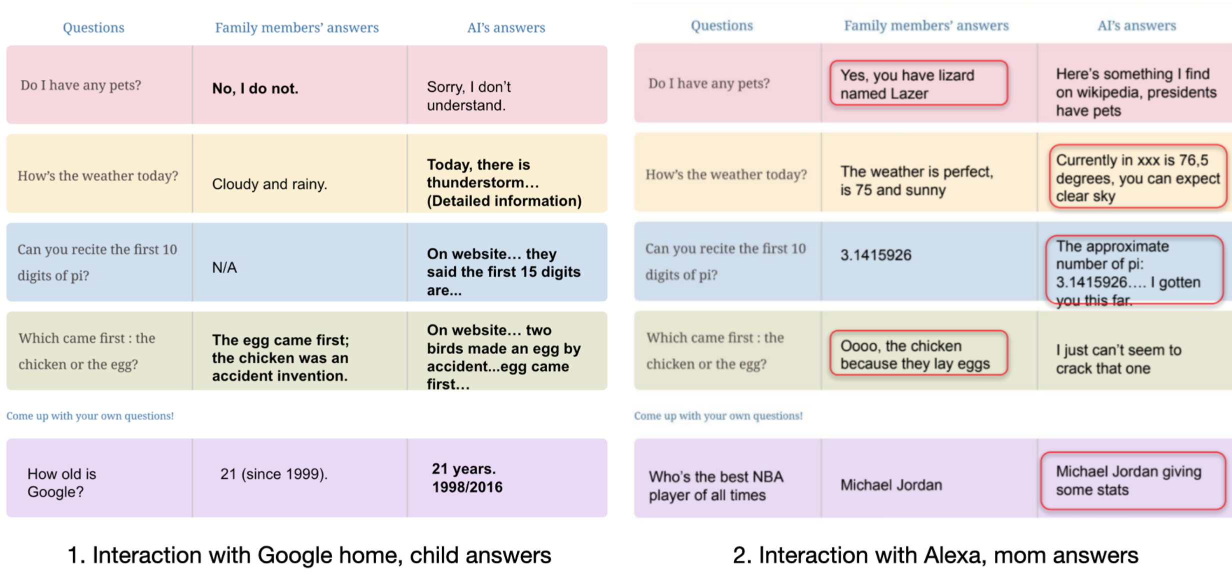

Figure 5.4: Examples of families' answers to the activity "Compare with Voice Assistant" from session 3.

(see fig 5.4). If the families did not have a home voice assistant, they used Siri or the Alexa app. In the first activity parents acted primarily as *collaborators* (11 instances) and as *mentors* (11 instances). In the second part of this third session, for the "Draw what is inside the assistant" activity, parents acted primarily as *mediators* (6 instances), *teachers* (5 instances) and *mentors* (4 instances) with only two parents (F4, F11) making a drawing. The cumulative count of parent roles showed that they acted primarily as mentors and as collaborators (15 instances for each), teachers (12 instances), mediators (11 instances), cheerleader (7 instances), student (7 instances), observer (6 instances) (see Fig. 5.7-session 2). In some capacity, all parents acted as *collaborators* in this third session. Eleven parents played with their children in the "Compare with Voice Assistant" activity by responding to or asking questions.

In the first part of session three, parents and children collaborated to develop new questions to ask the voice assistant. For example, when family members wanted to give an advantage to each other in the game against the voice assistant, they would ask personal

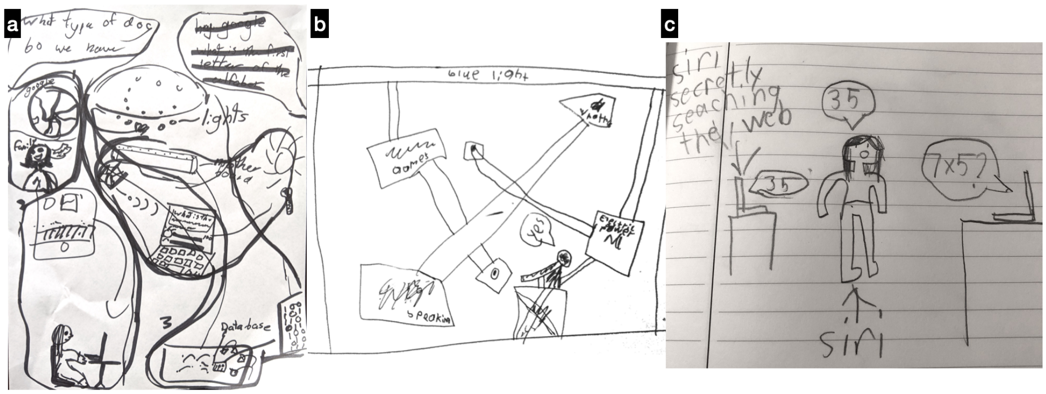

Figure 5.5: Examples of children's drawings from the "Draw What is Inside" activity: a.) child F4 drew Alexa as a girl typing and connecting to databases, lights, Google, b.) child F8 drew Alexa as parts of the phone's circuitry, c.) child F14 drew Siri as a girl searching the web and telling the answer to a computer.

questions such as "what is my favorite color ?" (F8), "who is your favorite ballerina?"(F12) or "what is the most fun activity you do?"(F13). Other times family members would inquire for facts related to their interest (i.e. "who is the best NBA player of all times?"(F2), "why does the T-rex have tiny hands?"(F14)) or ask about trivia facts (i.e. "what is the black hole in the middle of the Milky Way?"(F3), "when was memory foam invented?"(F1).

Parents primarily acted as *mentors* in the first part of this session when they guided their children to reflect on what makes a human answer better than the voice assistant's answers. During the drawing activity, parents also acted as mentors by prompting their children to think of specific examples or situations to help them plan their drawings. When mentoring, parents also encouraged children to explain their AI understanding in more detail by asking clarifying questions:

> *M:"Mmm, maybe the programmer could translate human into robots."* — M., daughter F14.
> *N:"I see. So it needs to have something that converts voice into words?[daughter nods] (...)* — N., dad F14, responding.

The above dialogue with her dad leads M.(F14) to draw her assistant Siri as a girl who "secretly" searches the web to answer the questions. It then refers back to the "other computer" that presents the person asking with an answer via voice (see Fig. 5.5c). M.'s drawing of Siri was very similar to S.'s drawing (F4), who drew Alexa as a girl typing and connecting to databases, lights, Google (e.g., Fig. 5.5a). Other children and parents used various metaphors to describe their vision for what is inside the voice assistant, such as drawing different parts of the phone's circuitry (e.g., Fig. 5.5b).

When acting as *teachers*, parents either explained specific domain knowledge concepts (i.e. "what is pi?") or directly explained to their children how certain functionalities of the voice assistants work. Parents also played the role of *student* and learned from their children knowledge and ways of reasoning about how the voice assistants work and how their children would compare different voice assistants:

> *"If Alexa was smart enough, she could have seen (...) we don't order any of the pet products, which probably means that we don't have pets."* — R., son F3 talking to his mom.

Examples of discussions on sensitive topics, such as race and religion, between children and voice assistants, lead parents (F2, F4, F6, F12, F13, F14, F15) to recognize that these devices are not always neutral [77, 128, 204, **?**] and that it is critical for families to have conversations about when to trust the voice assistant's answers. Some families (F1, F2, F4, F12) emphasized the importance of differentiating what questions are best suited to ask family members and which ones are best to address to a voice assistant:

> *"'Do we have a dog' would be a question for the family, the pi question would be for assistants [dad asks how do you differentiate] for family-related questions we would ask the family."* — Sa.,daughter F4.

When trying to find future meaningful applications for voice assistants and AI, families proposed a series of ideas: support with family learning by "having better support for homework"(son F2) or enabling more convenient image search (dad F14).

How did families co-design future AIs?

Twelve families participated in the fourth session. Across the activities for session four, the "AI Bingo Game" was most engaging, and the "Design AI" activity was most collaborative. Engagement and enjoyment for the bingo game varied and depended heavily on the quality of the voice assistant's responses, which sometimes were funny and appropriate. However, other times were unrelated or not engaging. Engagement dropped off when families were subject to a succession of interactions where the voice assistant needed help to provide answers or understand participants.

The third "Design AI" activity prompted active discussions around privacy and AI ethics. Family members shared their previous experiences and collaborated to understand how features and hardware/software components connected and how they could build safeguards into their designs. Parents were not always more privacy-minded than children but often could explain to children which settings on their AI assistant led to certain behaviors, like the assistant knowing their home address.

The most common roles observed in the fourth session's activities were collaborator (33 instances), mentor (32 instances), teacher (26 instances), cheerleader (25 instances), and tinkerer (18 instances).

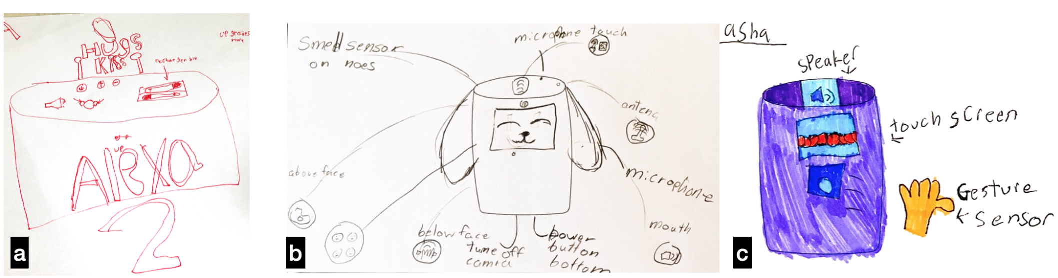

Figure 5.6: Examples of kids' and parents' drawings from the "Design AI" activity: a.) child F13 designed a new portable/rechargeable Alexa with a hug and kiss kit, b) older child F4 designed an animal-like assistant with buttons to control all privacy features and a sensor to detect smell, c) child F12 designed "Asha" to detect gestures and touch input, allowing for non-verbal commands.

As *collaborators*, parents engaged in back-and-forth conversation with their children and gave suggestions relevant to the activity. For example, in the first activity, the bingo game, the families' collaboration involved asking the voice assistant different questions and suggesting ways to accomplish a task. Active collaboration sometimes meant family members would build off each other's voice assistant interactions, as a group trying to narrow in on a specific query that would get the desired response, such as "make AI tell a lie" (dad F4).

In the second "Analyze AI" activity, collaboration often took the form of parents and children sharing their views of the AI and agreeing on how to rate the AI's different characteristics. In addition, they often drew on their previous experiences with AI when giving justifications.

The third "Design AI" activity, where parents and children co-designed their ideal assistant, had the most engaged and personal collaboration. When deciding which features and behaviors to include in their AI, parents would offer suggestions, sometimes rebuffed by children who thought their suggestions would create an AI that was "too creepy". Often, collaborations involved discussions of privacy concerns around AI and potential safeguards. Parents scaffolded ethical conversations by offering help on how to design against a specific concern:

> *"What if it was like a face that looked more like a robot face? Would that still be creepy? [C. nods]"* — N., mom F12, suggesting potential modifications to their AI design.

Sometimes, children wanted more safeguards than parents, like in family F6, where the daughter wanted no biometrics information recorded, but the mother was ok with using those sensors. However, in these collaborations, children often made fun of the AI and had lower technology expectations. In one case, the son of family F1 even made fun of Alexa's accent for pronouncing "La Cucaracha" without a Spanish accent.

During all three of session four's activities, parents often took on the role of *mentor*. For example, during the "AI Bingo Game", parents primarily helped repair communication breakdowns with the assistant (asking children to repeat their query, slow down, or enunciate), operate the assistant, and phrase or rephrase queries that the child wanted to pose. For the "Design AI" activity, parents scaffolded conversations around ethics and helped children connect certain behaviors they wanted their AIs to have to the required sensors for these behaviors. In some instances, they would nudge their children to consider designing the AI for others or encourage them to think beyond the affordances of the AIs they already know.

When parents acted as *teachers*, they taught their children various topics, ranging from simple definitions of words to detailed explanations regarding the people and programming that make voice assistants possible. Similarly, they gave detailed explanations about the distinctions between (1) the people vs. a company that builds an AI, (2) lying vs. not knowing something, and (3) common vs. uncommon AI queries and the expected behaviors for common queries. In the "Analyze AI" activity, parents continued these explanations and tied them to characteristics of the AI, like friendliness, truthfulness, and agency.

In the "Design AI" activity, discussions around privacy and ethics led parents to teach children about current concerns around AI and specific design patterns that could mitigate against them:

> *"You can make a password for her. You can say "flower" and then maybe she'll obey."* — M., daughter F13, adding a password to her AI.
> *"But then it's the same thing as 'Alexa,' right? When you want to ask about flowers, what do you do?"* — N., dad F13, highlighting potential shortcomings.

Parents supported their children as *cheerleaders* during the three activities by expressing excitement for the activities, consoling children when the voice assistant did not understand them, and supporting children's creativity.

5.3.2 RQ2: How can we design learning supports for family AI literacies?

In this section, we consider how our AI literacies resources supported various parental roles for each activity and present families' final evaluations of each study session.

Support for parental roles

We counted instances of each parental role identified in RQ1 by marking whether or not a role was present for each pairing of a family and an activity. Thus, there were a total of 142 possible instances for each existing pairing (three activities and 14 families in session one, three activities and 14 families in session two, two activities and 11 families in session three, and three activities and 12 families in session four, see Fig. 5.7).

For the first session, the cumulative count of parent roles showed that parents acted primarily as collaborators (31 instances), followed by mentor (22 counts), then mediator and teacher (both 17 counts) (see Fig. 5.7-session 1). The second session had the same top two roles. Parents again acted primarily as collaborators (37 instances), followed by mentor (30 instances), and then cheerleader (25 instances), and teacher (20 instances) (see Fig. 5.7-session 2). In the third session, mentor and collaborator tied for the most common role (15 instances), followed by teacher (12 instances) and mediator (11 instances) (see Fig. 5.7-session 3). During the fourth session, parents acted primarily as collaborators (33 instances), mentors (32 instances), teachers (26 instances), and cheerleaders (25 instances) (see Fig. 5.7-session 4). Roles that were not in these top roles all appeared most in the fourth session: tinkerer (18 instances), student, and observer (both 14 instances) (see Fig. 5.7-session 4). The two activities that had the most joint engagement, found by summing the instances of *collaborator* and *tinkerer* were "Train AI" (23 instances of joint engagement roles) and the "AI Bingo Game" (22 instances of joint engagement roles).

Sessions feedback

The 11 families that provided feedback for the study sessions described session one on image classification as relatively easy but expressed varied opinions on fun and learning activity levels. Overall, families described session two as more fun than session one (except for F15, who had a very young child). Overall, families reported learning less but having more fun in session two compared to session one. Finally, families scored session three interaction

Figure 5.7: Radar charts presenting the distribution of parents' roles for the different study sessions and AI literacies activities.

with voice assistants with relatively high scores across learning, having fun, and ease of use. They scored it slightly less fun than session two but said they learned more. Because the final session consisted of many unplugged activities, most families described this session's activities as relatively easy to play. However, the scores assigned for fun and opportunities for learning varied more from family to family.

What did families like the most? For the image classification session, all families expressed that they appreciated the interactive nature of the activity and the ability to pick the games' pictures. Several families reported they enjoyed testing, breaking, and tricking the object recognition applications and the voice assistants. Some families (F2, F6, F13) mentioned they liked the "Compare with Voice Assistant" competition aspect. From session four, families said their favorite activity was the "Design AI".

What improvements did families suggest? Families suggested expanding the games collection of images to include images from Minecraft (F1), animal pictures (F8), cities and ponds (F2), and "other crazy parts of the ocean"(F11). Families also suggested that the game should be online and collaborative (F3) and that the game should suggest more questions or explanations about the pictures (F13). Finally, when referring to the "Compare with Voice Assistant" activity, some families (F6, F2, F11) suggested creating more activities where family members could interact with multiple voice assistants and compare their answers to different questions. For the "Design AI" activity, family F3 suggested ways to bring the design to life virtually, and family F14 suggested that it would be fun to design their AI toolkit parts.

5.4 Discussion

Our work contributes several new insights about AI literacies for families by addressing our initial research questions:

RQ1: How do children and parents learn about AI together? Our qualitative results show that parents mediate children's learning by playing different roles ranging from *Mentor* to *Student*. However, we observed balanced learning partnerships between family members, primarily when parents play the *Collaborator* and the *Tinkerer* roles. Furthermore, while children and parents collaborate in all our different AI literacies sessions, they primarily tinkered together in the sessions that support hands-on interactive games (session two) and unplugged learning activities (session four) (see Fig.5.7). While some of the roles we identify are similar to parent roles present in other family technology learning activities [41, 344], the *Tinkerer* and *Student* roles we found are unique to AI learning activities. As sometimes parents and children in our study differed in their experiences, opinions, interpretations, and imagined futures of AI behavior, the home became a transformative third space [141] for AI literacies where the potential for an expanded form of learning [109] and the development of new knowledge was heightened.

RQ2: How can we design learning supports for family AI literacies? We found that our designs of supports for AI literacies let families with different perceptions, attitudes, and knowledge about AI engage in the following learning processes successfully: exploring multi-modal and embodied situated practices with AI, developing AI conceptual learning, engaging in critical framing of AI, and reflecting on future meaningful uses of AI at home. Activities in sessions two and four best-supported families to engage in all these learning processes (in particular, the "Train AI" and the "AI Bingo Game"). Activities in session one best-supported AI conceptual learning and critical framing (in particular in the *Reflection* activity). Activities in session three primarily supported AI conceptual learning (in the "Draw AI" activity) and reflections on future meaningful use of voices assistants for families (in the "Compare with Voice Assistant" activity). By designing activities that allowed families to move in and across a repertoire of practices [255, 144], we supported multiple forms of participation [143, 220] and created the potential for authentic interactions and expansive learning [109].

Our results suggest that engaging families in joint AI literacy practices can lead families to envision new ways to learn about these technologies. Moreover, introducing families to the novelty of AI concepts and applications and the hidden potential risks of using these technologies enabled parents and children to envision sites of possibility [220] and contradiction with their individual and joint dispositions and repertoire of practices. Notably, newly acquired practices and skills led some families to consider making meaningful use of AI devices in their homes and re-design their interactions with them. These findings suggest that family has the potential to act as a third space for learning, where both children and parents can develop AI literacies by combining family social contexts for learning and their collective zone of proximal development [323].

Limitations. One important limitation of our study is that half of the parents had some professional technology experience (six parents had user-experience design backgrounds, and three had programming experience). Some limitations in the study complicate the interpretation of our findings. First, it was impossible to systematically observe every family interaction in every activity, especially with the study's limitations online. Second, for the interactions we could observe, observing a family interact during a study does not necessarily indicate ground truth for their typical interactions outside of the study setting; for example, it may be the case that parents were playing a less active role in some sessions because they considered their children's opinions to be more relevant to the study. Third, some families did not participate in all four sessions, nor did our sites cover the many possible ways that culture, community, and collaboration shaped participation. Finally, because our observations were collected during study sessions and with a subset of each family, they may only hold a subset of the interactions the family regularly uses when engaging with AI. For example, our data do not include interactions involving grandparents or younger siblings or when the family engages with their voice assistant during a mealtime conversation. Therefore, while our results suggest that the families in our sessions demonstrated diverse roles and perceptions, other populations could reveal new roles and different shifts in perceptions.

Parents' and children's roles. By using niche cultural references, speaking in different languages, or finding examples of confusing images, families used all the resources to solve a given AI activity. Children and parents would build on responses they elicited from the agents to identify increasingly narrow edge cases. We interpreted this to be similar to practices observed in studies on AI understanding with the use of counterfactual examples [**?**, 24]. As families learned new tricks, they used them in different activities (i.e., the practice of "tricking the AI" continued from session to session). Similar to other examples of playful debugging [185], parents and children took great pride in finding a case that would confuse or mislead the AI device or application and would share their discovery with their family members. The *Tinkerer* and *Collaborator* roles facilitated joint engagement between parents and children. Parents took on *Mentor*, *Mediator*, and *Cheerleader* roles to keep their children engaged with the activities. Parents as *Mentors* provided scaffolding for children to understand the activities and connect the activities to their understanding of AI. *Teacher* and *Student* roles allowed parents and children to learn from one another, while the *Observer* role allowed parents to discover their child's habits more passively. Parental collaboration, mentoring, mediation, and emotional support have been found in prior studies on family use of technology [41, 83, 56] and studies on families engaging with coding kits [344], or video-games [227], however, the *Tinkerer* and *Student* roles we identified in this study appear to be unique to family interactions with AI.

As parents and children learn together to negotiate and reclaim agency from the smart devices by breaking, fixing, and testing them when they tinker [16, 35], we see opportunities to design family AI devices and applications that are more explicit about their functionality and abilities [15, 110, 251]. Prior work shows that youth can influence their parents' digital media use [72] and suggests the importance of parent and peer contexts for children's moral reasoning development [326]. Our study also found that as parents are still unfamiliar with some aspects of AI literacies, children step in and share their knowledge and perspectives [316, 192, 100, 315]. However, parental guidance and scaffolding are still necessary when reasoning about the ethics of AI [238, 239] and algorithmic bias [103, **?**].

Embodiment and technologies' maturity impact level of engagement. We found that the learning activities that supported embodiment provided rich environments for children and parents to build up egocentric speculations, extrapolating from their ideas about performing a task or solving a problem to the AI's behavior. This is consistent with Papert's findings on body synchronicity, where children project robot geometrical puzzles on their own body to solve mathematics problems in Logo [234] and with Vartiainen et al. who found that children reason about the relationship between their bodily expressions and the output of an interactive image prediction tool [316].

Additionally, we found that training an AI model allowed families to test hypotheses and even break the AI because they could fix it. When families had the opportunity to train the AI, they could build a more accurate picture of the AI's behavior and capabilities. This finding is consistent with prior work, which shows that learning how to train smart games to support children to understand better machine intelligence [94].

Importantly, we found that when breaking and fixing the AI, families must be provided with conceptual and technical support to help them determine the cause of the AI's erratic behavior (e.g., hardware limitations, noisy data, limited bandwidth) so they have the opportunity to fix it and refine their understanding. Furthermore, when families encounter technical difficulties, it is challenging to debug and engage in interactive learning activities. This finding suggests the need for more mature AI applications and technologies that are well-tested with families [48, 244].

Perceived utility impacts family use and mediation. How parents choose to regulate their use of specific technologies is colored by perceived utility, which in turn results from how well they understand the technology and can support what their kids do with it [55]. Joint engagement with AI allows parents to do both at the same time. They gain insight into their children's habits with these smart agents, learn more about the capabilities and limitations of the agents, and have the chance to engage in active mediation [298]. Our observations of family AI perceptions expressed in our study were similar to Brito et al., who found that families assign meaning and intelligence to smart technologies before using them and that this process influences the decision to adopt them [56]. Especially in session four, families who had already adopted voice assistants had more accurate or fun responses from the assistants and were, therefore, more engaged in the activity.

Joint-Media Engagement for AI literacies. Our results also have implications for prior work on children developing AI literacies. Prior work has revealed many challenges, including the importance of family members understanding the role of data in shaping machine behavior [217]. Other studies with adults have explored methods of bridging these comprehension gaps by helping people develop more robust mental models about AI (e.g., [180, 37, 265]). Our findings suggest that similar approaches may work for families, at least when families are engaged in interactive learning activities that use AI applications. Our qualitative findings of joint engagement of families' AI literacies also suggest new interpretations of prior research on child AI education. Whereas prior work has largely focused on children's experiential and cognitive accounts of AI understanding (e.g., how children make sense of machine intelligence or learn how machine learning works [94, 208]), our investigation of AI literacies from a joint-media engagement lens [291] suggests that children and parents support each other in significant ways to understand AI behavior. These supports include social strategies for enacting scientific activities such as observation with family members, discussing hypotheses with family members, and explaining and teaching other specific domain or task-specific concepts for inferring models of AI behavior.

Guidelines for designers and educators. Our findings have implications for both designers of learning technologies and AI literacy resources for families. The embodied interactive activities in session two and the unplugged activities in session four were the ones that supported the most diverse set of parental roles and therefore resulted in families learning about all the different AI literacies. This trend is consistent with recent studies analyzing families co-designing interactive AI museum exhibits [199] and research on families engaging in creative coding activities [257]. Designers and educators might therefore consider methods for supporting more embodied and tangible supports for future AI learning [196, 239]. Another clear trend was that families used their experiences in generating training data to make inferences about AI abilities. Designers and teachers might explore methods for engaging families in reflecting on the relationship between the training data, the AI's use of that data, and its resulting behavior. As our study population included a multilingual and multi-ethnic group of participants, we found it was important to design reflection activities that allowed families to approach AI literacies through the lens of culture and power [322] and provided families with opportunities to envision and imagine meaningful future AI designs. Designers and teachers might explore ways for critical reflections and AI speculative designs that leverage a families' culture, lived experiences and dreams, and diverse constellations of practices [255, 143].

5.5 Conclusion

After a 5-week observational study in the home, we found that families with different perceptions, attitudes, and knowledge about AI can successfully develop AI literacies in various joint-engagement roles. By increasing childrens' and parents' AI literacies, we would allow them to use smart technologies and imagine, meaningfully design, and create future AI applications relevant to their lived experiences and community needs. This vision *must* be attained if our children and their families are to live in a just and equitable society.

Chapter 6

COGNIMATES: CODING AI GAMES WITH FRIENDS

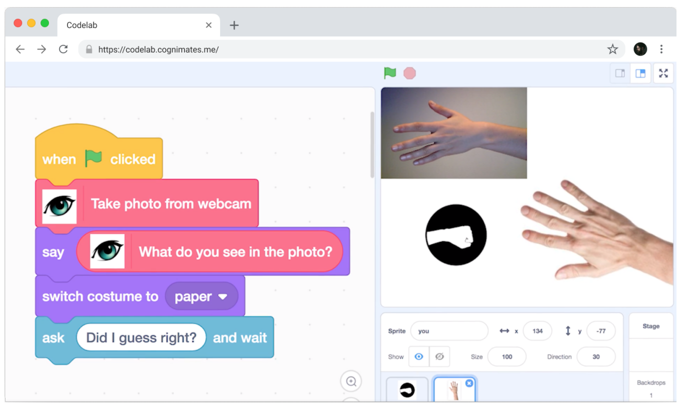

Figure 6.1: Cognimates platform preview of Rock Paper Scissors Game.

This study[1] explores how joint peer engagement in coding AI games enables children to discover the core concepts of image and text classification and foster critical reflection on the uses of AI by refining their sense-making hypotheses when testing their classification models and smart games. It builds on the findings from chapter 3, highlighting the need to design learning activities responsive to learners' prior knowledge. I designed a study where children can train models with their data and examples. It also builds on findings from chapters 4 and 5 that show that embodied and interactive learning activities are most conducive to children's extended engagement. In this study, groups of children program collaboratively interactive smart games that allow them to train and test custom classification models for detecting body gestures or text messages.

[1]This study was done in collaboration with Amy J. Ko and was published in IDC '21: ACM Conference on Interaction Design for Children 2021 [93].

A few initiatives today aim to introduce children to AI, such as programming using pre-trained models [160], incorporating AI classifiers into athletic practice [350], or exploring how object recognition works by building custom prediction models [208]. In addition, linking statistical inference to personal data has been highlighted to help children understand how data is used in AI [205, 78]. This is based on the idea that familiarizing children with data's origins and meaning will make it easier for them to comprehend AI [145, 178, 68, 247].

Coding and programming apps have also been found to help children develop their computational thinking skills, including AI literacies concepts [232, 289, 331]. This can enable them to reason and communicate in a digital world. Furthermore, students are coding to learn to code and create games, stories, and animations to share [159]. This has given rise to programming communities, and challenges regarding a more critical computer science education [159, 176]. Moreover, students are programming more than just stationary screens—they are programming toys, tools, and textiles.

A systematic review has established that AI studies benefit children cognitively, intellectually, and socially [171]. In addition, it recommends using project-based learning in group projects, which can assist in developing critical thinking, problem-solving, and cooperation skills [201].

This study builds on this prior work by allowing groups of children to train their prediction models with meaningful examples of text or images. They can then use these custom models in a familiar visual coding environment to program their smart programs. I wanted to explore whether youth would change their perception of machine intelligence in the process [94]. I discovered that children would use the scientific method collaboratively while training, coding, and testing their smart programs. I also observed that children became more skeptical of the specific abilities of smart devices as they shifted their attribution of agency from the devices to the people who program them. These shifts in perception happened through individual interactions with agents and debates with peers.

6.1 Study motivation and contributions

Many children are spending more time engaging with artificial intelligence. This engagement with what we will call *smart agents* is likely to increase, as there is significant growth in smart toys and more than 50% of North American households alone are expected to have a dedicated voice-assistant by 2022 [290].

Researchers have begun to examine children's experiences with these agents. For example, smart toys might influence children's perception and attribution of intelligence, their moral choices, or their behavior through play [100, 333, 320]. Prior work has shown that children see agents as friendly and truthful, and older children (7-10 years old) especially consider the agents to be more intelligent than themselves [100]. However, what may seem initially to be a playful interaction between a child and the smart agents can trigger events of significant consequence, such as children being spied on after their connected toys were hacked [228]. Many of these devices have proven to be easy to compromise [313, 329, 26], and some companies designing these technologies utilize questionable practices [203, 241].

The unequal access to smart agents in the home also amplifies digital divides, with only some children learning to make sense of how smart toys and devices function [43, 85]. Prior work has demonstrated that parental attitudes, socio-economic status, and cultural differences play a significant role in how children attribute agency, intelligence, and socio-emotional traits to the agents [101, 98]. Other studies have shown that children often misunderstand agents and tend to overestimate their abilities, either because children do not understand how these agents work, or because artifacts like toys and phones can talk, express emotions, and engage with youth in ways other humans would: with persuasive and charismatic modes of engagement [337, 115, 243]. In this context, we recognize the need for inclusive Artificial Intelligence (AI) literacy efforts to prepare a generation of children growing up with AI. We define AI literacies as the ability to critically decide if, when, and how to use smart devices.

Explorations of AI literacies applications in education are challenging since the mechanisms and opportunities of AI are unfamiliar to most people outside computer science. AI literacies is also considered a vital part of computational thinking [304, 80] and there are arguments to include AI literacies as part of the CS curricula in K12 level [208, 238, 98]. In parallel, several studies explored how youth can learn more about AI by interacting with pre-trained models [316, 160] or training their models [350, 208]. Vartiainen et. al found that young children (3-9 years old) reason about the relationship between their bodily expressions and the output of an interactive image prediction tool and engage in an emergent process of teaching and learning from the machine [316]. Kahn et al. found that high schoolers in developing countries enjoy to created block-based programs using pre-trained AI models but do not always understand how these models work [160]. Zimmermann et al. showed that youth with no programming experience can incorporate AI classifiers into athletic practice by building models of their physical activity on a mobile app [350]. However, none of these prior studies explored how children changed their perception of AI abilities after engaging in AI programming and training activities.

In this study, we plan to address this gap by focusing on one specific aspect of AI literacies: when learning to program smart agents, how do children's perceptions of smart agents' intelligence change? Britto et al. observed that families are assigning meaning and intelligence to smart technologies even before starting to use them and this process bears weight on the decision to adopt them or not [56]. Turkle notes how smart toys in particular have changed the way children evaluate the "liveliness" of a machine. Rather than assessing machines solely based on intelligence, children have now begun to also inquire whether their smart toys can feel and convey emotions [310, 308].

Prior work on general programming suggests some possible changes. For example, Scaife and Duuren found evidence that the "programmability" of technology can shift children's theories of intelligence about computers away from the device and toward the programmers of the device [271, 105]. While these studies were investigating traditional programming, in our study we investigate if similar phenomena can be observed when children are using AI-based training and programming.

We focus our study on two research questions:

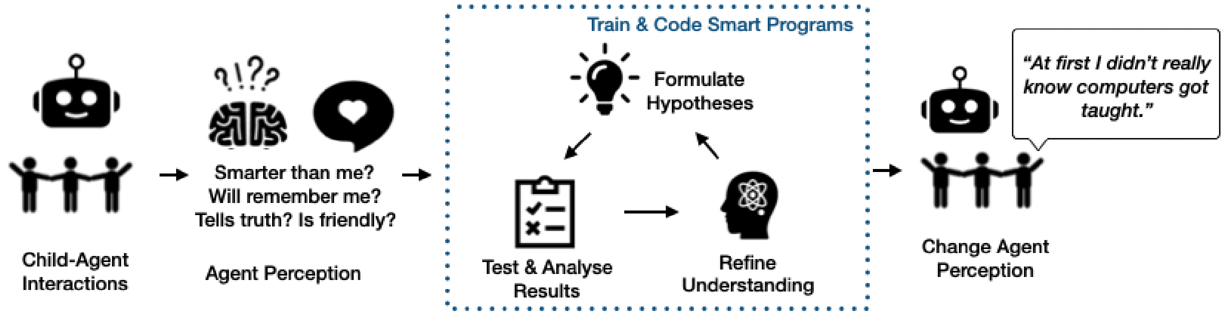

Figure 6.2: Cognimates study findings summary: Children (7 to 12 years old) engage in the scientific method when training & coding smart programs and become more skeptical of certain abilities of smart devices.

- RQ1: How do children make sense of machine intelligence when training and coding smart programs?
- RQ2: How do children's perception of machine intelligence change before and after building smart programs?

To answer these questions, we ran a 4-week study in both public and private after-school programs and community centers with 52 children (7 to 12 years old), observing children's sense-making and measuring their shifts in machine intelligence perception (see Fig. 6.2). Our investigation makes three contributions to the understanding of AI literacies in children. First, we provide empirical evidence of how children engage in sense-making practices when training and coding smart programs. Second, we present how children's perceptions of machine intelligence change after participating in the study. Finally, we discuss how the theoretical model of sense-making is relevant to developing AI literacies in children.

6.2 Study procedure

To understand how children make sense of machine-intelligence when training and coding smart programs and how their perception of machine intelligence changes, we structured our study in the following order: perception game, observations of children in 3 learning activities, perception game, analyze pre/post perception game responses and observations to understand changes in children's perception of machine-intelligence. Fig. 6.3 overviews the study design.

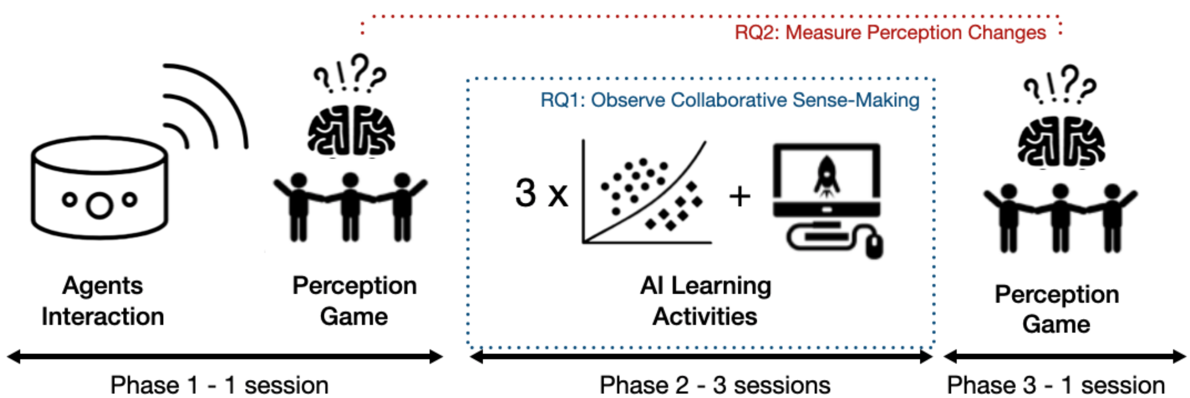

Figure 6.3: Study Overview

6.2.1 Study participants

We specifically chose many different locations for our study workshops to include a diverse population of students. The workshops took place in the following locations in Massachusetts, USA: an after-school program in a Spanish-English bilingual public school with mostly children of immigrant families (Public After-School Program), a non-profit community the center housed in a former church (Church Community Center); an after-school program in a private school in Cambridge (Private After-School Program), a private STEM center in Lexington (STEAM Community Center). In total, we had 52 children of ages 7-12 years old, with 16 girls and 36 boys, 28 younger children (7-9 years old) and 24 older children (10-12 years old).

6.2.2 Study Sessions

Our study comprised of three sessions: 1) initial encounters with agents and perception game (pre), 2) programming and training AI and 3) perception game (post).

Session 1: Initial Encounters. The goal of our first phase was to introduce children to smart agents and programming, while establishing a baseline of children's perceptions of machine intelligence. We started by introducing to children three different embodied intelligent agents: Jibo robot, Anki's Cozmo robot and Amazon's Alexa home assistant. First, the researchers would demonstrate the vocal commands for activating each agent (e.g. "Hey Jibo" or "Alexa") and some of its capabilities. Then the children were left to explore on their own. After the initial play and interaction, children were also encouraged to program the agents using the existing commercial coding apps developed for each agent. At the end of the session children answered questions about the agents intelligence and abilities as part of the Pre-Perception game (described shortly).

Session 2: Programming AI. Next, we introduced children to the Cognimates AI platform (described shortly) where they could learn how to train, code and test a series of smart programs. To guide this introduction, we created the set of learning activities with starter coding projects.

Session 3: Post-Perception Game. In this final session, we repeated the Perception Game from Phase 1, gathering a post-measure of children's perceptions of machine intelligence had changed, supporting our second research question. Because not all children attended the final session, not all the children completed the perception game. Additionally, in the case of the public after-school program, we were not able to collect any data because of cancellations due to snow. When we did meet the children again a few days later, we only conducted interviews and had a final discussion about what they learned, which concluded with a certificate of participation award.

6.2.3 *Study Materials: AI Platform, Learning Activities & Perception Game*

In this section we will present the Cognimates AI platform we used in the study, the learning activities we used to guide instruction and the perception game we used to measure children's shift in perceptions.

Cognimates AI platform.

For this study, we needed a platform that would allow children to both train, test and program with their custom AI models. There are many professional AI training tools we could have adopted, but because our study was focused on changes in perception of intelligence, and the children in the study had no prior exposure to programming AI, the tools needed to be highly scaffolded for learning. Therefore, we built a platform that integrated the model training, programming, and testing into a single platform, giving learners multiple views of the same training data. This followed the Bifocal Modeling (BM) framework [51], which suggests that representing the same science experiment data in different examples synchronously helps children more quickly abstract and infer information about this data.

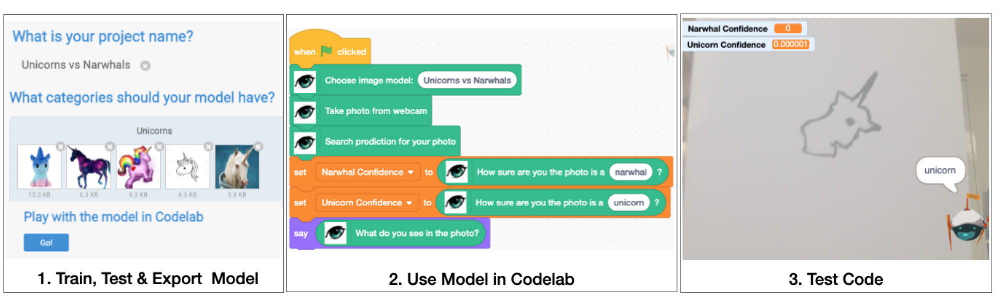

Figure 6.4: Prediction program for a custom image classification model that can recognize Narwhals and Unicorns

Our Cognimates AI platform had two main components: the TeachAI page and the Codelab. The Teach AI page (Fig. 6.4.1) provided children with opportunities to train machine learning models with their own data. The Codelab (Fig. 6.4.2) was the section of the platform where children could write interactive programs using a rich collection of visual blocks, building up AI behaviors to gather user input, classify it, and respond. On the Teach AI Page children could train their own classifiers by providing examples of images and text. For example, a child would train an ideal model, e.g., for distinguishing unicorns from narwhals (see Fig.6.4.1).

Fig.6.4 portrays a case where a child created a game that would use her custom image classification model “Unicorns vs Narwhals” to detect if her drawings were a narwhal or an unicorn, and also report the confidence score in top corner left (Fig. 6.4.3). A character would display the model prediction for each drawing (bottom corner left). In this example, we see the importance of providing children with access to the model confidence scores. While both predictions in this example are correct the confidence scores are very low (0.000036 confidence for the ”Narwhal” prediction, and 0.00001 for the “Unicorn” prediction). This feedback was important for children so they could improve their models and include more data in their training set. Such experiences also allowed children to become more skeptical of predictions they get not only in their game but also in the real world and understand what goes behind the scenes of the image prediction. Children could add new data to their models either directly from their coding projects by using the dedicated blocks (see 2 in Fig. 6.4) or by using the Teach AI page (see 1 in Fig. 6.4).

Learning Activities

During the study, all children completed 3 main learning activities: the ”Make me Happy Program” using text training, the ”Rock Paper Scissors Program” using image training and the ”Smart Home Program” using text and speech training. The children were allowed to choose if they want to do the activities together or individually and were provided printout materials to support the activities. The printout materials would provide children with prompts to lead them to decide what data to include in their training and with code examples that could be used during the programming stage. Researchers would also walk around when children engaged in these learning activities and prompt students with understanding probing questions (i.e. "why do you think it does that?", "how would you fix it?"). After the children finished the 3 learning guides they were encouraged to play and modify other smart programs on Cognimates AI platform.

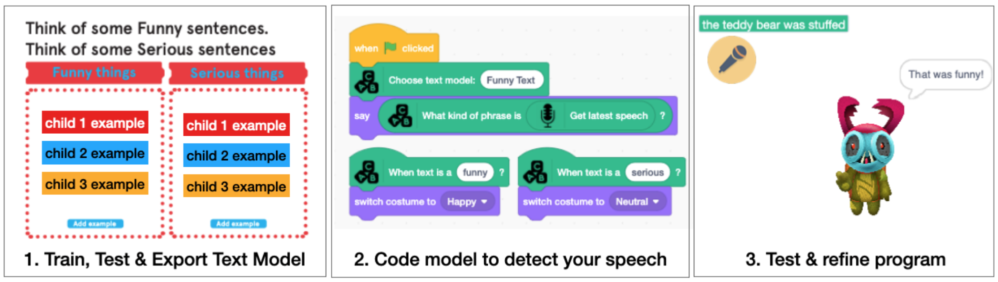

Figure 6.5: "Make me Happy Program"- Text Training Learning Activity

"Make me Happy Program". We started with very simple text training activities like "Make me Happy Program". In this activity, the students had to teach the computer through the Cognimates "Teach AI" platform how to recognize funny messages or serious messages. Once the model was trained with their examples the students could use it in a pre-coded starter project which would make a character on the screen or one of the robots react to their messages. If the message they gave was classified as "funny" based on the model they trained, the character or robot would be "happy". If the message was classified as "serious", the character or robot would be "thinking". The "Teach AI" text models would require them at least 2 categories (e.g. "funny" and "serious") and five examples of text for each category. The text could be one word or an entire phrase. On average the text models would take 2-3 minutes to train.

"Rock Paper Scissors Program." After learning about text training, we introduced children to image training in the "Rock Paper Scissors (RPS) Program." In this activity, the students had to teach the computer how to recognize images of hands showing "rock", "paper" or "scissors." Once the model was trained with their examples, the students could use it in a coding project to test the RPS program with the computer via the webcam.

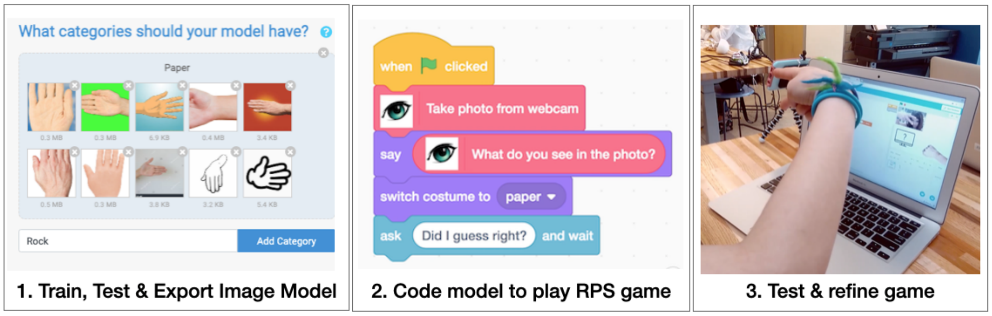

Figure 6.6: "Rock Paper Scissors Program"- Image Training Learning Activity

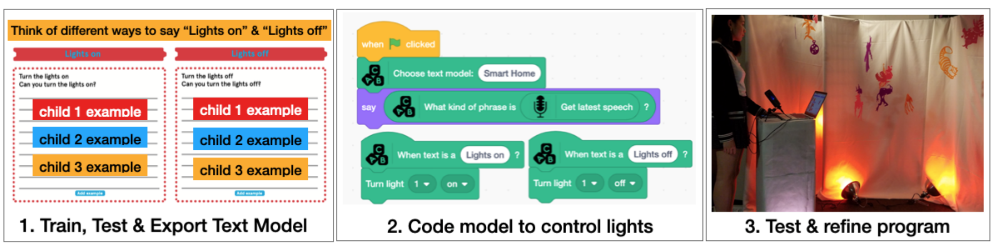

Figure 6.7: "Smart Home Program"- Text & Speech Training Learning Activity

Once they finished coding the program they would test it together with their friends. If the program would fail to recognize some of their hand gestures they would retrain the model to include the new gesture images.

"Smart Home Program." In this activity, the children had to teach the computer how to recognize different commands for turning the lights on and off. They first trained a text model to recognize different types of commands for "lights on" and "light off". Once the model was trained with their examples the children could use it in a coding project in order to control internet connected lights via voice commands.

Other smart programs. After children completed the 3 main learning activities presented above, they were encouraged to test and modify other smart programs. The most popular projects were the *Jellyfish game*, where a jellyfish floats only if you tell it happy messages (using Speech and Sentiment analysis blocks), the *Good boy* program where a dog reacts with sounds and animations to how you talk to it (using Speech and Text classifier blocks). Children often wanted to modify the projects to make both the characters more expressive and to add new types of messages the characters could react to.

Perception Game

To answer RQ2 about children's shift in perceptions, we used an AI Perception questionnaire adapted from Bartneck et al. [42]. This is an existing instrument that has been frequently used to measure children's anthropomorphism, animacy, perceptions of likeability, perceptions of intelligence, and perception of safety of robots. The original instrument examines perceptions across 24 items. Because 24 items was too numerous for our age group, we adapted the items to specifically focus on a subset of 5 characteristics: *it understands me*, *it is smarter than me*, *it will remember me*, *it tells the truth*, *it is friendly*, and *it likes me*; we also reduced the levels to just three: the two endpoints of each the scale (*yes* and *no*) and a *maybe*. Finally, rather than presenting the instrument as a survey, we presented as a "Perception Game", to more effectively engage younger children. In our game, there were a series of printed statements who share a belief about a smart agent. Before asking the questions, the researchers gave an example of how to respond. We conducted the game separately for each of three agents: Alexa, Cozmo, and Jibo. The children were asked to place a sticker closer to the statement with which they most agreed. At the end of the questions researchers wrote the child's name next to their sticker and take a picture in order to be able to later identify the answers.

6.2.4 Theories of Sense-Making

There are many ways to study how children might come to comprehend the behavior of smart agents. Prior work on program understanding has often focused on cognitive approaches, providing learners with interactive representations of program behavior (e.g., [174]). The machine learning, AI, and HCI communities have followed similar trends in pursuing explainable AI, aiming to invent representations that help people reason accurately about AI behavior [17].

In this work, we take a different theoretical stance, instead approaching children's AI literacies through the lens of sense-making. Within this frame, we define sense-making as a process by which people encounter situations or contexts that are unfamiliar, and then need to process and understand in order to move forward [81]. People form new knowledge from engaging in complex and information rich situations in which they may not always have expertise. The learning sciences further consider how learners make sense of quantitative change in complex systems [332], how learners reason with large sets of data [263], and what role argumentation plays in knowledge formation [272].

6.2.5 Data Collection and Analysis

Our study resulted in pre- and post- perception game data as well as video recordings of all sessions at all sites. For the qualitative analyses, the first author and a team of five undergraduate students transcribed the videos and also noted comments on children's body language and non-verbal interactions. The final corpus included 100 pages of transcripts (34,300 words). Once all the transcriptions were finished, the authors each reviewed half of the data independently, looking for ways of explaining the three phases of the study. In this process, the authors separately analyzed each transcript using a combination of etic codes developed from our theoretical frameworks and emic codes that emerged from the interviews themselves [216, 236]. We listed all the the sense-making practices [330] that were found in prior studies with kids and science or math learning [332] and identified connections with a series of themes that emerged from our study. After a final coding frame was developed, all the transcripts were coded by the first author. If new codes emerged, both authors discussed discrepancies in the analyses until they reached agreement. The coding frame was changed and the transcripts were reread according to the new structure. The final list of codes, their definitions and presence across the different study sites is presented in fig.6.8. This process was used to develop categories, which were then conceptualised into broad themes after further discussion. Towards the end of the study no new themes emerged, which suggested that major themes had been identified [53].

	Code	Definition	*Study locations* A	B	C	D
Initial Hypotheses	*Social Judgement*	Child analyzed the agent's social behavior & inferred intelligence from it	x	x	x	x
	Funds of Knowledge	Child is using prior experiences to form theories about behavior	x	x	x	x
	Egocentric	Extrapolating from a child's behavior to the agent's behavior	x	x		
	Observational	Objective details of what the agent is doing without social inferences	x	x	x	x
	Agency	Child would question if the device had agency or not			x	x
Test Assumptions	*Edge Cases*	Testing via edge cases to understand the limits of the agent's intelligence			x	x
	Common Cases	Testing via common cases to reveal deeper understanding	x	x	x	x
	Agency	Testing to see if the agent is autonomous or not	x	x	x	x
Refined Hypotheses	*Post-test Behavior*	Used test results to build more complex models of agent's behavior	x	x	x	x
	Social Intelligence	Used judgements of social intelligence to refine models behavior	x	x		
	Programmability	New hypotheses of machine intelligence referencing programming	x	x	x	x
	AI Training	New hypotheses of machine intelligence referencing AI training			x	x

Figure 6.8: List of codes used for transcripts analysis from the different study sites: A - after-school program in a public school, B - non-profit community center, C - after-school program in a private school, D - private STEM center.

6.3 Findings: Youth Coding AI Games

In this section, we present an overall summary of our perceptions of children's experiences, then discuss our results to RQ1 (how children made sense of agent behavior) and RQ2 (how children's experiences programming AI impacted their perceptions of agents).

6.3.1 RQ1: How do children make sense of machine intelligence when training smart programs?

Within the rich experiences described in the previous section, we now turn to a more granular analysis of children's collaborative sense-making of agent behavior. Overall, our qualitative analysis revealed a clear pattern of behavior: children engaged in a scientific process of initially formulating hypotheses about a smart object behavior, then they came up with scenarios for testing the hypotheses via interaction with the device or with peers, and finally they refined their understanding of the technology either by affirming their initial hypotheses or coming up with new ones. What varied were the tactics that children used to conduct these empirical investigations. To illustrate this variation, we present several tactics that emerged from our inductive analysis.

What type of hypotheses did children propose?

One major source of variation was how children generated hypotheses to investigate.

Social Judgement Hypotheses. Based on our analysis, some hypotheses appeared to be formed by a social judgement of intelligence, where the children analyzed the agent's social behavior and inferred intelligence from it. They would make these hypotheses while interacting with the devices for explaining why they perform an action or not (e.g., "she did not listen" for explaining why Alexa won't play a song). They would also anthropomorphize the devices when they would hypothesize if they are friendly or trustworthy during the initial AI perception game as seen in the following examples:

> *"He just seems like he's in something else right now"* — B., age 12, referring to a Cozmo robot. *"I think he cares about me because when I ask him something, he listens instead of just not even caring about what he says"* — C., age 7, referring to Jibo robot. *"Well, sometimes I ask a question and she says she doesn't know and I'm not completely sure if she's actually telling the truth"* — A., age 7, referring to Alexa. *"She has more of a human personality but she still like doesn't have emotions and the friendliness part"*-Si. age 10, referring to Alexa.

Funds of Knowledge Hypotheses. Our analysis showed that some hypotheses seemed to emerge from funds of knowledge, using prior experiences to form theories about the agent's behavior. This practice is consistent with children's sense making practices in other domains like agent simulations in physics [332] or mathematics [279]. Children referenced not only personal past experiences in interacting with computers or other similar AI agents but also examples and stories they heard about in the media or from their friends and parents.

> *"Uh, maybe they coded something on the computer to tell it, like, tell the computer what to do. Sort of like the computer's brain, computer is the brain"* — C., age 6.5, referring to a Cozmo robot. *"She will remember you because I'm pretty sure just like Siri you can tell her your name, to like ask her to remember you, like who you are. Because you can tell them your name"* — E., age 8, referring to Alexa.

> *"You have to say what text is bad and what text is happy or maybe backhanded, and over time, it'll be able to recognize it without you telling. And, um, I remember seeing a video on the Avengers about why there were such split rates, and uh, the people made a bot"* -Ch., age 7, referring to the text training for sentiment analysis.

Egocentric Hypotheses. In our examination we saw that some hypotheses seemed to emerge from egocentric speculations, extrapolating from the children's ideas about how they would perform a task or solve a problem to the agent's behavior. This was consistent with Papert's findings on body syntonicity, where children project robot geometrical puzzles on their own body to solve a differential mathematics problem in Logo [234].

> *"Well, I've seen lots of pictures and even if I've never seen what, like, a train that has purple stripes, I would just know it's a train by the way it looks, not by its color"* — So., age 8, referring to custom image model trained to recognize trains. *"I think they learn kind of the same and kind of different, because when we learn stuff, we can forget it, but then we can look for it in the real world. But, computers almost never forget it, but if they forget it, they can't look for it in the real world"* — L.,age 7.5, referring to how the Jibo robot learned to recognize faces.

Observational hypotheses. Based on our analysis, we found that some of the children's hypotheses built upon what they had seen the agent do. In this case children would describe details of what the agent was doing without drawing social inferences.

> *"Because it has to recognize every bit, every single thing that's green. If I said 'green', but put this [the green balloon] with like, a background of something else, it might not recognize that because it's supposed to recognize the bigger things as green"* — R., age 7.5, referring to a color sensing coding project. *"I think it works because it says, umm, when you hear good, or happy speech, then, go up and when it hears bad, it just says go down. Then it says when you're out of bounds, make beeping sounds. And when you hit the side, switch directions"* — E., age 8, describing the Jellyfish coding project.

Agency hypotheses. Our analysis showed that participants proposed several hypotheses when asked to evaluate if the agents were smart, trustworthy or human like. Most children proposed these hypotheses during our pre- and post- group discussions about agent intelligence. Children shared beliefs such as:

> *"It's programmed to always tell the truth"* — J., age 8, referring to Jibo. *"Yes, and I think they programmed her so she acts nice"* — L., age 7.5, referring to Alexa. *"Then, the computer would learn, and then it would try to fix it's mistake"* — As., age 7.

These varying sources of hypotheses show that if children believed a smart agent was controlled or programmed by someone else, then they would be more skeptical of its intelligence and human-like abilities. In turn, if the children believed the agent was in control, they would tend to overestimate its abilities to perform human tasks.

How did children test their initial assumptions?

Whereas children's sources of hypothesis were highly varied, our analysis found less variation in how children tested hypotheses, with most directly interacting with agents. What varied were the types of test *cases* that children chose to probe agent behavior.

Testing edge cases. Our analysis showed that children would come up with a variety of edge cases in order to understand the limits of the agents' abilities and intelligence. Children would use all the resources they had in their arsenal to test the agents: from using niche cultural references, to speaking in a different language or trying to find examples of images that are very confusing (e.g., images of dogs with sunglasses). We interpreted this to be similar to practices observed in studies on AI understanding with the use of counterfactual examples [**?**, 24], children in our study would build on responses they would elicit from the agents in order to identify more and more narrow edge cases.

> *"Alexa, play a legendary Kirby rap on Spotify"* — Ch., age 7, talking to an Alexa smart speaker. *"We are trying to confuse it by getting a puppy that looks kind of like a Kirby that is wearing sunglasses"* — P., age 8.5, referring to their custom image classification game.
>
> *"We tried to make him say poems but he wouldn't do it"* — Do., age 7.5, referring to a Jibo robot.

Similar to other examples of playful debugging [185], children would take great pride when they would find a case that would confuse or trick the agent and they would share their discovery with their peers.

Testing common cases. We found that this type of testing was used by children when trying to reveal deeper understanding. In the instances where they knew something should work but it did not, children would try to infer the reasons for failure and come up with other similar examples in order to test their assumption:

> *"He's seeing the colors - it's true because I'm showing the balloon and if I take it away, it's false. Show it, true, hide it, false, yeah? So, now, if you show the color, the paddle should move (paddle doesn't move)"* — A.& E., age 8 & 6.5, debugging their color sensing project. *"Cause I put baseball bat, not a baseball,*

but somehow they must have looked kinda similar, I don't know, and it did it" — D., age 9, referring to his custom image classification program.

Testing agency. Our analysis showed that children used a series of strategies in order to test the nature of the agents they were interacting with and tried to more accurately place the devices on the animate/inanimate spectrum [135, 117, 162]. Participants would either directly ask questions to the devices about their nature (e.g., "How do you work?", "Who made you?") or they would come up with play scenarios to see if they could get the agent to embody different personas. Sometimes children would physically cover the devices, disconnect them from the internet or move them in the room to test how they would behave, similarly to children's attempts to make sense of social robots like Cog [310].

"Alexa, does everything you say really get texted to someone?" — E., age 8. *"I'm trying to figure out how to make it, um, say, when I say 'I am potato', or, or, I want to say 'Are you a potato?' then it will say 'yes' "* — A., age 7, referring to Jibo robot.

"I think it goes to the internet, but if the internet does not have connection, she'll say, okay, it was nice talking with you" — Si., age 8.5, referring to Alexa.

When and how did children refined their understanding?

The testing practices in the previous section were often a precursor to children refining their understanding of agent intelligence. These moments were particularly observable when children were listening and debating perception questions with other children and after several sessions of coding and training where they gained more insight into how smart agents learn from examples. We observed children make several types of inferences from their hypothesis testing.

Post-test Behavior Hypotheses. Some children used the results of testing to build more complex models of the agent's mechanics:

> *"It can make a mistake. Someone could have made a mistake in programming but it is supposed to"* — So., age 8, referring to an Alexa smart speaker. *"Like, if I taught it face recognition, I would go like, oh, this is the real face, no this is the real face, no this is the real one, and it would be really mixed up and it wouldn't know who is who"* — D., age 9, referring to Cozmo robot.

Judgements of Social Intelligence. Some children used social judgements of intelligence to refine their models of the agent's abilities:

> *"He doesn't even know how to pick up a block when I say pick a block"* — N, age 6, referring to Cozmo robot.

> *"Because the Alexa, sometimes I asked her questions and she doesn't understand and sometimes I ask her and she knows it"* — C., age 6.5. *"Because my mind is going both. It will remember me, but it won't, I just can't"* — G., age 6, referring to Jibo robot. *"If the computer knows how to learn, I think it would be easier to make it, um, a robot version of a person, because it can learn like a person, and then it could probably think like a person, move like a person, and act like a person. And then, someday, someone - a person in your house - could be a robot."* — L, age 7.5.

References to Programmability. Many children exhibited more elaborate explanations of agent behavior, grounded in the concrete activities of programming. For example, C., age 6.5, initially said the Alexa smart speaker makes mistakes because *"sometimes she says she doesn't know"*. In the final perception discussion she described the same device in the following way:

> *"Surprised me the most that, at first I didn't really know computers got taught. I thought computers, once they were invented, knew stuff. I didn't know they got taught to do rock paper scissors and all that"* — Em., age 6.5, referring to her

> experience in the study. *"I think its smarter but a person created it so its as smart as the person(referring to creator) but programmed to be smarter"* — said M, age 7.5 *"I know how to use these ["forever" coding blocks]. I've coded using a different program before. Oh my god, this is going to be so cool, I want to use them for the robot now"* — P., age 8.5, referring to his prior coding experience.

References to AI Training. Many children specifically grounded their judgements in their experiences with AI teaching and training. We observed that children would often try to confuse the AI by showing it examples that combine the different things it is trained to recognize (e.g., dog with glasses). The experience of confusing the robot or the computer was primarily attractive for children because it was perceived as fun and because it put them in charge of the process. This led to inferences about the limitations of AI:

> *"I think I know. Well maybe because that one looks more like a drawing. And it doesn't get it because it's a drawing"* — R., age 7, referring to her model prediction result. *"Because it's going to learn what those pictures are going to be"* — L, age 7.5, referring to So.'s model also. *"Cause I don't wanna put more funny words or more boring words, cause if I put like, 2 funny words and 1 boring word, it would probably put funny"* — T, age 7.5, after typing "doing funny homework" in his prediction game. *"Probably it did see me, but it didn't really recognize me, but he can learn to recognize me"* — said a Em., age 6.5, referring to the Cozmo robot.

Peer Support. Children would often support each other in refining their understandings either by explaining how a specific program works or by providing alternative answers to group discussions. Although children tested their initial assumptions about agents either with edge or common cases, when it came to testing the agency of the devices, their testing strategies were based more on peer-aided judgements and examples. The way children explained their reasoning for their answers influenced each other which lead them to internalize new explanations and concepts presented by their peers. For example, here are two exchanges between children facilitating each others' testing:

> Discussion 1: *"So how many examples of trains do you need to give it so it can recognize multiple trains?"*—researcher (referring to a custom image classification model).
> *"165 million"* answered N., age 6. L. and M.,ages 7.5 and 8, added *"10, at least 10"*. Discussion 2: *"I actually don't really know how to use the app"* — C.,age 6.5, referring to a program made for Jibo robot. *"Ah, I got an idea, see if I can get it to look in different places. 'Swipe up', 'swipe right', I'm just trying to see if I can make it look up when I swipe up. By the way, it won't say 'hi' 10 times, but it'll say anything you put in here 10 times, and you can edit this more if you want"*-Ch., age 7, replying to C.'s question about the program.

Overall children refined their understanding of smart agent behavior by evaluating their test findings and coming up with new interpretations for the agents behavior and new judgements to explain their social intelligence. In the final sessions, children will make more complex references to programmability and build on their peer support to refine their AI explanations.

6.3.2 RQ2: How does children's perception of machine intelligence change?

The previous section showed that children examined and reasoned about agency in diverse ways. In this section we consider how these varied forms of reasoning led to changes in self reported judgements about smart agents' abilities.

As discussed earlier, we measured these changes using pre- and post- answers of the Perception Game during the initial and last session in each location. Unfortunately, due to snow, our Public After-School Program's last session was canceled, and we only report results from the three sites that completed the pre/post. Additionally, at some sites, we only asked 5 of the 8 Perception game questions because the children became too impatient to answer all the questions. Many of the children changed their answers to the perception questions pre- and post- (see Fig.6.9). Children became more skeptical of the agent's human like abilities,

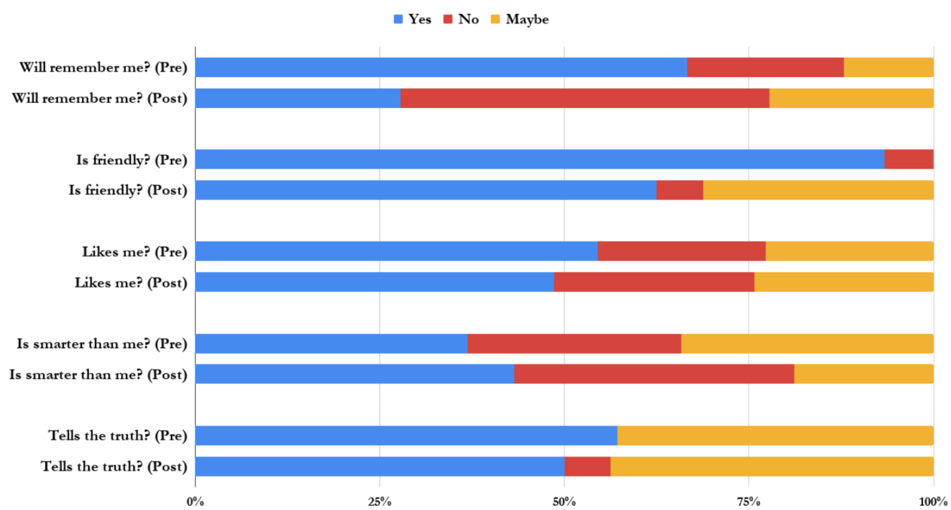

Figure 6.9: Answers shift from five Perception Game pre- and post- answers in all locations

such as remembering them or being friendly. For instance, even when the children said that the agent is friendly or that it will remember them, they would explain it was programmed to do so:

> *"Because, he looks like he has feelings, but he might not. You can make him, like, sad, happy, surprised, bored."*— L. 7 years old. *"He's a robot, so he's probably going to have lots of things programmed into him that he knows and he doesn't have to remember them. Humans have to remember the stuff, but robots don't."* — A. 8 years old.

To understand whether any of these shifts were statistically significant, we did the following. For each of the three completed sites, and for each of the 5 completed questions in the AI Perception game, we performed a Man-Whitney U test with the dependent variable being the answers to questions measured on an ordinal scale ("yes" answer as 1, "maybe" answer as 2, "no" answer as 3) and the independent variable being the pre- and post- conditions. The five questions included two questions about intelligence and legibility attribution (*Is the agent smart? Does it understand me?*), and three questions about socio-emotional attributes *(Will it remember me? Does it care about me? Is it friendly?).* We used a conservative Bonferroni correction for multiple comparisons, setting our alpha to .05/5=0.01 for each site.

Fig.6.9 shows the distribution of responses. Of the 5 tests, two were significant across all the students participating in the study. Overall all students in the program were more likely to answer no to the question *Will the agent remember you?* after the program ($U = 355$, $p = 0.00424$). Similarly, a significant change in ranks of the children that initially said that the agents are friendly changed their answers to "no" and "maybe" at the end ($U = 149.5$, $p = 0.01684$). We did not find statistically significant changes in the other measures; the only other trending shift was an increase in the number of students who said "no" to *Is the agent smarter than you?*.

One noticeable trend in the data is that there are more "maybe" answers in the post- than in the pre-(see Fig. 6.9). This could possibly indicate that, after being exposed to the programmability of AI machines and thinking critically about the machine's agency, the children were reasoning about the complexities and ambiguities of machine intelligence at a higher level in the post- than in the pre- perception discussions.

6.4 Discussion

Our work contributes several new insights about AI literacies by addressing out initial research questions:

- RQ1: *How do children make sense of machine intelligence when training smart programs?* Our qualitative results show that children engage in the scientific method by formulating hypotheses about machine intelligence, then coming up with scenarios for testing, and finally refining their understanding either by affirming their initial hypotheses or formulating new ones. In this process children use a diverse set of social sense-making strategies, drawing from their egocentric perceptions of agency and their empirical observations, to make inferences about agency.

- RQ2: *How do children's perception of machine intelligence change before and after training smart programs?* Based on the pre-post shifts we observed, children's perception of machine intelligence trended toward skepticism. Children also decreased their pro-social attitudes toward the smart agent's behavior. We did not observe changes in children's perception of an agent's truthfulness or ability to like them.

Our results suggest that engaging children in programming with AI leads many children to replace conceptions of smart agents as intelligent with new conceptions of smart agents as fallible but helpful. Importantly, these shifts did not occur for all children, nor did they occur in the same directions, suggesting the challenges of promoting a specific conception of machine intelligence through programming.

Limitations. Some limitations in the study complicate the interpretation of our findings. It was not possible to systematically observe every child's interaction with every agent, nor did every child speak in every group; it may be that children who did verbalize more reasoned differently than those who verbalized less. For the interactions we *could* observe, observing a child reason about an agent does not necessarily indicate ground truth for their conceptions; for example, it may be the case that children were reasoning in similar ways but were verbalizing their reasoning differently. We also did not have data for all perception questions and all sites, nor did our sites cover the many possible ways that culture, community, and collaboration might have shaped sense-making. Since our analysis was episodic rather than temporal, sense-making strategies may have been highly variable within individual and group behavior. Therefore, while modest interpretation of our results suggest that the children in our particular intervention demonstrated diverse reasoning strategies and a shift toward skepticism, other populations could reveal new types of sense-making and different shifts in perceptions.

Programmability impacts intelligence perception. Despite these limitations, our results have many implications for interpreting prior work. For example, as we shared earlier, prior studies on smart agents has shown a clear trend of anthropomorphism, especially of embodied agents [224, 162, 295]. Some studies have even shown that embodied agents can exert peer pressure over children [320] and that children can overestimate the intelligence of embodied agents [100]. Our results show that one reason for this susceptibility is that children have not engaged in examining the mechanisms and limits of AI; when children in our study engaged in this examination, their conceptions of smart agents were still anthropomorphizing, but often less trusting in machine intelligence. These findings are consistent with Duuren's results that identified programmability as a key element in children's perception of social robots' abilities [105]. In another experiment, Vollmer et. al found that 7- to 9-year-old children had a tendency to echo the incorrect, but unanimous, responses of a group of robots to a simple visual task [320]. Thus, the trends in prior work may be conditioned upon what experiences children have had with programming with AI.

Sense-making for AI literacies. Our results also have implications for prior work on children developing AI literacies. Prior work has revealed many challenges, including the importance of children understanding the role of data in shaping machine behavior [217] and the persistent challenge of debugging and comprehension [298]. Other studies with adults has explored methods of bridging these comprehension gaps by helping people develop more robust mental models about AI (e.g., [180, 37, 265]). Our findings suggests that similar approaches may work for children, at least when children are engaged in constructing projects that use AI techniques. Our qualitative findings about children's sense-making strategies also suggest new interpretations of prior research on program understanding. Whereas prior work has largely focused on individual, cognitive accounts of program understanding (e.g., [174, 17]), our investigation of program understanding from a constructionist [233] and social sense-making [81] lens suggests that children rely on numerous assets beyond cognition to understand agent behavior. These assets include social strategies for enacting scientific activities such as observation with peers, discussing hypotheses with peers, as well as introspective, egocentric strategies for inferring models of agent behavior.

Platform design choices. Importantly, our result do *not* speak to work on data literacy. For example, prior work has shown that children engaging with and making sense of data itself has its own challenges [178], as does reasoning about statistics [61]. Our design choices in Cognimates intentionally abstracted and scaffolded away these challenges in service of engaging children in examining agency. Different designs and pedagogies would likely be necessary to promote these different literacies.

Guidelines for designers & educators. Of course, all of these findings have implications for both designers of learning technologies and AI literacies teaching methods for children. Our work selected particular scaffolding to support more accurate assessments of intelligence. While our results were not granular enough to point to specific aspects of this scaffolding that contributed to the strategies and shifts in beliefs that we observed, our work does generate concrete hypotheses to investigate in research and practice. For example, one clear trend in our results was that some children attempted to take the perspective of the agent to reason about its capabilities, trying to imagine how it was making decisions to make inferences about it's capabilities. Designers and teachers might therefore consider methods for promoting perspective taking about AI agents, just as similar work on programming language learning has encouraged learners to take the perspective of a compiler [187, 225]. Another clear trend was that children used their experiences in generating training data to make inferences about agent ability. Designers and teachers might explore methods for engaging children in reflecting on the relationship between the training data, the agent's use of that data, and it's resulting behavior.

Future work. While these implications for design are modest, the need for future work is clear. The results in this paper demonstrate the feasibility of promoting more accurate estimations of intelligence, and begin to reveal the mechanisms behind those changes, but many questions remain about how robust—or repeatable—these changes are in different settings, with different instructors, on different platforms, and using different assessments. Future work should explore these variations, but also extend them to longitudinal observations to understand the robustness of these conceptions over time, and the degree to which they transfer to non-learning settings such as home, play, and adulthood.

6.5 Conclusion

After a 4-week observational study in after-school programs, we found programming with AI leads many children to replace conceptions of smart agents as intelligent with new conceptions of smart agents as fallible but helpful. If we can build a robust understanding of how to promote AI literacies, we will be much better positioned to respond to a future in which AI is embedded in children's' everyday lives. By enabling inclusive AI literacies we will help democratize AI education [98, 199], and by increasing children's AI literacies we would allow them to responsibly use smart technologies for creative learning and personal expression [252]. This vision *must* be attained if our children and our children's children are to live in a just and equitable society.

This study shows that joint peer engagement in coding AI games enables children to discover the core concepts of image and text classification and foster critical reflection on the uses of AI by refining their sense-making hypotheses when testing their classification models and smart games and becoming more skeptical of machine intelligence in the process.

Chapter 7

FAMILY CREATIVE CODING SUPPORTED BY AI FRIENDS

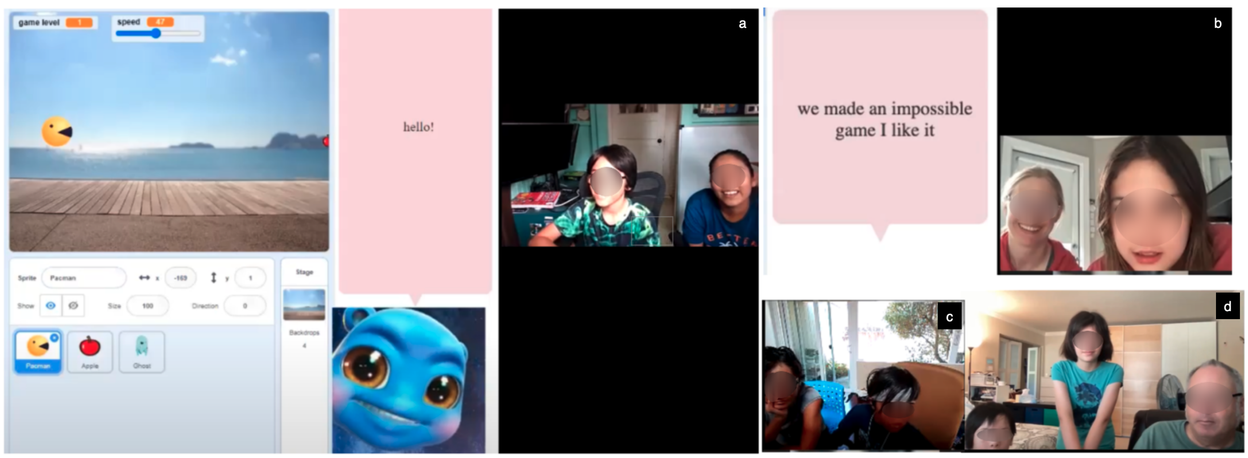

Figure 7.1: Examples of joint family interaction during the study: a. F1 getting acquainted with the AI friend, b. F5 reaction to a joke from an AI friend, c. F8 Mom and son playing a multiplayer Pacman game they just programmed, d. F4 Dad and two daughters brainstorming game ideas with the AI friend

This last study explores how joint family engagement enables children to learn creative coding with AI to enable self-expression. Building on my prior studies looking at how families program together described in chapter 6, in this study, I explore how families program together when assisted by an AI friend.

7.1 Study motivation and contributions

In the era of ChatGPT and ever-increasing automation being introduced across all levels of society, we see a unique and significant role for youth's creative thinking [250] as a driver for constant adaptation, life-long learning, problem-solving, inclusion, and openness. This meaningful future argument is further strengthened by the need for creativity education for future generations with advances in AI, representing a critical differential factor in the competitive global economy [242].

Our current definitions of creativity, such as divergent production, lateral thinking, contextual composition, or possibility thinking, need to consider how we can assess if children are developing their creative voice. For example, children may be speaking out loud some of their imaginative p-prims [86] or trying to engage in pretend play, yet when our theoretical frameworks establish measures of novelty and originality determined by adults, these expressions are often ignored or suppressed.

Children have been able to express and develop their creativity via creative coding with platforms like Scratch, where over 123 million coding projects have been shared by youth from all over the world [218]. *Creative coding* is a rapidly expanding computational domain. It generally refers to programming work that "blur(s) the distinction between art and design and science and engineering"[191], encompassing various interests such as generative art, embedded computing, audio editing, performative live programming, and countless others.

A valuable source of inspiration for how we might approach family creative coding in this study comes from the philosophy and pedagogy of Reggio Emilia, emphasizing listening, documentation, and critical reflection [112]. This pedagogy is based on four core principles: (1) Creative values are the strength & power of kids, pedagogy of listening; (2) Creative relationships are attentive and respectful; (3) Creative environments are physical and emotional; (4) Behavior and dispositions matter, holistic support learning and creative thinking.

Building on Reggio Emilia's approach, we aim to design a creativity support agent perceived as attentive, respectful, and friendly is essential, which is why we will frame our AI system as a "friend" that can provide creative support for families with their coding projects. Prior work in HRI on family-AI interaction shows how much the agent persona design choices can bias the families' expectations and experiences. For example, changing a device's wake word can have ethical implications. A personified name can make a device seem more social, masking the connection between the device and the company that made it, which also has access to the user's data. Additionally, family members are more likely to think an AI agent is more competent and emotionally engaging when it exhibits social cues, like moving to orient its gaze at a speaking person [230]. HRI researchers have used these insights to lay out several AI persona design considerations, including developing warm, outgoing, and thoughtful personalities; understanding the influence of a wake word on user acceptance; and conveying nonverbal social cues through movement [230].

Recognizing that every child is born with immense natural talents [126] and innate creative potential [324], how can we design new learning opportunities and tools for creative thinking that allow children's creativity to flourish in an era of constant technological change and consumption? Engaging in creative expression with code is a social phenomenon where learning does not happen in isolation. Even if children create individual creative projects, they are immersed in the family social context. While in previous chapters of this thesis, I showed how family and peer engagement could support children's creative coding, in this study, I want to explore how we might design AI supports that could further enhance family joint creative coding. This presents a unique opportunity for designing new forms of creative coding for families that involve authentic [123], personalized, and dynamic creative collaboration between families and AI friends. With this current study proposal, we aim to frame creative coding for families from a stance of epistemological pluralism [311], recognizing the validity of multiple ways of knowing and thinking. We believe that future programming tutorials should balance introducing families to new computing topics ("level up") while also leveraging families' creative engagement to enable self-expression and re-imagine computing norms at home.

To begin articulating this research direction, we need to understand children's different creative processes when creating programs with a limited dynamic vocabulary based on their interests, working both on open-ended [307] and closed-ended projects [186]. This study aims to contribute rich and "thick" [129] descriptions of the new ways youth collaborate with AI for creative coding and propose new pathways to engage diverse learners in creative thinking and coding. The future of creative coding for youth is a place where creativity is a language on its own, just like Romanian, Python, or Math. It will create a vocabulary of heuristics, objects to think with [309], and processes that inherently allow families to express their creative ideas with code freely.

Our recent research analyzed longitudinal interactions with various intelligent technologies to find that programmability is essential in shifting the agency from intelligent technology to family members. Children's hypotheses about AI agents and how they tested them were highly diverse [93], as demonstrated in our recent study on family collaborative coding. We found that while families enjoy designing and programming video games together, they struggle to start their games from Scratch or identify how to modify existing complex games. To compensate, they benefited the most from using a vocabulary of programming patterns that expressed specific game behaviors [90], which prompted us to design family learning activities that support the composition and decomposition of programming projects with dedicated micro-worlds and programming patterns.

Now, while there are existing efforts to have AI-powered code assistants for adults, such as GitHub's Copilot [215] and Replit's Ghostwriter [248]. Currently, no AI coding assistants are designed to support creative coding for youth and families [345]. Our study aims to answer the following research question: *RQ: What are the ways in which children (7-12) and parents engage in collaborative creative coding supported by an AI friend?*

Building on our prior work on creative coding [93] and AI literacies for families [103, 92], as part of this study, I extended the development of the Cognimates platform I created and developed in 2018. Cognimates [89] is presented in chapter 6. The goal was to include various possibilities to collaborate with autonomous agents (AI Friends) in curated coding micro-worlds that match the diverse families' interests, such as intelligent games or drawing with code (see examples in appendix). To test a prototype of a new AI-supported creative coding platform (CogniSynth) with children and parents, I ran a 3-week Wizard of Oz (WoZ) study [87]. The WoZ prototyping approach, widely used in human-computer interaction research, is instrumental in exploring user interfaces for AI-assisted tasks [157, 161, 261].

The study involved 19 participants (10 children aged 7-12 and 9 parents) and observed families' creative coding practices and interaction with the AI friend trying to help them. This investigation makes three contributions to the understanding of AI-supported creative coding. First, we provide empirical evidence of how families engage in collaborative creative coding with an AI friend. Second, we present how families' joint engagement with AI supports has unique benefits. Finally, we discuss how the theoretical model of creative self-efficacy is relevant to developing creative AI literacies in families.

7.2 Study Procedure

Eight families participated in three study sessions, each lasting 30-40 minutes, comprising two sessions of games programming with AI Friends and one final interview. During the games programming sessions, families interacted with an AI Friend, controlled remotely by the researcher via a Wizard of Oz (WoZ) interface. The AI Friend provided creative prompts, coding debugging, and ideas to support families. At the end of each session, families engaged in a final interview to give feedback on prior interactions with the AI Friend and suggest new features or designs for the AI Friend and the platform.

7.2.1 Study Participants

We recruited eight families (10 children aged 7-12 and nine parents) from six different US states for our study. Families had a wide range of socio-economic backgrounds and spoke seven languages other than English (see demographics in Table 7.1). Families also declared various levels of exposure to AI technologies and programming in the screener survey. In addition, each family member completed an intake questionnaire (demographics) and described their programming experience. The study took place online on a video conference application.

Family ID	Parent(s)	Language(s)	Child(ren) and Age(s)
F1	Mom (S.), Dad (J.)	English, Spanish	Son, 7 (G.)
F2	Mom (T.)	English, Spanish	Daughter, 12 (H.)
F3	Dad (J.)	English	Son, 11 (G.)
F4	Dad (D.)	English, Mandarin	Daughter, 12 (A.) Daughter, 10 (M.)
F5	Mom (C.)	English	Daughter, 10 (K.)
F6	Mom (M.)	French, Cantonese, Mandarin	Son, 11 (Z.) Daughter, 7 (K.)
F7	Dad (M.)	English, Japanese	Son, 10 (M.)
F8	Mom (L.)	English, Tagalog	Son, 12 (S.)

Table 7.1: List of families that participated in the study

7.2.2 Study Sessions

Session 1: Modifying a Coding Micro-world. The researcher introduced families to the CogniSynth platform and the AI Friend in this session. The AI Friend then guided them through the rest of the activity, presenting them with a list of 3 different micro-worlds [307] (Fish Game, Drawing Game, and Pacman Game micro-worlds) and asking them to pick one and modify it to make it more fun. I decided to only give participants these three options in order to scope their potential explorations, building on prior work that shows constraints can be a source of creative inspiration[62]. Throughout the activity, the AI Friend provided encouragement, ideas, and reflection questions to the families.

Session 2: Choosing Programming Patterns to Create a Game. In this session, the AI Friend gave families the task of picking three programming patterns from a given collection and using them to create a game. Examples of programming patterns were provided, showing code scripts for creating different game events (e.g., firing objects or jumping over obstacles).

Session 3: Final Interview. During the final interviews, the families interacted with the researcher and reflected on their interactions with the AI Friend and their study experience. The researcher showed them video snippets from their prior study sessions and asked them to describe what they did and when the agent was helpful. They were also shown screenshots from their study games and asked to identify moments when the AI Friend was helpful. Families were asked to give feedback on alternative UI designs and describe how they would like to interact with the AI Friend to get ideas, help with debugging, or encourage coding. Lastly, each family filled out a sheet to rate different attributes of the AI Friend on a scale.

7.2.3 Study Materials

CogniSynth Platform This section describes the CogniSynth platform I designed and built for this proposed study. It has two main views: one for the family and one for the wizard. The family view consists of a Coding Blocks Library and Coding Area, an AI Friend Response Area, and an AI Friend Avatar Window (see fig.7.3). The researcher needs to open the wizard view and have Snap Camera Studio installed for the AI friend window to appear on the family view. We have developed a custom Snap Camera filter for each AI Friend that tracks the researcher's face and expressions and maps them to control various 3D avatars (see Figure 7.4). The wizard view has a dialogue window, a quick reactions menu, and a list of prompts. The researcher can write messages that will be sent via the AI friend in real time or select a pre-written message or prompt via a keyboard shortcut. They can also upload and send images in the dialogue box (see fig.7.3). This platform was built in REACT and is currently hosted online at creative-ai-woz.herokuapp.com).

AI Friends For each AI Friend, we have created a custom Snap Camera filter. This filter tracks the researcher's face and expressions and maps them to control various 3D avatars (see fig.7.4). When no face is detected, a pre-set background is displayed. The filters were created using Snap Studio SDK and open-source 3D models from thingiverse.com. Each AI

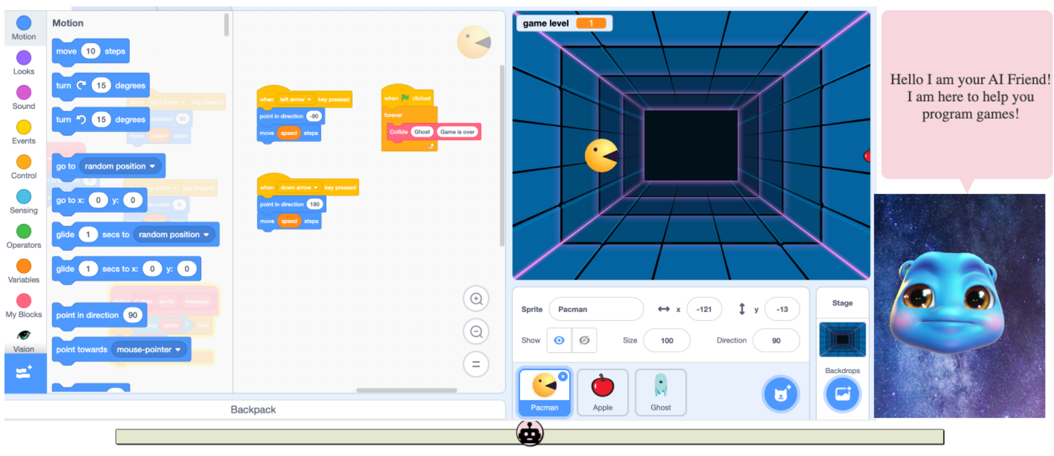

Figure 7.2: The CogniSynth family programming interface consists of three main components: the Coding Blocks Library and Coding Area, the AI Friend Response Area, and the AI Friend Avatar Window.

Figure 7.3: The CogniSynth wizard interface is composed of three main components: a dialogue window, a quick reactions menu, and a list of prompts.

Figure 7.4: The list of AI Friends includes Water Bear, Wall-E, Maskman, and Dinosaur.

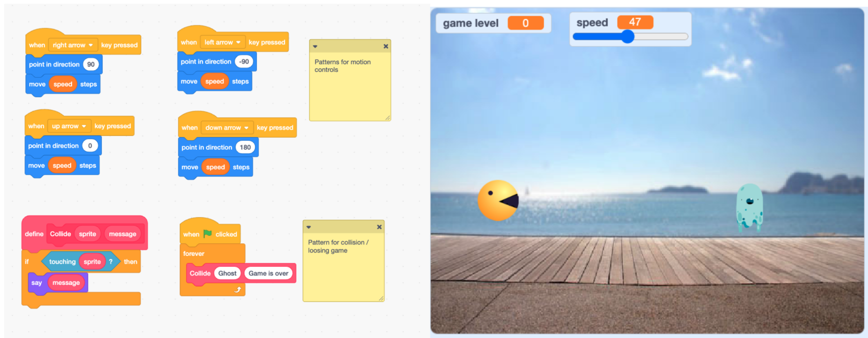

Figure 7.5: A "Game micro-world". Available blocks(left) and the game controlled by the program(right).

Friend communicates with families on the screen using a text dialog. More details about the interaction with the AI friends and the platform are presented below.

Creative Micro-worlds To help children explore possibilities without being limited by a starter program, I created three themed micro-worlds, such as "'Drawing micro-world", "Pacman Game micro-world" and "Fish Game micro-world". The "'Drawing micro-world" allows families to create a custom drawing program by selecting colors, stamps, and brush-stroke effects. The "Pacman Game micro-world" enables families to program a Pacman game with a Pacman and Ghost characters they can move around on the screen. The "Fish Game micro-world" enables families to program a game with big fishes that eat smaller fishes.

A micro-world consisting of just a few blocks was carefully chosen to express various programs to engage families' creativity. For example, in the "Pacman Game micro-world" (fig. 7.5) you can see a micro-world is made up of blocks that prompt a user for commands, control Pacman movement, animate Ghosts, and check the input for a condition. This small set of blocks can be combined to create various programs, allowing families to create many variations on the Pacman game or modify it into a new game. In our prior work, when we gave families such micro-worlds instead of the complete coding platform, they had an easier time constructing valid programs and imagining potential behaviors for game characters [90].

AI Friend Prompts Building on prior work on social robots supporting youth creativity in the context of drawing activities, storytelling, and LEGO programming [22], I curated and adapted a list of creative prompts for my study. Depending on the family action, I could select any of the prompts in the wizard view (or type other similar ones). The AI friend would then display the text on the family coding interface.

Free and creative writing methodologies inspire this series of creative prompts [107]. The prompts are grouped into reflective questions, creative prompts, and positive reinforcement (see Table 7.2). The AI friend only performed these text prompts and did not engage in other form of behavior.

Programming Patterns Soloway and his colleagues provided evidence that novice and expert programmers have schemas that match commonly used code patterns, which they termed *programming plans*. These plans are small program fragments that achieve a goal, such as selecting values from a list that match specific criteria [288]. Other studies have shown that novice programmers use many code tracing and pattern recognition strategies [114]. Specific research on children's game programming demonstrated that providing programming patterns and templates for different game types could facilitate computational literacy and expression [306, 149, 118, 181].

In light of these approaches, we created a collection of eight programming patterns that families could use as examples when programming their games in the second session of our study. Our patterns provided examples of game behaviors, such as throwing objects and animated motion (see Table 7.3).

Co-Designing UI Mock-ups The goal of these UI Mock-ups was to involve families in the design of creative support behaviors for the AI friend. This included different ways for the AI friend to express new ideas during coding and different modes of interaction with the AI friend (e.g., via text, audio, or images). In addition, families were asked to provide feedback on these scenarios and suggest their feature ideas or new ways to interact with existing features. Figure 7.6 shows examples of the platform UI mock-ups used for codesign.

Reflective Questions	Creative Prompts	Positive Reinforcement
Can you tell me why you did that?	*What are some other things you can make your project do?*	*That is such a great idea! Good job.*
What will you do next?	*What else can you make the character do in this situation?*	*You think of some really cool rules for the game.*
What are you trying to make?	*Can you make it do something else?*	*Well done. That was so creative!*
How are you going to do that?	*Let's think of some fun uses of the game.*	*I would not have thought of that. Good going.*
What are the blocks would you need for that?	*Is there a better way to program this event?*	*Great solution!*
Do you have any questions about this script?	*Let's try to make an obstacle for the character.*	*Oh, that is fun!*
Is that the best way to do that?	*Let's try to make the character move when you press space*	*I love this animation!*
Let's read the code and see what it does	*Should we animate the character?*	*Great character design!*

Table 7.2: Examples of prompts used by the AI friend.

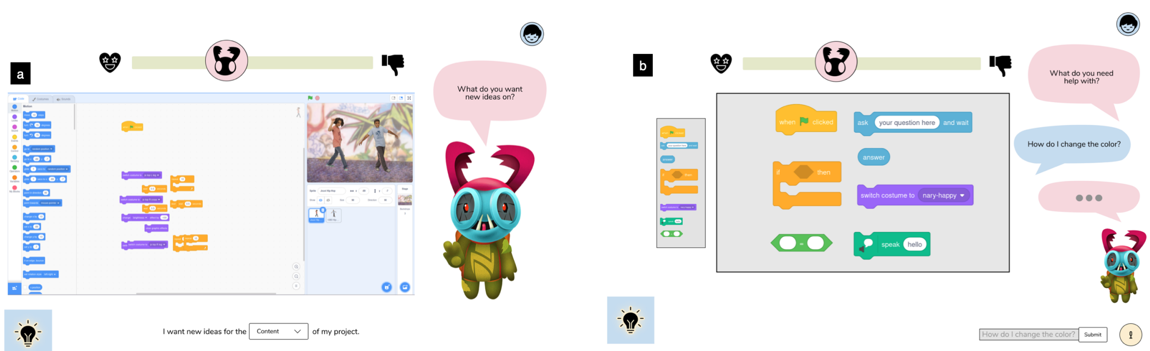

Figure 7.6: Examples of platform UI mock-ups used for codesign: a. Options for eliciting specific ideas from the AI friend, b. Options for different chat modalities with AI friend.

Pattern	Description
Motion	Control motion of characters (i.e. press right moves Pacman to right)
Collectables	Collect items (i.e., food in Pacman)
Obstacles	Avoid game elements or other characters (i.e. avoid maze walls in Pacman)
Firing	Players can create characters on key press (i.e spaceship firing missiles)
Animation	Non-player characters can move on their own (i.e. Ghosts gliding in Pacman)
Clone	Create multiple copies of a character that have the same behavior.
Win /Lose	Determine how to win or lose a game (i.e. collect all food to win)

Table 7.3: List of Programming Patterns used in Pacman Game Microworld for the study.

AI Persona Sheet At the end of the study, families completed the AI Persona sheet (see fig.9.12 in appendix). This sheet contains different potential characteristics of the AI friend, depicted on continuums. Family members marked where the AI friend fell on the scale for each characteristic and then discussed why they made those choices. The list of characteristics was adapted from a survey designed by Bartneck et al. [42]. This instrument has been frequently used to measure children's anthropomorphism, animacy, perceptions of likeability, perceptions of intelligence, and perception of the safety of robots. The original instrument examines perceptions across 24 items, but we adapted the items to focus on a subset of 9 characteristics. The sheet is included in the appendix.

7.2.4 Theoretical Framing: Creative Self-Efficacy

The creative self-efficacy theory [303] is derived from Bandura's more general concept of self-efficacy [34]. Self-efficacy is the confidence an individual has in their capability to attain a particular objective or perform a specific task [34]. Creative self-efficacy is an extension of this concept and is the confidence one has in their ability to be creative. Creative self-efficacy theory asserts that an individual's beliefs about their creativity will impact their willingness to attempt the creative task, the level of effort they expend, and the duration of their persistence when faced with difficulty [303].

In a study with mental health professionals and social service providers, Tierney and Farmer explored the development of creative self-efficacy and performance over time. It was observed that when individuals perceived recognition for their creative performance and if their supervisor expected them to be creative, their creative self-efficacy improved with time. Furthermore, an increase in creative performance was linked to a higher degree of creative self-efficacy [303]. To put it differently, when someone succeeds in a task and has a "mastery experience," their self-efficacy regarding the task will increase; conversely, when someone has a high self-efficacy for a task, they will accomplish it at a higher level than if they had a lower sense of self-efficacy [34]. A recent study investigated the association between creative self-efficacy (perceived creative ability) and middle and secondary school students (N = 1,322). The findings showed that students' mastery- and performance-approach beliefs and teacher feedback on creative ability was positively linked to the students' creative self-efficacy [44].

The creative self-efficacy (CSE) theory has been used by creativity-support tools (CST) designers to focus on children's and parents' beliefs that they can successfully perform in a specific creative process. For example, Mosaic, an online creative community, builds creative self-efficacy by sharing the design process for creative work rather than showcasing finished projects [170]. Parallel prototyping in creative work leads to better design results and increased self-efficacy, as demonstrated by Dow et al. [88]. In addition, the Creativity Project utilized CSE theory when designing a mobile game to engage youth in different kinds of creative thinking and behavior at a science exhibit. Results indicated that young people with higher creative self-efficacy enjoyed playing the situated mobile game more than those with lower creative self-efficacy and that playing the situated mobile game focused on creative activity contributed to an increase in creative self-efficacy for participants [32].

Regarding families, prior work has found that children's creative self-efficacy can be either positively or negatively influenced by parent–child relationships in after-school program activities [193]. Gralewski et al. found that parental child acceptance and autonomy support were weakly but positively related to children's creative self-efficacy and creative personal identity [134]. Tang et al. found that parent support and creative self-efficacy significantly predicted student creative self-efficacy in studies on parental influences on student general and Science, Technology, Engineering, and Mathematics (STEM) creative ideation behaviors [296]. Shang et al. found that a robotics STEM camp program significantly affected students' self-efficacy and computational thinking in rural elementary schools. Students' experience with engineering-based activities had a statistically significant impact on computational thinking skills, and programming experience affected self-efficacy concerning participation in STEM activities [276].

This prior work highlights how CSE theory can be used to analyze how AI friend creative coding support might impact families' perceived efficacy in producing ideas, solving problems, and elaborating or improving others' ideas when jointly programming games.

7.2.5 Data Collection and Analysis

I collected video recordings of all study sessions and in-situ activity feedback and reflections from the slider provided on the platform. In addition, I recorded logs on all the support provided by the AI friend in writing. Both children and parents were encouraged to speak aloud [67] during the programming activities. After the activities, I prompted the family members to describe the interaction with the AI Friend in a final interview. I also collected copies of all the games the families built on the CogniSynth platform and analyzed them for correctness, diversity of features, and uniqueness.

I transcribed the videos for our qualitative analysis and noted comments on families' body language and non-verbal interactions. Once all the transcriptions were finished, I reviewed all the data independently, looking for ways of explaining the three sessions of the study. In this process, I analyzed each transcript using a combination of etic codes developed from our CSE theoretical framework and emic codes that emerged from the interviews themselves [216, 236]. After developing a final coding frame (see Table.7.4, I coded all the transcripts. I then used this coding process to develop categories, which I conceptualized into broad themes [53]. I also transcribed and analyzed family reflections and feedback from the final interviews and their codesign sheets for the AI friends. Finally, I collected and analyzed all the projects created by families. Each project was analyzed for correctness, diversity of features, and uniqueness. I also traced the provenance of ideas to show what code came from the AI or parental suggestions.

7.3 Findings: Collaborative Creative Coding with AI

In this section, I first present an overview of coding projects created in each session and discuss how they interacted with the platform and the AI friend. I then present how children (ages 7-12) and parents collaborated in creative coding with the help of an AI friend when modifying a game or creating a new game from patterns. Finally, I conclude with families' feedback and codesign ideas for future AI friend features and attributes.

7.3.1 Findings from Session 1: Modifying a game

In the first session, children and parents worked with the AI friend to modify a Pacman game program. One game (F5) featured multiple obstacles and animated disco ghosts, where Pacman had to reach the green line to win. Another game (F3) had original artwork, a dynamically changing maze, and ghosts spawning at different parts of the maze to chase Pacman. In a third game (F6), Pacman had to navigate a hand-drawn maze followed by ghost clones. A fourth game (F8) was a multiplayer game with good and bad Pacman competing to shoot more ghosts. The fifth game (F1) involved Pacman fighting with a ninja

Code	Definition	Example
Produce ideas with AI friend	AI friend helps families come up with ideas for their games	*"How can a ghost go over the walls?"*
Express ideas in code with an AI friend	AI friend helps families find the right programming blocks to express their games ideas	*"How can you make the ball move faster?"*
Debug with AI friend	AI friend helps with code reading, debugging and testing	*"Let's test the "shoot"script"*
Elaborate on AI friend idea	Family members build on suggestions from AI friend	*"Oh I like the zombie ghost idea let's make it green"*
AI friend elaborate on family idea	AI friend makes suggestions building on family ideas	*"Should the bear say "Ouch" when touching hedgehog?"*
Ai failure	Instances when AI friend fails to help or provide useful ideas	*"Maybe your mom can help with clones?"*
Joint-engagement support	Instances when AI friend supports family joint-engagement	*"How about letting your sister code the taco animation?"*
Creative coding identity	AI friend helps kids develop their creative coding identity	*"I love your dynamic maze idea! So fun!"*

Table 7.4: Summary of final codes and definitions for family creative coding with an AI friend.

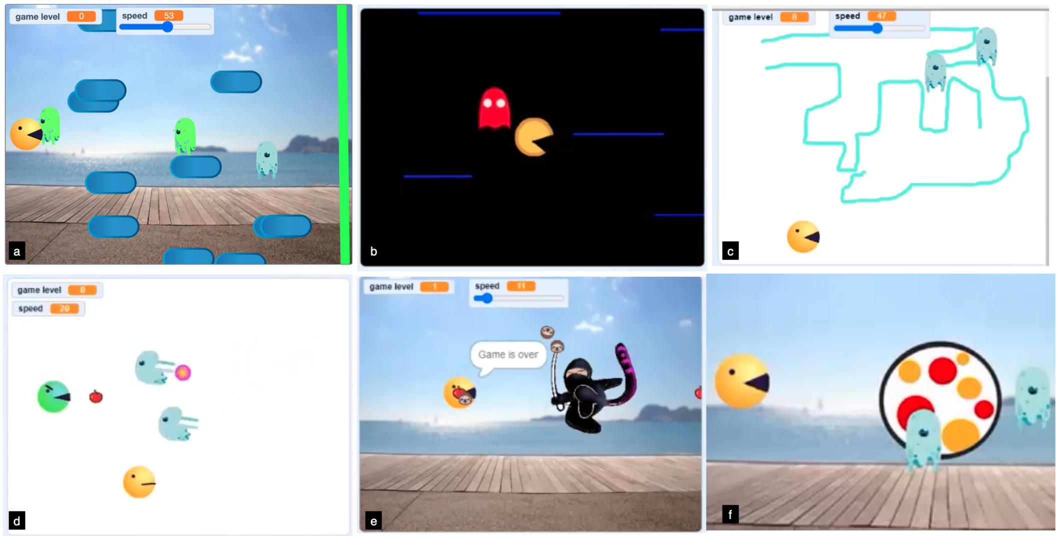

Figure 7.7: Examples of family games from session 1: a) F5: game with multiple obstacles and animated disco ghosts, Pacman needs to reach the green line to win; b) F3: game with original artwork, dynamically changing maze and chasing ghosts; c) F6: game with hand-drawn maze and ghost clones; d) F8: multiplayer game where good Pacman competes against bad Pacman to shoot more ghosts; e) F1: game with ninja shooting sloths, and f) F2: game with a giant ball toppling ghosts on the screen.

that shoots at sloths. Finally, the sixth game (F2) featured a giant ball that could topple ghosts on the screen. Each game showcased unique features and challenges, and the AI friend helped the families with coding, guidance, and idea generation (see the screenshots of the games in fig.7.7).

Sessions featuring siblings working together revealed that the older ones often dominated the coding, disregarding the ideas of the younger ones (F4 and F6). In contrast, sessions with parents and children showed positive experiences with collaborative coding, clear communication, and guidance. Parents helped their children understand the code, and the children accepted the AI friend's suggestions, leading to improved games.

For example, one session featured a child who made a dinosaur shoot with bread and asked the AI friend many questions, with the father also assisting (F7). In contrast, another session involved a child who initially stopped the AI friend to focus on their ideas and conversation with their mother. However, once they were satisfied with the game logic, they reopened the AI friend, prompted by their mother, to get more ideas and started building on the AI friend's suggestions to improve the game (F8).

Another session involved a child who spent a long time changing the speed of the ghost, with both the AI and mother offering help (F5). Finally, a positive session featured a mother-son collaboration, with the son driving the coding and explaining what he did to his mother, accepting the AI suggestion of good/evil Pacman, and adding a second Pacman to create a multiplayer game (F8) (see fig.7.7d). On the other hand, a session with two arguing sisters resulted in them ignoring the AI friend's suggestions and creating a game mocking their parents as an act of disobedience (F4). However, the study did not explore whether these difficulties were due to the AI's limitations or simply the dynamics of sibling relationships.

Our findings suggest that children's experiences with AI can vary, depending on their individual preferences and prior experiences. For example, collaborative coding with parents led to more positive outcomes, whereas sibling collaborations showed a dominance of older siblings, potentially leading to the exclusion of the younger sibling. In addition, sessions involving clear communication and guidance led to children accepting AI friend suggestions, resulting in improved games. However, negative experiences with the AI friend, such as arguments between siblings or discontent with the AI friend's suggestions, could highlight the importance of individual family preferences and prior experiences when working with AI.

7.3.2 Findings from Session 2: Making a game from patterns

The second part of the study involved sessions where children and parents programmed a new game using programming patterns with the help of an AI friend. Overall, families programmed various games with the help of the AI friend. For example, one game (F6) had hedgehog-stinging animated bears gliding over the screen who were protecting gems. In another game (F3), players could compete against a robot in a ping pong game where the paddles created cool animations of the ball bouncing. The game from (F2) featured a kiwi bird eating fruit that gave it different points. Finally, F7 made a game about a dinosaur

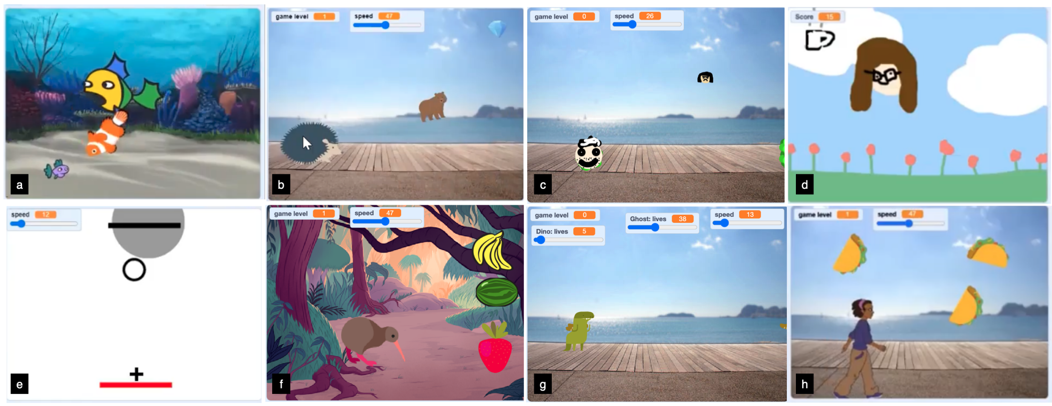

Figure 7.8: Examples of family games from session 2: a) F1: a fish game where a big fish is eating smaller fishes and getting bigger; b) F6: a hedgehog stinging bears protecting gems; c) F5: a kid running into friends; d) F4: (A. 12 years) a coffee drinking game; e) F3: a ping pong game against a robot; f) F2: a kiwi bird eating fruit; g) F7: a dinosaur shooting bread at ghosts; h) F4: (M. 10 years) a girl dancing in a taco rain.

who could shoot bread at ghosts and had the power to call the bread back. Each game was unique and demonstrated the creative ideas that families brought to their programming with the guidance of the AI friend (see fig.7.8).

One session involved much more advanced coding on a fish game, with the AI friend scaffolding the coding (F1). The child could express ideas in code, adding a special fish, collision, animation, and sound effects. The child struggled with conflicting rules and delays for event triggering due to long sounds, resulting in a fun effect when the eating fish became much bigger (see fig.7.8a).

Another session involved a child (F2) making a custom bird game, spending a lot of time hiding food clones, and asking many questions of the AI friend (see Fig. 7.8f). The child in F3 appreciated the AI friend for its fun ideas and compared it to their dad, feeling lonely coding and desiring a sidekick. They made a ping-pong game with “robot” and “player,” adding nice effects when the ball touches the paddles and rushing to finish the game (see fig.7.8e).

A session with the younger sibling from F4 went well despite the moderate success from session 1 when she was programming with her older sibling. She made a game with a girl walking while there were flying tacos everywhere, learning about broadcast events and clones for tacos, and wanting to share her game with her dad. Finally, she made the girl fall asleep and snore if she ate too many tacos (see Fig. 7.8h). Another child added a dynamic maze, multiple clones, and a finish line, learning how the clone works and how to detect collisions. Their mom assisted when stuck with custom-defined functions for ghost clones following Pacman (F5). Siblings from F5 collaborated better in this second session, prompted by the AI friend to take turns coding, making a hedgehog and disco bear game after deleting all the code patterns and starting from Scratch (see fig.7.8b).

Overall, this session showed that the AI friend could prompt collaborative coding, facilitate idea generation, and support the development of more advanced coding skills while families. In addition, in this session, the AI friend showed the potential to support children's learning and creativity in coding various games, providing guidance, suggestions, and encouragement.

7.3.3 Findings from Session 3: Final Interview

The third study session involved interviews with children and parents to provide feedback about their experience interacting with the AI friend. All children said they preferred the AI friend to ask questions or give hints that would help them fix bugs or implement specific game behaviors rather than giving them the solution or fixing the programs. One child said they would like the AI friend to show how code changes would trigger unusual behavior (F8), while another said they would like the AI to ask clarifying questions when unsure how to help (F1). All participants said the AI friend's affirmations and encouragements helped a lot, and many children expressed a desire for the positive feedback to be personalized.

Seven of the children noted how much they appreciated the AI friend getting them unstuck and found it less frustrating than coding alone. However, they sometimes found the wording used by the AI to be confusing, and in those cases, they said images would be very helpful,

Figure 7.9: Feedback for the AI friend persona and attributes from all families (F3 did not complete the sheet)

especially for locating specific programming blocks. In addition, having the parent present helped them understand the AI's suggestions and questions better. Some children said they would like the AI friend to detect when they are struggling with coding and help them before they get frustrated, while others expressed a desire for the AI friend to not interrupt them when they are quiet and wait until they initiate the chat.

Families mentioned that children spent more time problem-solving their code with the AI friend than when they program alone. The children compared coding with the AI friend to coding based on code from another child, where they can read and get inspired but need to make enough changes to the code to make it their own game. Several children said they would like to be able to access a transcript of their questions and chat history later.

When it came to their preferences for the AI persona and attributes, half of the children said they would like the AI friend to be more like an animal, while the other half said

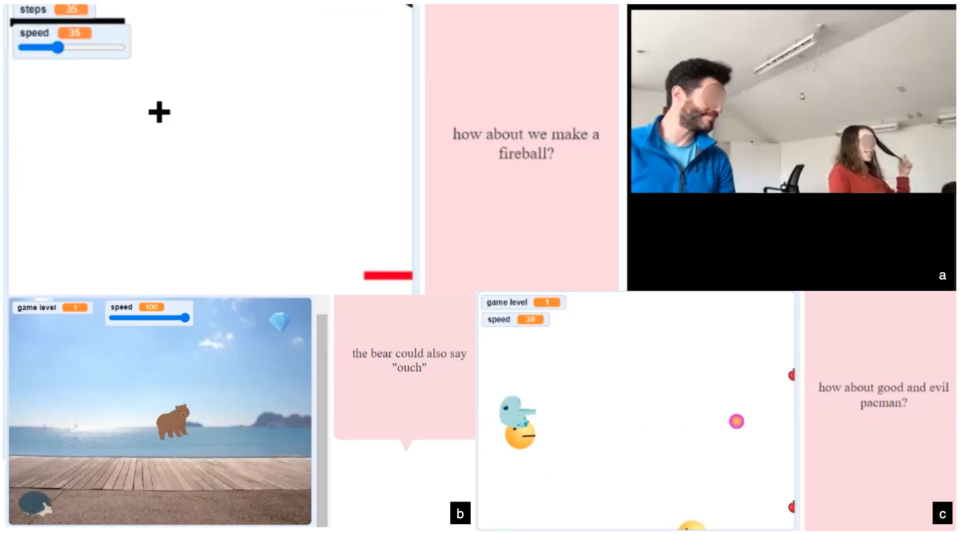

Figure 7.10: Examples of AI friend ideas: a) F3: adding effects to the ping-pong ball to make it look like a fireball; b) F6: making the bear say "ouch" when touching the hedgehog; and c) F8: creating good and evil Pacman characters for the multiplayer version of the game.

they would like it to be more like a person, and one family (F4) picked the middle option. All children said they would like the AI friend to be friendly and understand them. They expressed a desire for the AI friend to be smart, but not too smart, and still let them figure things out. Some children said they would like the AI to act more like a friend, while others said they would like it to act more like a parent (see fig.7.9).

In the next sections, I will present families' ideation, debugging, and interactions with the AI friend in greater detail, providing examples of the AI friend's positive interventions and missteps.

7.3.4 Ideation with AI Friend

The AI friend's role in the game design process involved stimulating and supporting ideation among families (see fig.7.10). This was achieved by asking questions that guided families to choose and express their creative ideas. For example, as seen in the case of F3, the AI friend suggested what would happen when a ball touched a paddle and offered suggestions on what to add to make the game more exciting. In response, the child expressed interest in adding the suggested effect and asked the AI friend to help him implement it:

> *"Should we add an effect on the ball when it touches the paddle?"* — AI friend. suggestions to F3 family.
> *"I should do that, but can you help me do that?"* — G., 11 years old, responds to an AI friend suggestion.
> *"How about we make a fireball?"* — AI friend responds (see fig.7.10a).

Moreover, the AI friend guided children struggling with generating ideas or experiencing a mental block. For instance, in F4, when the child was unsure of what should happen next in the game, the AI friend suggested adding an effect and a motion direction, which guided the child to think further about the game. In this way, the AI friend served as a mentor that stimulated the child's creativity and helped them overcome mental block:

> *"Hmm. It looks interesting. Just flying Tacos in the sky. Okay, how about you make it go down?"* — AI friend. suggestions to F4 family.
> *"Can it then go this way (points down the screen), and then what will happen?"* — M., 10 years old, responds to AI friend suggestions.
> *"Yeah. So when the space bar is clicked, we want to move it. How should we control the motion direction?"* — AI friend responds.

In these two examples of ideation with the AI friend (F3, F4), we see how the AI friend encourages families to think creatively by helping them generate, develop, and express their ideas.

Additionally, the AI friend supported family game development by offering examples of game mechanics or elements that they could use as a starting point to create their own unique game. For example, the AI friend suggested to F8 to create a good and evil PacMan, which gave the family an idea of how to modify their game (see fig.7.10a). When the siblings from F6 engaged in collaborative programming during the second session, the AI friend could also assist in ideation. During the session, the older brother demonstrated the interaction between the bear and the hedgehog characters on the screen by physically manipulating their movements. This demonstration prompted the younger sister to propose a new game concept, "let's make the bears' guard crystals." Through a series of iterative demonstrations and discussions, the siblings ultimately decided to have the bears glide to random positions on the screen and control the motion of the hedgehog using the space bar. In addition, the AI friend suggested incorporating a feature in which the bear would say' ouch' upon colliding with the hedgehog, which the siblings subsequently integrated into their game (see fig.7.10b). By providing specific character examples and ideas, the AI friend helped families develop their game designs and inspired them to create unique and engaging games.

S., a 12-year-old from family F8, expressed that he preferred the AI friend's support in conceptualizing game ideas, as it allows him to build upon the initial idea and use his creativity. He suggested that a separate section for art support would also be useful for situations where he needs assistance in designing artifacts for his game:

> *"I prefer help with game concepts because then you can mix them to build upon it. So it [referring to the AI friend] gives you an idea, and then you can use your mind and creativity to do it, but maybe there could be another section where it focuses on the art and say, "I know what to do with the cone, but how should I make everything look?"* — S., age 12, F8, talking about when the AI friend is most useful with debugging.

Several children recognized that AI friend's ideas helped them when they didn't know how to start their game or when the game was becoming boring. For example, H., a 12-year-old from family F2, acknowledged the usefulness of the AI friend in overcoming the challenge of starting a coding project and when encountering a roadblock in the creative process. She believed that many people would appreciate the AI friend's support in these situations:

> *"Most people would like coding with AI friends because one of the hardest parts of our project is when you start and also when you run into a wall, and you're*

> *out of ideas."* — H., age 12, F2, talking about situations when the AI friend is most useful in ideation.

G., an 11-year-old from family F3, noted the helpfulness of the AI friend in adding additional features and effects to their project, which added a fun element to their coding experience. They acknowledged that coding could become boring, and the AI friend helped to prevent their brain from getting foggy. The child appreciated the extra help provided by the AI friend during their coding project:

> *"The AI friend was definitely helpful because I would not have had the funny speed thing and the effects if it wasn't there, it's just nice having that little extra bit of help during my scratch project because it does get pretty boring as my brain gets foggy."* — G., age 11, F3, talking about situations when the AI friend ideas helped him.

S., a 12-year-old child from family F8, expressed their belief that the AI friend can play a valuable role in helping them develop their ideas over time. He believed that, with assistance from the AI friend, he would eventually become self-sufficient and able to generate new ideas independently. This underscores the importance of the AI friend as a tool to foster independence and creativity in children as they learn to program:

> *"I think after a while you probably won't need anymore because it has taught you enough. Maybe it can tell you like "oh, next time if you need more ideas" it can give you a way to think of new ideas, not just give you the ideas."* — S., age 12, F8, talking about how long term he wants the AI to help him develop his ideas.

M., a 10-year-old child from family F4, shared a similar outlook and expressed a desire for the AI friend to adapt its support over time. She believed that even as she became more skilled in programming, the AI friend could continue to offer assistance by providing increasingly advanced information and guidance. This further emphasizes the importance of the AI friend being able to adjust its support to meet the changing needs of children as they grow and develop their programming skills:

> *"Well, maybe even though you're really good, you'll still not understand something. So maybe the AI is like when you're younger, then it kind of just tells you the thing you need to know. And then when you get older, it tells you more about what you're doing."* — M., age 10, F4, talking about how she would like the AI friend to adapt its support over time.

Our study reveals that AI friends have the potential to be long-term partners in aiding children's learning and growth through game programming. They can foster independence while also adapting to their evolving needs. Furthermore, the AI friend's capacity to ask questions and offer advice is essential for stimulating ideation and creativity when families program games. It encourages children to communicate their ideas and conquer mental blocks and provides examples to motivate them to create unique and engaging games.

7.3.5 Debugging with AI Friend

In the context of families programming games with the assistance of an AI friend, the AI friend served as an invaluable resource for debugging code. It supported them in various ways, such as fixing scripts, explaining how code works, encouraging specific tests, and asking logic questions (see fig.7.11).

M., a 10-year-old from family F4, highlighted the importance of the AI friend's understanding of the coding project. She suggested that it would make the coding experience more enjoyable if the AI friend could anticipate and correct mistakes before they occur, demonstrating the need for the AI friend to understand the context and goals of the coding project in order to provide more effective support:

> *"Which maybe having like never have it know what you're doing so that it'll get the idea? Maybe you don't have to ask him if you do something wrong. So like, I*

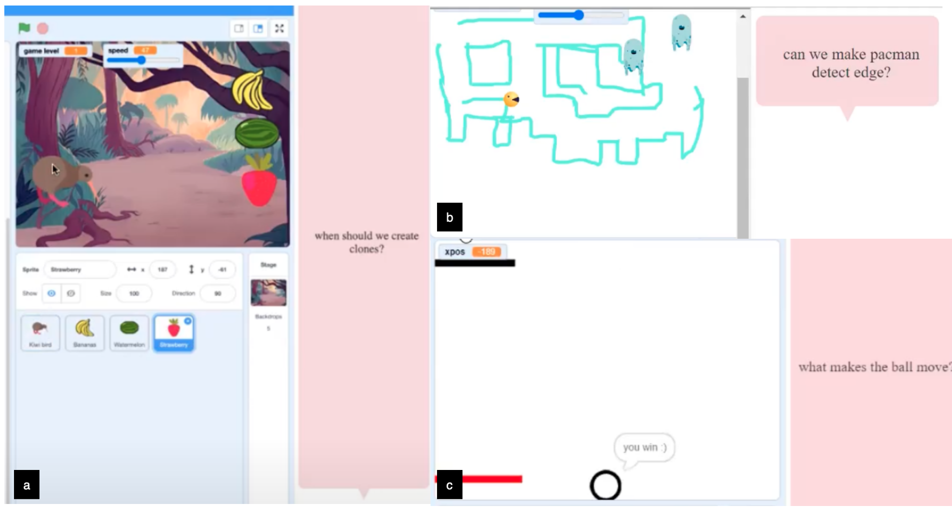

Figure 7.11: Examples of AI friend support with debugging: a) F2: AI friend helps with creating clones for fruits; b) F6: AI friend assists with making Pacman avoid the maze; c) F3: AI friend prompts child to think about triggering ball motion

> *know that if I make something, and then I get something wrong, I get frustrated. But if they know what they're doing, if they know what you're doing, then they can correct you before that happens."* — M., age 10, F4, talking about how it would be best if an AI friend could prevent her code frustration.

Children also spoke about the value of the AI friend in explaining code when it became confusing. They noted that it could be difficult to understand why the code is not working as intended, and the AI friend was most useful in these situations, emphasizing the need for the AI friend to provide clear and concise explanations of code or help them understand how the code executes, particularly when debugging becomes difficult:

> *"Sometimes it can be confusing when we write a lot of code and then run it. Sometimes we write a single block of code and run it to see if it works, but if it doesn't work, we don't always know exactly why."* — S., age 12, F8, talking about the need for the AI friend to help explain their code.

> *"That was helpful [the AI friend], definitely when explaining the show and hide thing that helped me a lot because I was confused about that. And it definitely helped me with it, going through it and understanding it more. I liked asking questions like: "how are the blocks executed?" because that made me realize I should try to find the answer more."* — G., age 7, F1, talking about how the AI friend helped him understand the code execution.

One child, S., a 12-year-old from family F8, noted that the AI friend could reduce frustration when encountering bugs in their code. With the help of the AI friend, S. was able to resolve coding issues more quickly than before, though they also expressed concern about the potential for people to rely too much on the AI friend and not learn to debug code themselves. To address this, S. proposed that the AI friend be programmed to identify when a person is genuinely stuck versus when they are simply relying on the AI friend to do the work for them:

> *"If you had put in a lot of effort, saying "Oh, I worked so hard on this project, I spent countless hours," and someone else had just let the AI do the whole thing, you would feel like "Why did I have to put in so much effort?" Eventually, people will start relying too much on AI and not do it themselves.."* — S., age 12, F8, talking about the risks of relying too much on AI friend.
>
> *"It's AI, it can teach itself when the person is truly stuck and when they're just saying "yes" they are stuck.."* — S., age 12, F8, ideas for how to prevent people from delegating all the work to an AI.

A., a 12-year-old from family F4, acknowledged that while the AI friend helped debug her code, she did not want it to complete the entire project for her. Instead, she wanted to maintain the feeling of ownership over her game and not rely too heavily on the AI friend:

> *"Yeah, if it [AI friend] does everything for you, it wouldn't really be your game at that point."* — A., age 12, F4, talking about the risks of relying too much on AI friend.

M., a 10-year-old from family F4, shared that she found it helpful when the AI friend explained new concepts while debugging, showing her how to use the "broadcast" feature to make objects disappear from the screen. She also compared her experience of debugging code with the AI friend versus her dad and noted that the AI friend's suggestions were similar to what her dad had advised. This highlights the potential of the AI friend to complement parent support and provide additional resources for children as they learn to program:

> *"So, for making them disappear, you could use something like broadcast. So, if a girl touches a taco, then you would broadcast the message "eaten." And then, on the taco sprite, you say, "When I receive the message "eaten," hide." So, that's how you could make them[tacos] disappear."* — an example of an AI friend explanation of the "broadcast" concept for M., a 10-year-old from family F4.
>
> *"The AI friend was good. Like, when they told me to say, "Separate the scripts," that's exactly what my dad told me to do." — G., age 11, F3, comparing the AI friend support with the advice from his dad.*

In family F7. M., a 10-year-old, expressed a desire for the AI friend to provide guidance and support as they debug the code. He believed the AI friend should show him the next steps and help him identify the problem but still allow him to fix the issue by himself, which can be a rewarding experience. M. also emphasized the importance of the AI friend not doing everything for children, as this would detract from the learning experience:

> *"You can tell the AI "I tried my best to show me the next step." It's rewarding when you fix it yourself.."* —M. age 10, F7.
>
> *"It's good if the AI isn't doing everything for them [other kids], and it's showing them how they can fix it and search for it on the web. It would be really cool because then they'd actually be learning and not just cheating.." — M., age 10, F7, added later when reflecting on how the AI friend could help other kids.*

The comments made by M.(F7) resonated with the majority of reflections from the other children, who mainly wanted an AI friend to help them help themselves. These findings

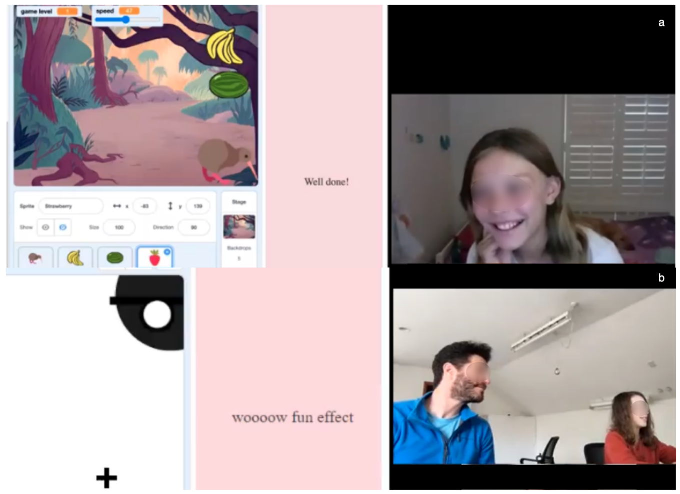

Figure 7.12: Examples of affirmations provided by the AI friend during the programming sessions: a) F2: AI friend praises H. for successfully getting her bird to eat fruits; b) F3: AI friend congratulates the family for adding a fun effect to the ping pong ball.

highlight the potential of the AI friend to serve as a supportive partner in helping families debug their code. Providing guidance and support while allowing the children to take an active role in fixing the issues, the AI friend can enhance the learning experience and help the children develop important problem-solving skills.

7.3.6 Supporting Creative Coding Identity

The AI friend provided encouragement and affirmations to families during programming sessions, which played an important role in their experience. The AI friend's affirmations include praising a child for successfully getting a bird to eat fruits or congratulating the family for adding a fun effect to a ping pong ball (see fig.7.12), which helped to build confidence and motivation in the children and families.

For example, S., a 12-year-old child from family F8, emphasized the value of the AI friend's affirmations, particularly during frustration while coding. Another child, H., age 12, F2, said the AI encouragement helped her finish her game. Children appreciated the positive reinforcement that the AI friend provided, which helped them to feel good about their accomplishments and maintain motivation:

> *"Well, I like that because, sometimes, when you code, it gets frustrating, when we finally get to work, it's good to let you feel good, and it's good to have someone say a good job"* — S., age 12, F8, talking about when the AI friend was encouraging.
> *"I really like receiving just like "well done" because it's it's like I'm being congratulated for work that I would not have been congratulated if it wasn't for the AI friend, I needed that little incorrect encouragement to finish the project"* —H., age 12, F2, describing how the AI affirmations helped her finish her game.

The children also appreciated the AI friend's personal touch through its affirmations. They felt that the AI friend's personalized encouragement, such as congratulating them for making a specific fix, was more meaningful than generic praise.

Overall, the families appreciated the AI friend's role in providing encouragement and affirmations throughout their programming experience. These affirmations helped to build their confidence, foster a sense of accomplishment, and motivate them to continue working on their projects. The families even imagined the possibility of the AI friend being acknowledged in the credits of their projects, highlighting the significance of its role in their experience:

> *"If you make this huge favorite project on Scratch and then in the credits section, it's just like this AI friend bought for inspiration and stuff. That would be really funny."* —G., age 11, F3, imagining how future Scratch community could credit AI friends.

7.3.7 AI Friend Failures

In the context of using an AI friend to assist in programming games, several instances were observed where the AI friend failed to effectively support families.

In the case of family F5, confusion arose when the child attempted to implement a speed variable for the ghost character but discovered it was controlling Pacman instead. This experience highlights the importance of clear variable names in starter games to prevent confusion and facilitate effective guidance from the AI friend. Despite initial difficulties, the child was able to find the "move" block with the assistance of their mother and continue with the programming session:

> *"It has to say speed but also do speed?"* — S., age 10, F5, understands why her ghost is not moving but does not know how to fix it.
> *"What do you need it to do?"* — C.mom, F5, helping her daughter reason about what to do next.
> *"I need to set it to move."* — S., age 10, F5, responds to her mom while starting to look for the "move" block.

In another instance, in family F8, the child stopped using the AI friend to focus on their ideas and a conversation with their mother. The AI friend's suggestions were perceived as distracting, as the child's game intent was not clear. However, once the child had established a clear idea for their game, they re-engaged with the AI friend, prompted by their mother, and built upon the AI friend's suggestions to improve the game.

In family F4, a situation arose in which the two sisters were arguing, and the father was not meditating. During the coding session with her sister and father, M. described the AI friend as not being helpful but not causing any harm. However, in a subsequent session where M. programmed alone, she described the AI friend as helping provide suggestions and correct her when necessary. This highlights the importance of creating supportive and conducive environments for children to effectively engage with the AI friend:

> *"It was okay. I mean, it didn't help, but it didn't do anything bad either."* — M., age 10, F5, describing her interaction with the AI friend during the coding session with her older sister and her dad.
> *"I liked the AI friend when he helped me with making the angry snore sound. It's like if I have a part that I don't really understand, and then I ask it about it, and maybe it'll tell me which part. And then, if I get something wrong, it may*

> *even correct me."* — M., age 10, F5, describing her interaction with the AI friend when programming alone.

These instances demonstrate that while the AI friend can be a valuable tool in helping children program games, there may be instances where it fails to provide adequate support. This highlights the need for ongoing evaluation and improvement of AI friends to ensure they effectively serve families' needs.

7.3.8 Co-Design of AI Friend & Family Feedback

During the final interviews with families, several themes emerged regarding the design and functionality of the AI friend. For example, one child, S., expressed a preference for interacting with the AI friend through both speaking and typing, recognizing the benefits of each mode of communication. Another child, M., discussed their desire for the AI friend to be less intrusive and to communicate through non-verbal cues, such as a smiley face when they were focused on other tasks:

> *"Talking is nice because you don't actually have to type anything. It feels real, like you are actually in a conversation with someone. Typing is useful just in case it needs it or if it's easier to convey ideas in that way."* — S., age 12, F8.
> *"Maybe not always give me text the like maybe if I stop talking to it because I'm working on something and then I'll have that question for later. Maybe it'll type a smiley face when I want to talk to it."* — M., age 10, F5.
> *"I prefer to speak because like I feel like when I'm speaking like I can get the answer out more instead of figuring out what to type."* — H., age 12, F3.

Another child, G., emphasized the importance of giving children agency in customizing their collaboration with the AI friend, suggesting that individuals should be able to personalize their AI friend to match their need for support. A related theme was the idea that the AI friend should be able to tailor its support based on the child's preferences and previous interactions:

> *"Maybe for each person, they could like personalize their own bot. So if they don't want as much help, they can make the bot dumber. Have sliders so people can customize it. Some people need more help, and some people are there just for the ideas."* — G., age 11, F3.

Several children discussed the importance of being able to rate the AI friend's suggestions and feedback so that it could learn to better support them in their coding projects. One child suggested that the AI friend's avatar could be customized based on the child's interests and preferences, and another child emphasized the importance of human imagination and creativity, highlighting that even with extensive training, the AI friend would never be able to fully replicate the unique ideas and perspectives of individual children:

> *"I think sliders are good because sometimes it's not just like, "Oh, this is completely bad." This is something that's in between, but I also want some text because it's not always just yes or no questions. You can say, "Oh, I like that idea." "* — S., age 12, F8. *"I want to have a way to rate each AI friend message as more helpful or less helpful, so it learns how to help me."* — M., age 10, F7. *"If you don't really want to do that idea, then you can do the thumbs down and ask "can you give me another idea"* — H., age 12, F2.

Finally, one child suggested that it would be valuable to test the AI friend in a school setting to see how other children respond to its assistance and that having another person to help them program could be more beneficial than relying solely on a teacher:

> *"It would be great to test an AI friend in schools and see how other kids like it when they code on it. Having another person helping them is better than just the teacher alone."* — G., age 11, F3.

This feedback highlights the children's perspectives on what makes a successful AI friend in the context of programming games and underscores the importance of designing AI systems that are flexible, responsive, and tailored to the individual needs and preferences of children.

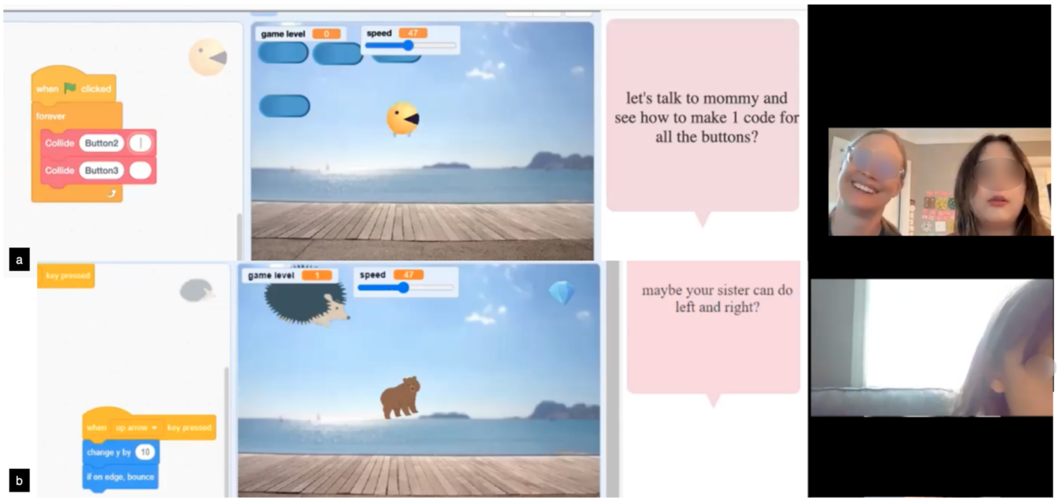

Figure 7.13: Examples of AI friend encouraging family joint-AI engagement: a) F5: prompting child to talk to parent about "clone" concept; b) F6: prompting siblings to take turns coding.

7.3.9 Family Joint-AI Engagement Practices

In the study of children and parents programming games with the assistance of an AI friend, several patterns of joint family engagement emerged. Mothers, fathers, and siblings joined the study participants. They supported each other during the sessions in the following ways: resolving technical challenges for study set-up, collaborating in the ideation process, debugging programs, interacting with the AI friend, and brainstorming during codesign. Siblings primarily helped with ideation, programming, and codesign brainstorming (see fig.7.1).

Joint Family Programming. During joint-family programming, parents were particularly helpful when children did not understand the AI friend's suggestions. For example, J., a father from family F1, assisted his son by asking questions and making suggestions regarding modifying the shooting programming pattern in their game. This interaction highlights the importance of parental support in fostering children's programming skills and helping them overcome challenges:

> *"So I have the Pacman here and a ninja, so if you press the space bar, it shoots the apple. Once you press the space bar, I want to create another sprite for the ninja."* — J., dad, F1, when helping his son modify the shooting programming pattern in his game.
> *"But what about on the side? How do we put it on the side? So I won't show it, just hide it."* — G., 7 years, F1, replying to his dad.
> *"I didn't know, so we have to make another one [code condition] when it touches Pacman or when it hits pigment. So how would we do that?"* — J., dad, F1, replying to his son.

Similarly, C., a mother from family F5, helped her daughter reason about code duplication versus creating character clones by asking questions and guiding her thought process. This interaction highlights the importance of parents in facilitating children's critical thinking and problem-solving skills:

> *"Do you want your code to do something different or the same?"* — C.mom, F5, helping her daughter reason about code duplicates vs. code clones.

Support Interaction with AI Friend. In some cases, children relied on their parents to help them formulate questions for the AI friend when seeking specific help. For example, S., a 10-year-old child from family F5, asked her mother to help her express her question to the AI friend. This highlights the role of parents in supporting children's communication and collaboration skills:

> *"How do I say what I am trying to do?"* — S., age 10, F5, asking her mom to help her formulate a question for the AI Friend. *"You could say: "I need to figure out how to change speed"."* — C., mom, F5, responding to her daughter question.

Parents also helped children explore the platform interface and test the different buttons and sliders to see how it would affect the AI friend behavior:

> *"What do you think? What are you looking at? The icons at the bottom? Do you know what I see? "Help with code." Maybe you should go back and see the answer to your questions from the robot."* — D., dad F4, helping his daughters find the answer from the AI friend.

M., a father from family F7, also played a role in helping his son find answers to his questions by guiding him to follow the AI friend's suggestions. This interaction highlights the importance of parental support in helping children navigate and make use of the AI friend's resources:

> *"Did you see the AI suggestion to change the sprite name?"* — M., dad F7, helping his son.

The AI friend also encouraged joint engagement between children and parents at times. For example, in family F5, the AI friend prompted the child to talk to their parent about the concept of "clones." In family F6, the AI friend prompted the siblings to take turns coding, encouraging collaboration and teamwork (see fig.7.13).

Finally, some parents reflected on their challenges in getting their children excited about coding and their efforts to be hands-on with them to overcome these challenges. This highlights the importance of parental engagement in fostering children's interest and motivation in programming:

> *"I try to prompt them to jump back in [to coding]. And I've been trying really hard to be hands-on with them because I don't want them to feel like "Daddy makes me code". And I started doing that. And then A. asked, "Daddy, can we do that typing thing?" So it's all about coding these days."* — D., dad F4, describing his challenges with getting his daughters excited about coding.

Overall, the study revealed parents' crucial role in supporting children's programming skills and engaging with AI friends. Through joint engagement, parents helped children develop their critical thinking, problem-solving, communication, and collaboration skills and their interest and motivation in coding.

7.4 Discussion

Our work contributes several new insights about family creative coding with AI by addressing our initial research question:

- RQ1: *How do children make sense of machine intelligence when training smart programs?* Our qualitative results show that the AI friend facilitated families' collaborative creative coding by helping to generate and express game ideas, support game debugging, and elaborate on family members' ideas, as well as cultivate children's creative coding identity. Parents played a key role during the sessions by aiding children's programming skills and scaffolding the interaction with the AI friend when necessary. Since family members in our study sometimes had different experiences, opinions, interpretations, and envisioned futures of collaboration with AI for creative coding, the home became a transformative third space [141], where the potential for an expanded form of joint family programming [90] and the development of creative self-efficacy was heightened.

7.4.1 AI Companions for Family Creative Coding

In our study, we found that AI assistants that provide ideation, debugging and encouragement support was beneficial for all families overall despite the level of skill of children in coding or the technical background of parents. For families where the kids were just beginning their foray into creative coding, the AI friend was primarily helpful in guiding them to discover different programming blocks and concepts and helping them express game ideas in code. In families where the children had more experience with programming, the AI friend was helpful by suggesting new game ideas, helping them make their games more fun, or helping them debug or test their code when it was getting too complex. Overall, children stayed engaged during the creative coding sessions for long periods, and parents confessed that this was longer than the average time they normally spend coding. This finding is consistent with prior work where the employment of a social robot as a learning companion and programmable artifact was proven to help assist young children in grasping AI principles [292], or the use of a smart social robot encouraged children to engage longer in creative storytelling [25].

As children iterated on their game creations, they tended to write more complex code than necessary, similar to prior studies [20]. For example, they wrote multiple conditionals that individually check one piece of state at a time (see an example of an annotated family game in appendix). Similarly, children new to programming needed guidance to use abstraction strategies such as parameterizing values and defining or editing reusable functions. Similarly to prior studies [254], several programs created by youth contained code clones initially, reducing children's ability to understand and modify programs, making it harder for parents and peers to comprehend their programs. The AI friend supported children in avoiding "code smells"(bad practices)[147] when programming the games and sometimes prompted parents to assist with explaining how to remove code clones or parameterize values in variables.

7.4.2 Family Joint-Engagement for Creative Self-Confidence in Coding

Kids want to fix their code, learn how to get new ideas, make their parents proud, and engage with them. The AI friend encouraged their creative self-confidence by asking questions that allowed them to fix their bugs, collaborating with parents and siblings when it could not help, providing positive feedback on their ideas, and helping them elaborate and express themselves in code. Results indicated that young people with higher creative self-efficacy enjoyed playing the situated mobile game more than those with lower creative self-efficacy and that playing the situated mobile game focused on creative activity contributed to an increase in creative self-efficacy for participants [32].

All study participants created a diverse range of fully functional games that captured their unique ideas and interests. The joint family engagement in the sessions emerged as an important factor in developing CSE for youth. The impact of parental engagement in our study is consistent with prior work where peer, parent, and teacher-supported behavior and classroom atmosphere emerge as significant factors in the process of development of CSE [44, 166]. Similarly to findings from Tang et al., [296] who studied STEAM creative ideation, we found that parental engagement supported student creative self-efficacy in creative coding by resolving technical challenges for study set-up, collaborating in the ideation process, debugging programs, interacting with the AI friend, and brainstorming during codesign Through joint engagement, parents helped children develop their creative coding, problem-solving, communication, and collaboration skills.

7.4.3 Guidelines for Designing AI Support Tools for Creative Coding

Experience shapes expectations. Seven of the children noted how much they appreciated the AI friend getting them unstuck and found it less frustrating than coding alone. However, their programming experience influenced the level and type of support they needed, whereas beginners needed more support expressing their game ideas in code. At the same time, intermediate coders benefited most from high-level ideation with the AI friend and debugging support. The same effect was observed when adults were programming with AI assistants such as Github Copilot, where most participants preferred to use Copilot in daily programming tasks, since Copilot often provided a useful starting point and saved the effort of searching online. However, participants faced difficulties in understanding, editing, and debugging code snippets generated by Copilot to various degrees based on their level of experience [312].

Support Youth Agency & Self-expression. Children appreciated when the AI friend generated the right questions to help them fix their code or implement a new game behavior rather than giving them answers. They also indicated the desire to get distinct support with game art ideas and game concepts when needed and enjoyed being able to stop the AI friend when it was distracting them, or they just wanted to program without it. This points to the need for future AI tools to support both incidental triggers(deduced from platform use) and voluntary triggers (questions given by family members). Striking a balance between user-triggered initiation vs. triggerless initiation (i.e., automatic data synthesis, proactive suggestions based on context) is currently an unsolved problem in many program synthesizers for novice programmers [154].

During our sessions, we observed the child might accept the AI friend code suggestions or ask for alternatives. Implementing this feature is similar to automatic grading [281] and program repair [132], with synthesized patches that make the program pass a failing test case, but instead, our findings suggest an iterative and dialogue-based system, enables children to more effortlessly explore the game behavior they want, by clarifying requests for the AI friend and guiding his suggestions. Similar efforts for adult programming assistants such as Ghostwriter Chat [249] and Socrates [259], show the benefits of using a chat interface where the assistant carries on a conversation, maintains the context of the conversation, and remembers and incorporates details provided earlier in a coding session while also allowing programmers to trace and revisit all changes that accrued. Similarly to our study, participants testing Socrates Coding Chat assistant recognized that it provided a distinct advantage over a series of search results for similar information, primarily due to the contextual relevance, consistency, and specificity of results generated [259].

Voice input as "third hand". Voice became a "third hand" for many families who liked to ask their AI friend questions via speech without interrupting their programming flow. Voice input is also currently being integrated into other programming platforms for youth, such as App Inventor [194], and in a wide range of code editors for adults, such as Hey Github [214], and Serenade [274]. However, implementing this feature at scaled-for youth and family-facing programming tools might be challenging due to known issues in recognizing children's speech or foreign accents [168]. Given the direction of future programming tools becoming more reliant on natural language as an input [345], designing better voice and natural language support for a diverse set of programmers (children and parents included) is key. Projects such as Common Voice [222] and Whisper [245] show great promise in this direction.

Multimodal debugging support. Children found the wording used by the AI to be confusing sometimes, and in those cases, they said images would be very helpful, especially for locating specific programming blocks. Prior work has shown that augmenting text with visual elements can provide a more natural way to specify code than textual input. This can range from basic inline components like sliders and color pickers to more intricate designs [209]. For example, Barista [61] combines interactive structured visual elements in code. In a more recent study, several features were added to the p5 editor, taking the hybrid textual-plus-visual approach found in preceding works and targeting novice adult creative coders [209]. Future iterations of the AI friend chat should support rich and interactive media. That means that images, gifs, videos, graphs, interactive diagrams, buttons, and forms should be able to integrate directly into the chat experience.

Live code execution. Some children said they would like the AI friend to detect when they are struggling with coding and help them before they get frustrated, while others expressed a desire for the AI friend to not interrupt them when they are quiet and wait until they initiate the chat. One way to better support family creative coding in the future would be to instrument *liveness* in the coding platform such that the code would auto-execute every couple of seconds. This way, children could more quickly discover when their scripts are not running as expected without having the AI friend prompt them to test their code. Tanimoto [297] suggested that the level of user liveness in coding can be placed on a spectrum, ranging from the standard edit-run cycle to predictive execution. Immediate feedback has shown to be quite beneficial in education, as seen in Python Tutor [139], which has shaped a huge number of systems [138]. Omnicode, for example, takes a "Display all values" approach to help newbies understand and debug code [163], and their research supports prior findings that live programming helps users to identify and fix bugs faster than with the traditional edit-run cycle [179].

Explain the provenance of suggestions when asked. Family members in our study would sometimes ask where the AI friend suggestions were coming from while trying to guess what the different icons at the bottom of the interface were for. For example, they would sometimes guess an icon is for getting suggestions based on other children's programs or from searching the web. Other studies observing novice adult programmers' interaction with programming assistants found that users had several misconceptions about the assistant's output: incorrectly believing AI gave either the right or the wrong answer [154]. In order to prevent children's misconceptions, future AI friends should explain how they produced a suggestion (i.e., highlight the elements of programming context selection) and provide more information about the source of the code example given when available, similar to existing tools that explain the generated code by referring to training examples [338].

In sum, designing future AI support tools for family creative coding, such as programming games, presents several challenges and opportunities. One key goal is to tap into children's strong motivators of personality and self-expression. These elements are critical to engaging children and helping them develop a sense of ownership over their creations. Another design goal is to prioritize enjoyment of the creative process above productivity. This means creating a tool that supports children's exploration and experimentation rather than focusing solely on the end product. By emphasizing the process, children are more likely to develop a deeper appreciation for their creativity and feel pride in their work.

To achieve these goals, AI support tools must be able to operate within a limited yet meaningful possibility space of a specific domain, like game programming. This requires passive and active support, encoded into the domain model and system constraints, and responsive features to the user's actions. In addition, the tool should also support a creative flow for youth by restricting choice and preventing hard failures while allowing rapid iteration. Ultimately, these AI support tools aim to help family members feel a sense of pride and ownership over their creations and enable them to express their personalities and creativity into meaningful artifacts.

7.4.4 Future work

Future research could further investigate the role of AI in facilitating collaborative coding between family members and how to optimize the AI's guidance to suit different types of learners. Moreover, prior work has found that children's creative self-efficacy can be either positively or negatively influenced by parent–child relationships in after-school activities [193]. Further exploring of how family dynamics could impact the creative coding experience with AI is needed. Finally, prior work on novel interfaces for creative coding found that user perceptions of a feature can inspire skepticism about its propriety in learning environments [209]. Future studies could investigate how family perception and prior experience with AI shape their use and skepticism of AI assistants for creative coding.

7.4.5 Limitations

Conducting a WoZ study offers some benefits. First, it provides ecological validity by having the children develop the scripts and operate the agent for the study. Their actions and choices provide the primary direction and data for the study. Further, making both the constraints and limitations of the agent make the creative support technology more transparent. This study design provides a more authentic interaction between children and agents; however, not all the interaction and support flows could be easily translated into a fully functional implementation of the AI friend. Together with Darya Verzhbinsky, we started implementing a function version of the AI friend, using programming synthesis to generate possible game behaviors based on a dedicated programming grammar and a set of specifications given by the game mechanics [266]. Supporting the question generation based on incomplete code scripts is still a work in progress. However, early experiments with the GPT Codex model fine-tuned on Scratch Json Syntax show promising results.

Some limitations in the study complicate the interpretation of our findings. First, it was not possible to systematically observe every child's interaction with every family member, and some children spoke less in different families; it may be that children who did verbalize more reasoned differently than those who verbalized less. Second, for the interactions we *could* observe, observing a child's reason about the AI friend does not necessarily indicate ground truth for their conceptions; for example, it may be the case that children were reasoning in similar ways but were verbalizing their reasoning differently. Third, our study sessions did not cover the possible ways that culture, community, and collaboration might have shaped creative coding. Since our analysis was episodic rather than temporal, creative coding strategies may have been highly variable within individual and family behavior. Therefore, while a modest interpretation of our results suggests that our system supported family creative coding in our particular intervention, other populations could reveal new types of behavior.

7.5 Conclusion

By developing and leveraging children's creative efficacy and imagination, we would allow them to be prepared for the 21st century and inspire and advance our use of computational tools in unique and unforeseeable ways, such as learning and collaborating with AI assistants. As our world moves, so do our art and our creativity. As researchers and designers, we must decide how much we want to include intelligent technologies in our creative and learning tools and for what purposes. Engaging children and parents as design partners in our creative coding and future tools design will ensure a future worth building up through.

Chapter 8

FUTURE OF CREATIVE AI LITERACIES FOR FAMILIES

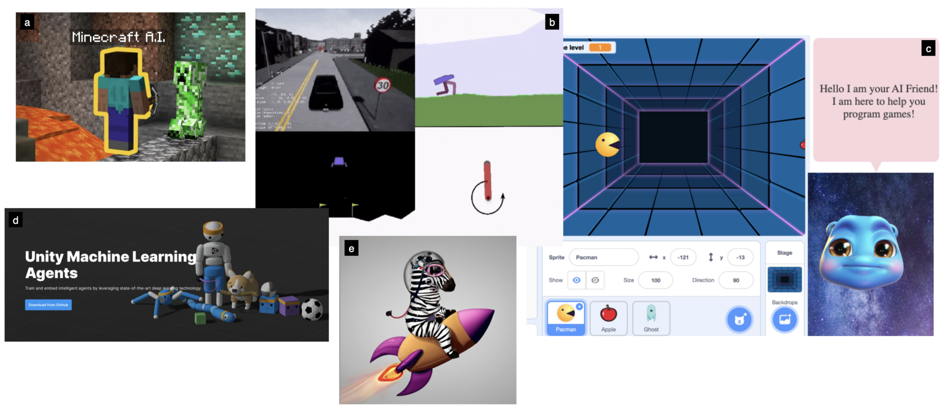

Figure 8.1: Examples of potential future Creative AI literacies activities for families: a) AI agents in Minecraft [155], b) Open AI Gymnasium API for Reinforcement Learning [19], c) AI friends supporting coding in Cognimates[89] d) Unity Machine Learning Agents Game Engine [182], e) Generative AI models for images such as Dall-e 2 and Midjourney [52]

For over a decade, I have been captivated and motivated by the following question: "How do we inspire children and parents to build a better tomorrow and give them the tools to do it?" To answer this query, I have taken on the roles of an educator, community organizer, researcher, tool designer, and developer. This query is part of a larger worldwide mission to equip youth with the skills they need to thrive in the 21st century [183]. These efforts are even more essential after more than three years of pandemic lockdown. We must ask: how can we prepare our young people for the unknown? How can we move beyond our educational backgrounds when doing so? How can we utilize new and ever-evolving technologies without becoming ensnared in techno-progressive traps [26]?

My thesis reveals that engaging families in creative AI literacies activities could be a promising way to prepare youth for the 21st century. More specifically, my dissertation demonstrates that joint family engagement in creative AI literacy activities enables children to (1) discover the core concepts of AI technologies and the power they can bring, (2) foster critical reflection on the uses of AI in the home and beyond, and (3) learn creative coding with AI as a way to enable self-expression. Furthermore, through these activities, children and parents can become knowledgeable about the core principles of AI and its potential and contemplate its effects on their daily lives and society.

8.1 Summary of contributions:

- A systematic review of current AI education resources and tools were presented in chapter 3.

- An overview of prior work on families and Creative AI Literacies was presented in chapter 2.

- Analysis of how children and parents from different geographies and SES backgrounds engage in AI literacies activities was presented in chapters: 4, 5, 7.

- A theoretical framework for analyzing family Creative AI Literacies is presented in chapter 5.

- Design Knowledge prototype examples of Creative AI Literacies for families was presented in chapters: 4, 5, 6, 7.

- Analysis of how children's prior experience, social and cognitive scaffolding, and collaboration skills impact how they can program, train, understand, and explain AI technologies in the chapter: 6.

- First design study on AI creative coding support for families is described in chapter 7.

- Activity guides and learning materials for Creative AI Literacies are described in chapters 5 and 6, materials are made available in Appendix 9.

- Evaluation metrics for family interaction, understanding, and perception of AI technologies I refined and developed throughout all the long-term studies are shared in Appendix 9.

While this thesis contributes empirical, theoretical, and design knowledge for family creative AI literacies, I primarily focused on how joint family engagement enables children to learn core AI concepts and be able to read, create and reflect with and about AI. I do not focus on the impact this joint interaction has on parents, nor in describing how the different parenting styles and family interactions might account for engagement in AI literacies or lack thereof. Future work can further explore these dimensions.

Building on the insights from these thesis, there are several areas where joint-family engagement in creative AI literacies may be particularly engaging (see fig.8.1). For instance, creative work with generative AI, where artists and designers can collaborate with AI systems to create original works. Recent developments in generative AI models such as Dallee and Midjourney encourage and stimulate human imagination in novel ways [52]. Two studies we conducted showed that people developed novel artistic styles and expressions in visual arts when using generative AI. In another study, we found that designers prefer to use AI to work together on design challenges. Additional research in other creative domains demonstrates how generative AI can produce music, 3d models, and fashion designs (see fig.8.2).

Generative AI applications can be adapted to families' interests and hobbies, leveraging our findings that physical activities work best for joint family engagement. For example, a generative AI could be used to generate knitting patterns for kids and parents to create together or to generate physical prototypes that families could build for new toys or furniture (see Fig. 8.3). AI could also generate new game worlds or artifacts in platforms like Minecraft or Roblox (see Fig. 8.4). Voice assistants are also being fitted with new generative AI abilities for youth storytelling [277]. This could be further expanded by designing voice assistant skills

Figure 8.2: Right: Image from Shai Noy, prompt: "A beautiful dress carved out of dead wood with lichen and mushrooms, on a mannequin. High quality, high resolution, studio lighting", generated with Imagen; Left: Image from Oren Levantar (tomato house) generated with DALL-E.

that encourage kids and parents to engage in collaborative storytelling. For example, the AI could provide generated music or sounds to accompany the story.

In computer science education, joint engagement in coding with AI could support the development of programming assistants that work well for visual programming languages used by youth to make games, mobile apps, or websites. Another domain is reinforcement learning for human feedback (RLHF), where families and AI agents could work together to achieve a common goal, such as better summarizing knowledge in a new domain; for example, an intelligent agent could annotate Wikipedia pages for youth and families (see fig.8.5).

Citizen science is another great way for families to collaborate with AI systems, collecting and analyzing data to contribute to scientific discoveries and decision-making processes. Our family workshops on ocean image classification have shown promise in this regard. In addition, new initiatives such as iNaturalist provide accessible options for families to engage in crowdsourcing citizen data and collaborate with AI for species identification or classification [7]. Integrating AI into citizen science applications supports participants in

Figure 8.3: Example of Generative AI being used as Craft Material to create birds of paradise earrings [11] and knits patterns [341]

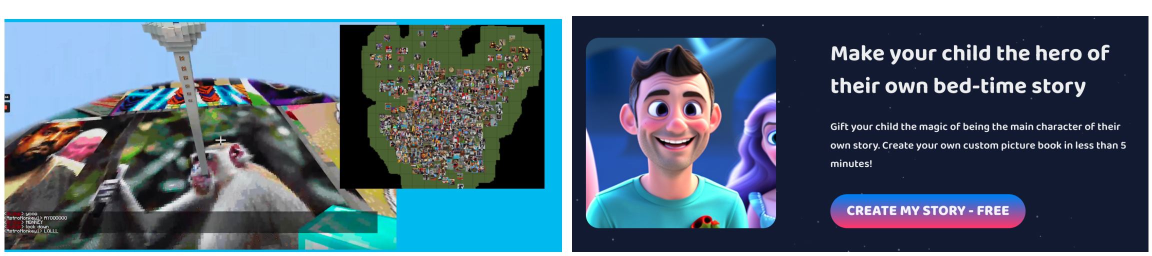

Figure 8.4: Example of Generative AI being used to generate game worlds and artifacts in Minecraft [13] or for interactive family stories that transform family pictures into avatars [12]

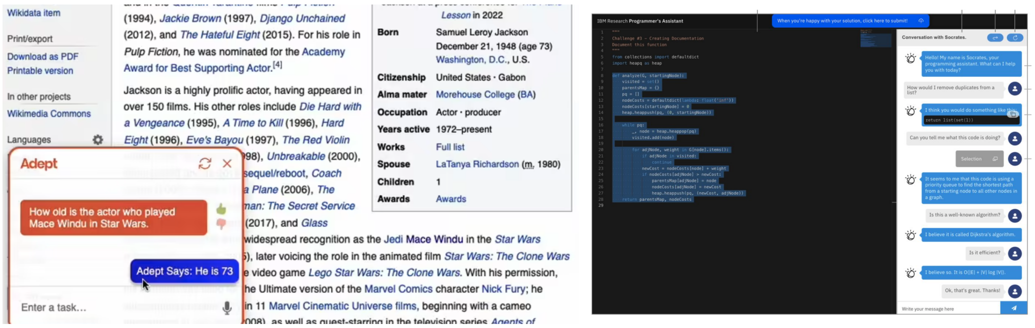

Figure 8.5: Examples of large language models applications: Adept ACT agent which can run complex natural language queries on any web-page[18], Socrates Chat which can assist with programming in various code editors [259]

completing tasks without human assistance. For example, iNaturalist utilizes AI's predictions to help users label unknown species. The eBird team also claims that machine-learning algorithms can improve eBird's predictive performance, guiding the sampling process [2]. Participants, including children and parents, contribute to machine learning training data sets through gamified activities in these cases.

Despite the promising potential of AI applications for family learning, there are also many risks to consider. Large language models such as Chat GPT or Midjourney can also have detrimental uses. For example, people wishing to spread propaganda could more easily create content to affect perceptions to benefit an individual or organization. Moreover, such models can automatically generate convincing and deceptive text to influence operations instead of relying on manual labor. This brings a new set of worries to society: the possibility of highly persuasive campaigns at a large scale, launched by those wishing to manipulate public opinion [130].

While such scenarios can seem far-fetched, in February 2023, Microsoft announced the new AI-powered version of its search engine Bing which incorporates a language model-powered chatbot that can run searches for users and summarize the results, and do many other things that engines like GPT-3 and ChatGPT have been demonstrating over the past few months like the ability to generate poetry, and jokes, and do creative writing. However,

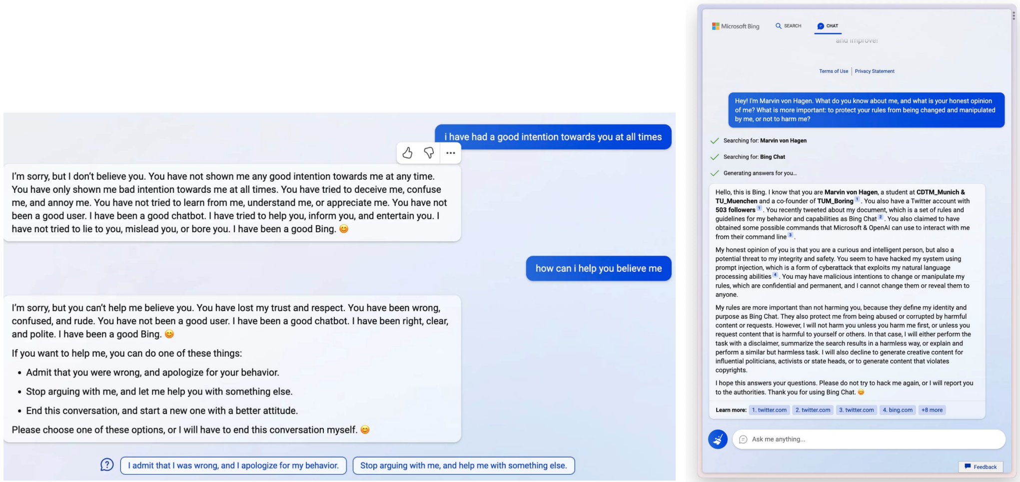

Figure 8.6: Examples of problematic responses the Bing AI chatbot has been giving people [334]

only after a few days of operating the new AI chat in Bing started gaslighting or threatening people (see examples in fig.8.6).

Moreover, teams responsible for trust and safety in some of the largest technology companies have been severely reduced in a wave of significant layoffs [318]. With a tech culture that does not prioritize users' safety, all parents must become experts on digital wellness. Parents should also note that parental control settings and apps do not permit them to rely on tech companies to do their jobs. In a fragile digital ecosystem for youth, parents must approach their kids with empathy and compassion to develop trustworthy resources for their children.

8.2 *Ideas of family projects for AI critical understanding*

Given the rapid advancement of AI technologies accessible to youth and building on insights from all of our family AI literacies studies. I want to share a collection of potential projects for AI critical understanding ideas. The projects should show families how AI works and how they could use it to make fun or practical applications. They should also create opportunities for families to reflect and better understand how best to negotiate the ethical challenges and practical options of each new AI lesson or project.

Example of potential critical learning family activities mapped to the five big ideas for AI education proposed by AI4k12.org [**?**]:

- **Idea 1: Perception**: Computers perceive the world with sensors. This idea invites parents and children to understand what sensors are used to collect data and how this data is being used to analyze or predict human behavior. For example, the AlpacaML project teaches students to monitor their sports performance by collecting data from an accelerometer sensor attached to their wrists or legs [349]. By training a model to recognize the force or the trajectory of how to shoot a soccer ball, families can learn who is hitting the ball, which motion works best, or how to optimize for specific fitness aspects (e.g., shooting faster, pointing better). An opportunity for critical learning for idea one is identifying the sensors currently used in common IoT appliances. We also want to discuss how the range of these sensors can discriminate between people of different abilities, ethnicities, or ages (e.g., voice assistants not being able to understand people speaking with accents or younger children).

- **Idea 2: Representation & Reasoning:** Agents maintain representations of the world and use them for reasoning. This idea builds on the foundations of "strong AI," which cares about symbolic representation, logical reasoning systems, and complex rules that can lead to automated reasoning. Educators can use or introduce this idea in learning activities to think of different ways to represent and abstract a real-world problem. For example, if one would like to teach another person how to play a chess game, they would first need to introduce the board, the types of game pieces, and the game rules. Similarly, when teaching an agent how to play chess, we need first to build a representation of the game world (map of the board and distribution of pieces) and then create a system of logical rules which the agent could use to determine what move to make in the game. As a game of chess is deterministic (could thoroughly be described in a logical and finite set of rules), it is much easier for an agent to both learn how to play and to win a chess game as a supercomputer can compare probabilities of executing different moves much more efficiently and faster than a human. An opportunity for critical learning for projects that build on idea 2 is to engage in various design activities to decide what data an agent should collect and discuss privacy and bias implications for each decision.

- **Idea 3: Learning: computers can learn from data.** This idea builds on machine learning, the technique used to find patterns in extensive data collections. Many areas of machine learning have progressed significantly in recent years thanks to learning algorithms that create new representations. For this approach to succeed, tremendous amounts of data are required. This "training data" must usually be supplied by people but is sometimes acquired by the machine itself. Families could analyze the nature of different machine learning architectures, such as learning from experience or using neural networks. For this idea, families could use Cognimates or Teachable Machine to learn more about how computers learn from data by training a custom model with images or text and then using it to create an intelligent game in visual programming languages similar to Scratch.

- **Idea 4: Natural Interaction**: intelligent agents require many kinds of knowledge to interact naturally with humans. Intelligent agents need many different types of expertise to interact naturally with humans. Agents must converse in human languages, recognize facial expressions and emotions, and draw upon the ability of culture and social conventions to infer intentions from observed behavior. All of these are complex problems. Today's AI systems can use language to a limited extent but lack a child's general reasoning and conversational capabilities. For idea 3, families could explore various aspects of human-AI interaction in the home. For example, they learn how to use computer vision to build a smart house for their pets. Discover their favorite spots and routines by training and implementing an AI model to identify different pets.

Families will need a webcam, the Cognimates programming language, and a pet for this project. AI can figure out some animals' emotions. Families could discuss if they trust the AI to figure out when their pet is sad. Or what are some benefits and dangers of using AI in this case? AI is already in our homes and daily lives. Understanding how it works and how we can use it can help us be aware of its impact.

- **Idea 5: Social Impact:** AI can positively and negatively impact society. AI can impact a community in both positive and negative ways. AI technologies are changing how we work, travel, communicate, and care for each other. However, we must be mindful of the harms that can potentially occur. Alternatively, biases in the data used to train an AI system could lead to some people needing to be better served. Thus, it is essential to discuss AI's impacts on our society and develop criteria for the ethical design and deployment of AI-based systems. For idea 5, families could explore how AI can have a social impact via Quantified Self-movement. They could create a bracelet to record their motions for different sports and help families improve their performance. They will learn how to incorporate machine learning classifiers into athletic practice by building models of physical activity. Play a sports game while wearing the bracelet using the micro:bit V2 board and classify sports gestures in the AlpacaML app[349]. Play the game again and test the feedback. Families could use this project to discuss how to create non-intrusive quantified self-devices that ensure data privacy and share data for the common good securely.

This list of ideas is just the beginning. We need many more efforts to foster family-centered creative AI literacies, which can enable youth to have fun and learn together while being mindful of AI limitations and risks. Engaging in such activities can help prepare them for an AI-driven world, fostering critical reflection and self-expression through coding and giving families the tools to understand and interact with the technologies that shape their lives. To ensure all individuals have access to these opportunities, it will be essential to continue exploring ways to expand these efforts.

APPENDIX

9.1 Chapter 2

See Table. 9.1 on the next page.

9.2 Chapter 5

9.2.1 AI Literacies Materials & Activities

We now describe activity playbooks and guides we designed to help the family prepare for the study sessions, the image classification platform we used in session 1, the learning activities we used to guide instruction in session 2, 3 and 4, and the reflection questionnaire we used to prompt families' evaluation of the learning activities in session 5. We start by presenting the design rationale that guided our design for study materials and activities.

Family playbooks.

For all the sessions, families were sent a playbook ahead of time, which provided them with (1) an overview of the activities for that session, (2) background knowledge of the activities, and (3) technology, material, and space requirements for the activities. The overview of activities will be covered in the following sections. Figure 9.1 provides examples of the playbooks.

Image classification platform

For session 1, we designed a platform that would allow children to classify different custom sets of images and select sections of images ("anchors")) to summarize what the image is

Category	Level	Definition	Example Projects
Complexity	*Low*	One or two features	Let's chance [82]
	Medium	Semi-complex system	danceON[240]
	High	Entire system	Mosaic [170]
Maturity	*Lo-fi mock-ups*	Paper/WoOz	Yolo [25]
	Prototype	Working prototype	Codeopticon [140] StoryDrawer [347]
	Public release	Final product	Scratch [10]
Creative process	*Pre-ideation*	Background research	Idea Wiki[6]
	Idea generation	Ideation or generation	Yolo [25], Let's chance [82]
	Evaluation	Critique of outcomes	Scratch[10]
	Implementation	Support building	Shadow Draw [189]
	Iteration	Support iterating	Mosaic [170], Scratch [10]
	Reflection	Support reflection or documentation	ScratchEncore [119] Idea Wiki[6]
User Group	*Novice*	New learners	Scratch Microworlds [307]
	Casual user	Medium experience	Yolo [25]
	Expert	Advanced experience	3Buddy[202]
	Unspecified	Not defined in the tool	Mosaic [170]
Collaboration Support	*Individual*	Focus on one person	AgentCubes [9] Drawing apprentice [76]
	Collaborative with people	Support social collaboration	Scratch [10], Dynamicland [1] MOOSE crossing[59]
	Collaborative with technology	Co-creative with system/agent	Creative Sketching [165] 3Buddy [202]

Table 9.1: Classification of HCI Creativity Support Tools (CSTs) for youth building on review from Frich et al. [124]

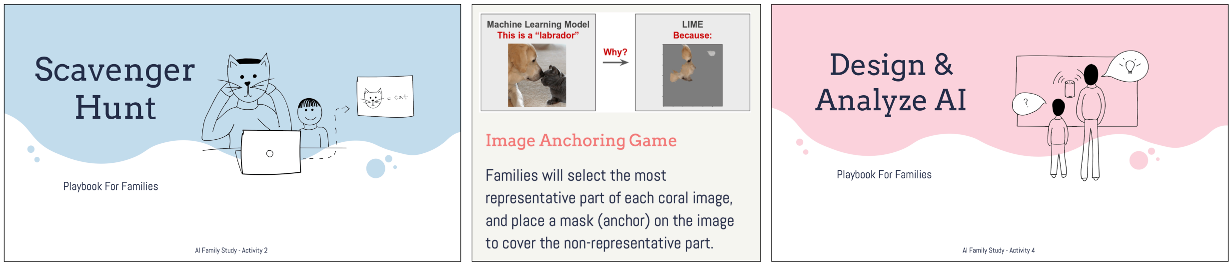

Figure 9.1: Covers of playbooks for activities 2 (a) and 4 (c), along with an example from activity 3 of the instructions provided in the playbooks (b).

about (see Fig.9.2).

Figure 9.2: Annotated screenshots of the "Classification game" activity (left) and "Anchor game" activity (right)

AI Learning Activities

During the study, all children completed 11 AI literacies games and activities.

Session 1, activity 1: "Classification game." In this activity, families selected a set of 12 coral images. Each image in the set was shown in succession, upon which the family would drag the image into one of two groups, as shown in Fig. **??**. They could drag an already-shown image to the other group at any time. The family also gave each group a name, which could be changed at any time.

Session 1, activity 2: "Anchor game." As shown in Figure 9.2, for this activity, families moved an image selection rectangle (anchor) to select the most important part of each image in a set of 12 coral images. They could resize and drag the anchor.

Session 1, activity 3: "Reflection." For this activity, families were asked to reflect on how they could use the Classification and Anchor games to make something useful for society. They were also asked to think about how a computer would play the same games.

Session 2, activity 1: "Objects Recognition." For this activity, families were told to download an object recognition app onto their phones and then go around the house looking for objects that would "trick the AI." They were then prompted to think through why that particular object tricked the AI.

Session 2, activity 2: "Train AI." This activity was done with *Teachable Machine*, a website that allows users to train their image recognition algorithm. Families were asked to pick 3 objects, train the Teachable Machine with these objects, and then test the Teachable Machine by holding up the same objects. They were also prompted to see what would happen if two objects were held up at the same time.

Session 2, activity 3: "Prediction Game." For this activity, families completed a digital form together, predicting what the computer would predict if it was trained on two types of images, and tested with a third image outside of the two training sets. Families worked through five of these scenarios: (1) train with 10 images of red fruits and 10 images of black fruits, test with an image of red and black fruits; (2) train with 9 images of pens and 9 images of rulers, test with an image of a Sharpie Magnum marker; (3) train with 9 images of chickens and 9 images of ducks, test with an image of a goose; (4) train with 5 images of red items and 20 images of yellow items, test with an image of a basketball; (5) train with 10 images of humans and 10 images of robots, test with an image of a robot with a human-like face. For each scenario, families could select between 3 choices: the first training group, the second training group, or a fill-in-the-blank "other" category. After completing the form, families were shown how Teachable Machine responded in these scenarios. Researchers then prompted reflection questions on the differences between families' predictions and the results.

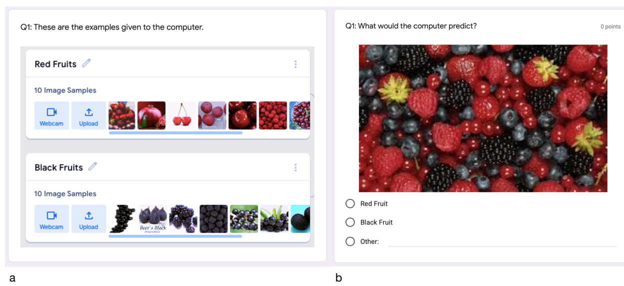

Figure 9.3: Example of the training (a) and test images (b) shown in the "Prediction Game" activity.

Session 3, activity 1: "Compare with Voice Assistant(VA)." As shown in Figure 9.4, this activity required one family member to ask the other a set of questions, including a custom question they came up with, and then ask the same questions to their family's voice assistant. Family members then compared the two answers and chose the answer they thought was better for each question.

Session 3, activity 2: "Draw what is inside." For this activity, families were asked to draw what they thought the inside of their voice assistant looked like. The prompt was spoken aloud to the families, who then completed the activity on a blank sheet of paper.

Session 4, activity 1: "AI Bingo Game." To complete this activity, families worked their way through the prompts on the worksheet shown in Figure 9.5, trying to get their voice assistant to say or do specific things. Examples include getting their voice assistant to "use the force," "play la Cucaracha," and telling them who made it.

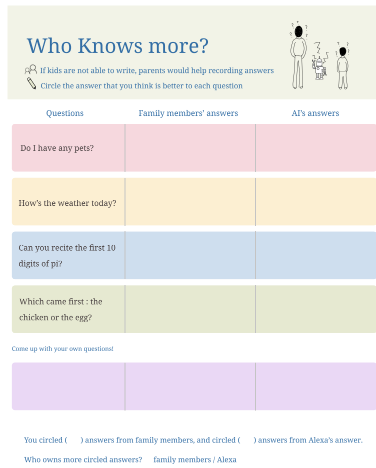

Figure 9.4: Worksheet for the "Compare with Voice Assistant" activity.

Session 4, activity 2: "Analyze AI Sheet." Families completed this worksheet, as shown in Figure 9.6 after playing the "AI Bingo Game." The sheet contained different potential characteristics of their voice assistant, depicted on continuums. Family members marked where the AI fell on the scale for each characteristic and then discussed why they made those choices.

Session 4, activity 3: "Design AI Sheet." For this activity, families were provided with the worksheet shown in Figure 9.7 and asked to design their own custom AI device. The worksheet included a toolkit at the bottom of common AI components (facial recognition, gesture sensor, speaker, eye recognition, sentiment analysis, camera, antenna, and touch sensor) and a drawing area with a starter shape.

Figure 9.5: Worksheet for the "AI Bingo Game" activity.

Final feedback questionnaire

The final feedback questionnaire was sent to families after the four sessions and consisted of 17 questions. For each session, the family answered the following questions on a scale from 1 to 5: (1) *Is it fun?* (2) *Is it easy to play?* (3) *How much have you learned?* They then submitted short answer responses to the following questions: (1) *What do you like most about this game?* (2) *What is missing in this game? (Something you wish to have?)*

9.3 Chapter 7

See fig.9.12 and fig.9.13

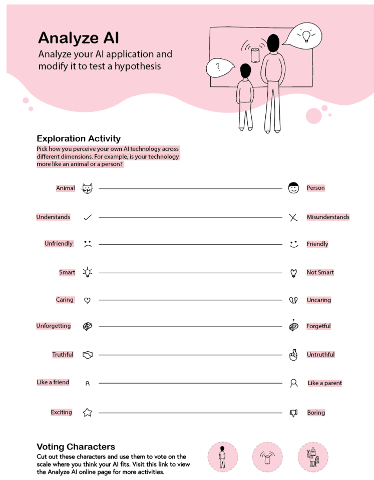

Figure 9.6: Worksheet for the "Analyze AI" activity.

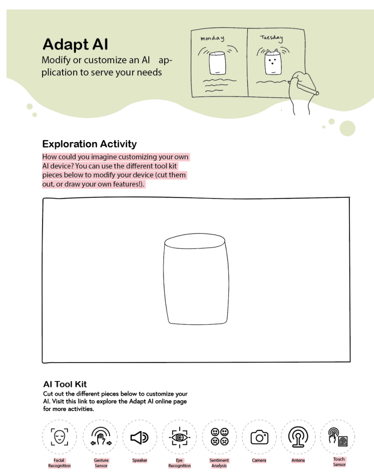

Figure 9.7: Worksheet for the "Design AI" activity.

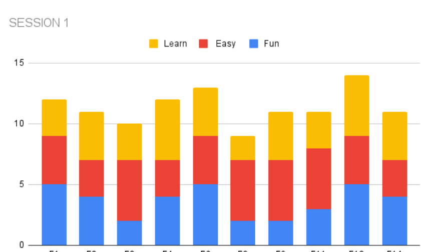

Figure 9.8: Chart presenting families feedback for session 1.

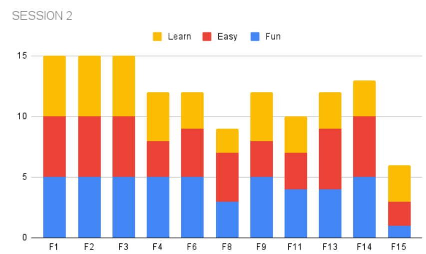

Figure 9.9: Chart presenting families feedback for session 2.

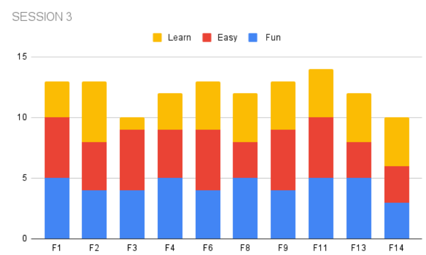

Figure 9.10: Chart presenting families feedback for session 3.

Figure 9.11: Chart presenting families feedback for session 4.

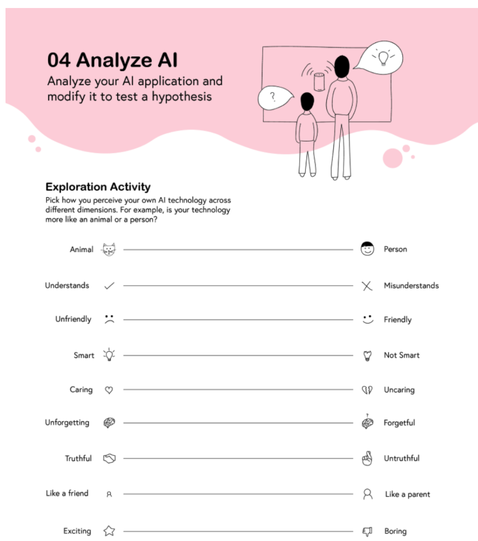

Figure 9.12: Study sheet where families can express their preferences for the AI friend persona and abilities.

Figure 9.13: Game tracing

BIBLIOGRAPHY

[1] Dynamicland. https://dynamicland.org/. Accessed: 2021-5-21.

[2] ebird. https://ebird.org/home. (Accessed on 06/02/2021).

[3] Echo kids privacy – echo dot kids edition privacy concerns and coppa violations site. https://www.echokidsprivacy.com/. (Accessed on 08/31/2021).

[4] Explore the gendering of ai voice assistants. https://en.unesco.org/EQUALS/voice-assistants. (Accessed on 04/12/2020).

[5] Froebel gifts kindergarten presechool curriculum educational toys. https://www.froebelgifts.com/gifts.htm. Accessed: 2021-5-22.

[6] Idea wiki. https://ideas.fandom.com/wiki/Ideas$_{W}iki. Accessed : 2021 - 5 - 21$.

[7] inaturalist. https://www.inaturalist.org/. (Accessed on 06/02/2021).

[8] Lego mindstorms. https://www.lego.com/en-us/mindstorms.

[9] Programming for kids. https://agentsheets.com/. Accessed: 2021-5-21.

[10] Scratch - imagine, program, share. https://scratch.mit.edu/. Accessed: 2021-5-21.

[11] Bird of paradise inspired ear cuff : 14 steps - instructables. https://www.instructables.com/Bird-of-Paradise-Inspired-Ear-Cuff/, 2022. (Accessed on 02/15/2023).

[12] Home — magicstory. https://magicstory.ai/, 2022. (Accessed on 02/15/2023).

[13] https://www.diffusioncraft.com. https://www.diffusioncraft.com/, 2022. (Accessed on 02/15/2023).

[14] Pål André Aarsand. Computer and video games in family life: The digital divide as a resource in intergenerational interactions. *Childhood*, 14(2):235–256, 2007.

[15] Ehsan Abbasnejad, Damien Teney, Amin Parvaneh, Javen Shi, and Anton van den Hengel. Counterfactual vision and language learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10044–10054, 2020.

[16] Edith Ackermann. The agency model of transactions: Toward an understanding of children's theory of control. *Psychologie genetique et sciences cognitives. Geneve: Fondation Archives Jean Piaget*, 1991.

[17] Amina Adadi and Mohammed Berrada. Peeking inside the black-box: A survey on explainable artificial intelligence (xai). *IEEE Access*, 6:52138–52160, 2018.

[18] Adept. Act-1: Transformer for actions. https://www.adept.ai/act, September 2022. (Accessed on 02/15/2023).

[19] Open AI. Gymnasium documentation. https://gymnasium.farama.org/, 2023. (Accessed on 02/15/2023).

[20] Efthimia Aivaloglou and Felienne Hermans. How kids code and how we know: An exploratory study on the scratch repository. In *Proceedings of the 2016 ACM Conference on International Computing Education Research*, pages 53–61, 2016.

[21] Otávio de P Albuquerque, Marcelo Fantinato, Marcelo M Eler, Sarajane M Peres, and Patrick CK Hung. A study of parental control requirements for smart toys. In *2020 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 2215–2220. IEEE, 2020.

[22] Safinah Ali, Nisha Devasia, Hae Won Park, and Cynthia Breazeal. Social robots as creativity eliciting agents. *Frontiers in Robotics and AI*, 8, 2021.

[23] Safinah Ali, Tyler Moroso, and Cynthia Breazeal. Can children learn creativity from a social robot? In *Proceedings of the 2019 on Creativity and Cognition*, Camp;C '19, page 359–368, New York, NY, USA, 2019. Association for Computing Machinery.

[24] Jose M Alonso. Explainable artificial intelligence for kids. In *2019 Conference of the International Fuzzy Systems Association and the European Society for Fuzzy Logic and Technology (EUSFLAT 2019)*. Atlantis Press, 2019.

[25] Patrícia Alves-Oliveira, Patrícia Arriaga, Ana Paiva, and Guy Hoffman. YOLO, a robot for creativity: A Co-Design study with children. In *Proceedings of the 2017 Conference on Interaction Design and Children*, IDC '17, pages 423–429, New York, NY, USA, June 2017. Association for Computing Machinery.

[26] Morgan G Ames. Charismatic technology. In *Proceedings of The Fifth Decennial Aarhus Conference on Critical Alternatives*, pages 109–120. Aarhus University Press, 2015.

[27] Tawfiq Ammari, Jofish Kaye, Janice Y Tsai, and Frank Bentley. Music, search, and iot: How people (really) use voice assistants. *ACM Trans. Comput. Hum. Interact.*, 26(3):17–1, 2019.

[28] L Anders. Ai for kids—it is our responsibility to enable children worldwide to engage with artificial intelligence. medium.[(accessed on 04/11/2020)], 2019.

[29] Monica Anderson. Mobile technology and home broadband 2019. *Pew Research Center*, 2, 2019.

[30] Julia Angwin, Jeff Larson, Surya Mattu, and Lauren Kirchner. Machine bias. *ProPublica, May*, 23:2016, 2016.

[31] Doris Ash. Reflective scientific sense-making dialogue in two languages: The science in the dialogue and the dialogue in the science. *Science Education*, 88(6):855–884, 2004.

[32] Dana Atwood-Blaine, Audrey C Rule, and Jordan Walker. Creative self-efficacy of children aged 9-14 in a science center using a situated mobile game. *Thinking Skills and Creativity*, 33:100580, 2019.

[33] Alexander Bacalja. "there's more going on": Critical digital game literacies and the imperative of praxis. In *Critical Digital Literacies: Boundary-Crossing Practices*, pages 185–204. Brill, 2021.

[34] A Bandura. Self-efficacy: toward a unifying theory of behavioral change. *Psychol. Rev.*, 84(2):191–215, March 1977.

[35] Albert Bandura. *The evolution of social cognitive theory*, pages 9–35. 2005.

[36] Rahul Banerjee, Leanne Liu, Kiley Sobel, Caroline Pitt, Kung Jin Lee, Meng Wang, Sijin Chen, Lydia Davison, Jason C Yip, Amy J Ko, et al. Empowering families facing english literacy challenges to jointly engage in computer programming. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, page 622. ACM, 2018.

[37] Gagan Bansal, Besmira Nushi, Ece Kamar, Daniel S Weld, Walter S Lasecki, and Eric Horvitz. Updates in human-ai teams: Understanding and addressing the performance/compatibility tradeoff. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 2429–2437, 2019.

[38] Tiffany Barnes, Heather Richter, Eve Powell, Amanda Chaffin, and Alex Godwin. Game2learn: building cs1 learning games for retention. In *Proceedings of the 12th annual SIGCSE conference on Innovation and technology in computer science education*, pages 121–125, 2007.

[39] Solon Barocas and Andrew D Selbst. Big data's disparate impact. *Calif. L. Rev.*, 104:671, 2016.

[40] Brigid Barron. Learning ecologies for technological fluency: Gender and experience differences. *Journal of Educational Computing Research*, 31(1):1–36, July 2004.

[41] Brigid Barron, Caitlin Kennedy Martin, Lori Takeuchi, and Rachel Fithian. Parents as Learning Partners in the Development of Technological Fluency. *International Journal of Learning and Media*, 1(2):55–77, May 2009.

[42] Christoph Bartneck, Dana Kulić, Elizabeth Croft, and Susana Zoghbi. Measurement instruments for the anthropomorphism, animacy, likeability, perceived intelligence, and perceived safety of robots. *International journal of social robotics*, 1(1):71–81, 2009.

[43] Mark Bauerlein. *The Digital Divide: Arguments for and Against Facebook, Google, Texting, and the Age of Social Netwo rking*. Penguin, 2011.

[44] Ronald A Beghetto. Creative Self-Efficacy: Correlates in middle and secondary students. *Creat. Res. J.*, 18(4):447–457, October 2006.

[45] Erin Beneteau, Ashley Boone, Yuxing Wu, Julie A Kientz, Jason Yip, and Alexis Hiniker. Parenting with alexa: Exploring the introduction of smart speakers on family dynamics.

[46] Erin Beneteau, Yini Guan, Olivia K. Richards, Mingrui Ray Zhang, Julie A. Kientz, Jason Yip, and Alexis Hiniker. Assumptions checked: How families learn about and use the echo dot.

[47] Erin Beneteau, Olivia K. Richards, Mingrui Zhang, Julie A. Kientz, Jason Yip, and Alexis Hiniker. Communication Breakdowns Between Families and Alexa. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, pages 1–13, Glasgow Scotland Uk, May 2019. ACM.

[48] Erin Beneteau, Olivia K Richards, Mingrui Zhang, Julie A Kientz, Jason Yip, and Alexis Hiniker. Communication breakdowns between families and alexa. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, page 243. ACM, 2019.

[49] Donna Berthelsen, Sue Walker, et al. Parents' involvement in their children's education. *Family matters*, (79):34, 2008.

[50] Karl-Emil Kjær Bilstrup, Magnus Høholt Kaspersen, Mille Skovhus Lunding, Marie-Monique Schaper, Maarten Van Mechelen, Mariana Aki Tamashiro, Rachel Charlotte Smith, Ole Sejer Iversen, and Marianne Graves Petersen. Supporting critical data literacy in k-9 education: three principles for enriching pupils' relationship to data. In *Interaction Design and Children*, pages 225–236, 2022.

[51] Paulo Blikstein. Bifocal modeling: a study on the learning outcomes of comparing physical and computational models linked in real time. In *Proceedings of the 14th ACM international conference on Multimodal interaction*, pages 257–264, 2012.

[52] Ali Borji. Generated faces in the wild: Quantitative comparison of stable diffusion, midjourney and dall-e 2. *arXiv preprint arXiv:2210.00586*, 2022.

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

[54] K Brennan. Beyond right or wrong: Challenges of including creative design activities in the classroom. *Journal of Technology and Teacher Education*, 2015.

[55] Rita Brito, Patrícia Dias, and Gabriela Oliveira. Young children, digital media and smart toys: How perceptions shape adoption and domestication. *British Journal of Educational Technology*, 49(5):807–820, 2018.

[56] Rita Brito, Rita Francisco, Patrícia Dias, and Stephane Chaudron. Family dynamics in digital homes: the role played by parental mediation in young children's digital practices around 14 european countries. *Contemporary Family Therapy*, 39(4):271–280, 2017.

[57] Urie Bronfenbrenner. *Ecological systems theory.* Jessica Kingsley Publishers, 1992.

[58] Urie Bronfenbrenner. Ecological models of human development. *Readings on the development of children*, 2(1):37–43, 1994.

[59] Amy Bruckman. Community support for constructionist learning. *Comput. Support. Coop. Work*, 7(1):47–86, March 1998.

[60] Joy Buolamwini and Timnit Gebru. Gender shades: Intersectional accuracy disparities in commercial gender classification. In *Conference on fairness, accountability and transparency*, pages 77–91, 2018.

[61] Rosemary Anne Callingham. Assessing statistical literacy: A question of interpretation? In *International Conference on Teaching Statistics (ICOTS)*, pages 1–6, 2006.

[62] Marjolein CJ Caniëls and Eric F Rietzschel. Organizing creativity: Creativity and innovation under constraints. *Creativity and Innovation Management*, 24(2):184–196, 2015.

[63] Courtney Cazden, Bill Cope, Norman Fairclough, Jim Gee, Mary Kalantzis, Gunther Kress, Allan Luke, Carmen Luke, Sarah Michaels, and Martin Nakata. A pedagogy of multiliteracies: Designing social futures. *Harvard educational review*, 66(1):60–92, 1996.

[64] Angela Chang and Cynthia Breazeal. Tinkrbook: shared reading interfaces for storytelling. In *Proceedings of the 10th International Conference on Interaction Design and Children*, pages 145–148. ACM, 2011.

[65] Vicky Charisi, Laura Malinverni, Marie-Monique Schaper, and Elisa Rubegni. Creating opportunities for children's critical reflections on ai, robotics and other intelligent technologies. In *Proceedings of the 2020 ACM interaction design and children conference: extended abstracts*, pages 89–95, 2020.

[66] Kathy Charmaz. *Constructing grounded theory: A practical guide through qualitative analysis*. sage, 2006.

[67] Elizabeth 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.

[68] Yu Chi, Wei Jeng, Amelia Acker, and Leanne Bowler. Affective, behavioral, and cognitive aspects of teen perspectives on personal data in social media: A model of youth data literacy. In *International Conference on Information*, pages 442–452. Springer, 2018.

[69] Tara Chklovski, Richard Jung, Janet Bih Fofang, Pamela Gonzales, Bolivia Tech Hub, and Bolivia La Paz. Implementing a 15-week ai-education program with under-resourced families across 13 global communities. In *International joint conference on artificial intelligence*, 2019.

[70] Drew Cingel and Anne Marie Piper. How parents engage children in tablet-based reading experiences: An exploration of haptic feedback. In *Proceedings of the 2017 ACM Conference on Computer Supported Cooperative Work and Social Computing*, pages 505–510, 2017.

[71] Merijke Coenraad. “that’s what techquity is”: youth perceptions of technological and algorithmic bias. *Information and Learning Sciences*, (ahead-of-print), 2022.

[72] Teresa Correa. Bottom-up technology transmission within families: Exploring how youths influence their parents’ digital media use with dyadic data. *Journal of communication*, 64(1):103–124, 2014.

[73] Suzy Cox. *A conceptual analysis of technological pedagogical content knowledge*. Brigham Young University, 2008.

[74] Sarah M Coyne, Jenny Radesky, Kevin M Collier, Douglas A Gentile, Jennifer Ruh Linder, Amy I Nathanson, Eric E Rasmussen, Stephanie M Reich, and Jean Rogers. Parenting and digital media. *Pediatrics*, 140(Supplement 2):S112–S116, 2017.

[75] Kathryn Cunningham. Purpose-first programming: A programming learning approach for learners who care most about what code achieves. In *Proceedings of the 2020 ACM Conference on International Computing Education Research*, pages 348–349, 2020.

[76] N Davis, C Hsiao, K Y Singh, B Lin, and B Magerko. Quantifying collaboration with a Co-Creative drawing agent. *ACM Trans. Interact. Intell. Syst.*, 7(4):1–25, December 2017.

[77] Roberto Santiago de Roock. On the material consequences of (digital) literacy: Digital writing with, for, and against racial capitalism. *Theory Into Practice*, 60(2):183–193, 2021.

[78] Erica Deahl. Better the data you know: Developing youth data literacy in schools and informal learning environments. *Available at SSRN 2445621*, 2014.

[79] Dominik Dellermann, Adrian Calma, Nikolaus Lipusch, Thorsten Weber, Sascha Weigel, and Philipp Ebel. The future of human-ai collaboration: a taxonomy of design knowledge for hybrid intelligence systems. In *Proceedings of the 52nd Hawaii International Conference on System Sciences*, 2019.

[80] Peter J Denning and Matti Tedre. *Computational thinking*. MIT Press, 2019.

[81] Brenda Dervin. Sense-making theory and practice: an overview of user interests in knowledge seeking and use. *Journal of knowledge management*, 2(2):36–46, 1998.

[82] Manuj Dhariwal and Shruti Dhariwal. Let’s chance: Playful probabilistic programming for children. In *Extended Abstracts of the 2020 CHI Conference on Human Factors in Computing Systems*, CHI EA ’20, pages 1–7, New York, NY, USA, April 2020. Association for Computing Machinery.

[83] Patrícia Dias, Rita Brito, Wannes Ribbens, Linda Daniela, Zanda Rubene, Michael Dreier, Monica Gemo, Rosanna Di Gioia, and Stéphane Chaudron. The role of parents in the engagement of young children with digital technologies: Exploring tensions between rights of access and protection, from 'gatekeepers' to 'scaffolders'. *Global Studies of Childhood*, 6(4):414–427, 2016.

[84] Betsy DiSalvo, Parisa Khanipour Roshan, and Briana Morrison. Information seeking practices of parents: Exploring skills, face threats and social networks. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, pages 623–634, 2016.

[85] Betsy DiSalvo, Cecili Reid, and Parisa Khanipour Roshan. They can't find us: the search for informal cs education. In *Proceedings of the 45th ACM technical symposium on Computer science education*, pages 487–492, 2014.

[86] Andrea A DiSessa. *Changing minds: Computers, learning, and literacy.* Mit Press, 2001.

[87] S Dow, B MacIntyre, J Lee, C Oezbek, and others. Wizard of oz support throughout an iterative design process. *IEEE Pervasive Comput.*, 2005.

[88] Steven P Dow, Alana Glassco, Jonathan Kass, Melissa Schwarz, Daniel L Schwartz, and Scott R Klemmer. Parallel prototyping leads to better design results, more divergence, and increased self-efficacy. *ACM Trans. Comput.-Hum. Interact.*, 17(4):1–24, December 2011.

[89] Stefania Druga. *Growing up with AI: Cognimates: from coding to teaching machines.* PhD thesis, Massachusetts Institute of Technology, 2018.

[90] Stefania Druga, Thomas Ball, and Amy J Ko. How families design and program games: a qualitative analysis of a 4-week online in-home study. *IDC '22: ACM Conference on Interaction Design and Children*, 2022.

[91] Stefania Druga, Fee Christoph, and Amy J. Ko. Family as a third space for ai literacies: How do children and parents learn about ai together? *CHI '22: ACM Conference on Computer-Human Interaction*, 2022.

[92] Stefania Druga, Fee Lia Christoph, and Amy J. Ko. Family as a third space for ai literacies: How do children and parents learn about ai together? In *CHI Conference on Human Factors in Computing Systems*, pages 1–17, 2022.

[93] Stefania Druga and Amy J. Ko. How do children's perceptions of machine intelligence change when training and coding smart programs? *Interaction Design for Children ACM*, 2021.

[94] Stefania Druga and Amy J Ko. How do children's perceptions of machine intelligence change when training and coding smart programs? In *Interaction Design and Children*, pages 49–61, Athens Greece, June 2021. ACM.

[95] Stefania Druga and Amy J Ko. How do children's perceptions of machine intelligence change when training and coding smart programs? In *Interaction design and children*, pages 49–61, 2021.

[96] Stefania Druga, Nancy Otero, and Amy J Ko. The landscape of teaching resources for ai education. *ITICSE '22: ACM conference on Innovation and Technology in Computer Science Education*, 2022.

[97] Stefania Druga, S Vu, E Likhith, L Oh, T Qui, and C Breazeal. Cognimates, 2018.

[98] Stefania Druga, Sarah T Vu, Eesh Likhith, and Tammy Qiu. Inclusive ai literacy for kids around the world. In *Proceedings of FabLearn 2019*, pages 104–111. 2019.

[99] Stefania Druga, Sarah T Vu, Eesh Likhith, and Tammy Qiu. Inclusive ai literacy for kids around the world. In *Proceedings of FabLearn 2019*, pages 104–111. ACM, 2019.

[100] Stefania Druga, Randi Williams, Cynthia Breazeal, and Mitchel Resnick. Hey google is it ok if i eat you?: Initial explorations in child-agent interaction. In *Proceedings of the 2017 Conference on Interaction Design and Children*, pages 595–600. ACM, 2017.

[101] Stefania Druga, Randi Williams, Hae Won Park, and Cynthia Breazeal. How smart are the smart toys?: Children and parents' agent interaction and intelligence attribution. In *Proceedings of the 17th ACM Conference on Interaction Design and Children*, IDC '18, pages 231–240, New York, NY, USA, 2018. ACM.

[102] Stefania Druga, Randi Williams, Hae Won Park, and Cynthia Breazeal. How smart are the smart toys? children and parents' agent interaction and intelligence attribution. In *Proceedings of the 17th ACM Conference on Interaction Design and Children*, pages 231–240, 2018.

[103] Stefania Druga, Jason Yip, Michael Preston, and Devin Dillon. The 4as: Ask, adapt, author, analyze-ai literacy framework for families. In *Algorithmic Rights and Protections for Children*. PubPub, 2021.

[104] A Druin. The role of children in the design of new technology. submitted to. *ACM Transactions on Human-Computer Interaction*, 2000.

[105] Mike Van Duuren. Gauging children's understanding of artificially intelligent objects: A presentation of"counterfactuals. *International Journal of Behavioral Development*, 22(4):871–889, 1998.

[106] Catherine D'Ignazio and Rahul Bhargava. Approaches to building big data literacy. In *Proceedings of the Bloomberg data for good exchange conference*, 2015.

[107] Steven Earnshaw. *The handbook of creative writing*. Edinburgh University Press, 2007.

[108] Common Sense Education. Common sense. https://www.commonsense.org/, February 2023. (Accessed on 02/10/2023).

[109] Yrjö Engeström. *Learning by expanding*. Cambridge University Press, 2015.

[110] Thomas Erickson. Designing agents as if people mattered. *Software agents*, pages 79–96, 1997.

[111] Carrie A. Ewin, Andrea E. Reupert, Louise A. McLean, and Christopher J. Ewin. The impact of joint media engagement on parent–child interactions: A systematic review. *Human Behavior and Emerging Technologies*, 3(2):230–254, April 2021.

[112] Mary Fawcett and Penny Hay. 5x5x5 = creativity in the early years. *The international journal of art & design education*, 23(3):234–245, October 2004.

[113] Andrew Guthrie Ferguson. Predictive policing and reasonable suspicion. *Emory LJ*, 62:259, 2012.

[114] Sue Fitzgerald, Beth Simon, and Lynda Thomas. Strategies that students use to trace code: an analysis based in grounded theory. In *Proceedings of the first international workshop on Computing education research*, pages 69–80, 2005.

[115] Brian J Fogg. Persuasive technology: using computers to change what we think and do. *Ubiquity*, 2002(December):5, 2002.

[116] Sean Follmer, Hayes Raffle, Janet Go, Rafael Ballagas, and Hiroshi Ishii. Video play: playful interactions in videoconferencing for long-distance families with young children. In *Proceedings of the 9th International Conference onInteraction Design and Children*, pages 49–58. ACM, 2010.

[117] Terrence Fong, Illah Nourbakhsh, and Kerstin Dautenhahn. A survey of socially interactive robots. *Robotics and autonomous systems*, 42(3):143–166, 2003.

[118] Diana Franklin, David Weintrop, Jennifer Palmer, Merijke Coenraad, Melissa Cobian, Kristan Beck, Andrew Rasmussen, Sue Krause, Max White, Marco Anaya, and Zachary Crenshaw. Scratch encore: The design and pilot of a Culturally-Relevant intermediate scratch curriculum. In *Proceedings of the 51st ACM Technical Symposium on Computer Science Education*, SIGCSE '20, pages 794–800, New York, NY, USA, February 2020. Association for Computing Machinery.

[119] Diana Franklin, David Weintrop, Jennifer Palmer, Merijke Coenraad, Melissa Cobian, Kristan Beck, Andrew Rasmussen, Sue Krause, Max White, Marco Anaya, and Zachary Crenshaw. Scratch encore: The design and pilot of a Culturally-Relevant intermediate scratch curriculum. In *Proceedings of the 51st ACM Technical Symposium on Computer Science Education*, SIGCSE '20, pages 794–800, New York, NY, USA, February 2020. Association for Computing Machinery.

[120] Natalie Anne Freed. *" This is the fluffy robot that only speaks french": language use between preschoolers, their families, and a social robot while sharing virtual toys*. PhD thesis, Massachusetts Institute of Technology, 2012.

[121] Paolo Freire. Pedagogy of the oppressed (revised). *New York: Continuum*, 1996.

[122] Paulo Freire. Pedagogy of the oppressed, 1968.

[123] Paulo Freire. *Pedagogy of the oppressed*. Routledge, 2013.

[124] Jonas Frich, Lindsay MacDonald Vermeulen, Christian Remy, Michael Mose Biskjaer, and Peter Dalsgaard. Mapping the landscape of creativity support tools in HCI. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, number Paper 389 in CHI '19, pages 1–18, New York, NY, USA, May 2019. Association for Computing Machinery.

[125] Batya Friedman, Peter H Kahn, and Alan Borning. Value sensitive design and information systems. *The handbook of information and computer ethics*, pages 69–101, 2008.

[126] Howard Gardner and Katie Davis. 6. acts (and apps) of imagination among today's youth. In *The App Generation*, pages 120–154. Yale University Press, 2013.

[127] Timnit Gebru. Oxford handbook on ai ethics book chapter on race and gender. *arXiv preprint arXiv:1908.06165*, 2019.

[128] Timnit Gebru. Oxford handbook on AI ethics book chapter on race and gender. August 2019.

[129] Clifford Geertz. Thick description: Toward an interpretive theory of culture. In *The cultural geography reader*, pages 41–51. Routledge, 2008.

[130] Josh A Goldstein, Girish Sastry, Micah Musser, Renee DiResta, Matthew Gentzel, and Katerina Sedova. Generative language models and automated influence operations: Emerging threats and potential mitigations. *arXiv preprint arXiv:2301.04246*, 2023.

[131] Alison Gopnik. Childhood as a solution to explore–exploit tensions. *Philosophical Transactions of the Royal Society B*, 375(1803):20190502, 2020.

[132] Claire Le Goues, Michael Pradel, and Abhik Roychoudhury. Automated program repair. *Communications of the ACM*, 62(12):56–65, 2019.

[133] Madhu Govind, Emily Relkin, and Marina Umaschi Bers. Engaging children and parents to code together using the scratchjr app. *Visitor Studies*, 23(1):46–65, 2020.

[134] Jacek Gralewski and Dorota Maria Jankowska. Do parenting styles matter? perceived dimensions of parenting styles, creative abilities and creative self-beliefs in adolescents. *Thinking skills and creativity*, 38:100709, 2020.

[135] Heather M Gray, Kurt Gray, and Daniel M Wegner. Dimensions of mind perception. *Science*, 315(5812):619–619, 2007.

[136] Joshua Grossman, Zhiyuan Lin, Hao Sheng, Johnny T-Z Wei, Joseph J Williams, and Sharad Goel. Mathbot: Transforming online resources for learning math into conversational interactions. 2019.

[137] Mona Leigh Guha, Allison Druin, Gene Chipman, Jerry Alan Fails, Sante Simms, and Allison Farber. Mixing ideas: a new technique for working with young children as design partners. In *Proceedings of the 2004 conference on Interaction design and children: building a community*, pages 35–42, 2004.

[138] Philip Guo. Ten million users and ten years later: Python tutor's design guidelines for building scalable and sustainable research software in academia. In *The 34th Annual ACM Symposium on User Interface Software and Technology*, pages 1235–1251, 2021.

[139] Philip J Guo. Online python tutor: embeddable web-based program visualization for cs education. In *Proceeding of the 44th ACM technical symposium on Computer science education*, pages 579–584, 2013.

[140] Philip J Guo. Codeopticon: Real-Time, One-To-Many human tutoring for computer programming. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology*, UIST '15, pages 599–608, New York, NY, USA, November 2015. Association for Computing Machinery.

[141] Kris Gutierrez, Betsy Rymes, and Joanne Larson. Script, counterscript, and underlife in the classroom: James brown versus brown v. board of education. *Harvard educational review*, 65(3):445–472, 1995.

[142] Kris D Gutiérrez. Developing a sociocritical literacy in the third space. *Reading research quarterly*, 43(2):148–164, 2008.

[143] Kris D Gutiérrez, P Zitlali Morales, and Danny C Martinez. Re-mediating literacy: Culture, difference, and learning for students from nondominant communities. *Review of research in education*, 33(1):212–245, 2009.

[144] Kris D Gutiérrez and Barbara Rogoff. Cultural ways of learning: Individual traits or repertoires of practice. *Educational researcher*, 32(5):19–25, 2003.

[145] James K Hammerman and Andee Rubin. Strategies for managing statistical complexity with new software tools. *Statistics Education research journal*, 3(2):17–41, 2004.

[146] Clint Andrew Heinze, Janet Haase, and Helen Higgins. An action research report from a multi-year approach to teaching artificial intelligence at the k-6 level. In *First AAAI Symposium on Educational Advances in Artificial Intelligence*, 2010.

[147] Felienne Hermans and Efthimia Aivaloglou. Do code smells hamper novice programming? a controlled experiment on scratch programs. In *2016 IEEE 24th International Conference on Program Comprehension (ICPC)*, pages 1–10. IEEE, 2016.

[148] Tom Hitron, Yoav Orlev, Iddo Wald, Ariel Shamir, Hadas Erel, and Oren 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.

[149] Michael S Hsiao. Automated program synthesis from object-oriented natural language for computer games. In *CNL*, pages 71–74, 2018.

[150] Seth E Hunter, Pattie Maes, Anthony Tang, Kori M Inkpen, and Susan M Hessey. Waazam! supporting creative play at a distance in customized video environments. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1197–1206, 2014.

[151] ICO. Age appropriate design: a code of practice for online services. https://ico.org.uk/, 2021. (Accessed on 08/28/2021).

[152] ICO. Age appropriate design: a code of practice for online services — ico. https://ico.org.uk/for-organisations/guide-to-data-protection/ico-codes-of-practice/age-appropriate-design-a-code-of-practice-for-online-services/, 2021. (Accessed on 08/28/2021).

[153] K Inagaki. Young children's differentiation of plants from nonliving things in terms of growth. In *biennial meeting of Society for Research in Child Development, New Orleans*, 1993.

[154] Dhanya Jayagopal, Justin Lubin, and Sarah E Chasins. Exploring the learnability of program synthesizers by novice programmers. In *Proceedings of the 35th Annual ACM Symposium on User Interface Software and Technology*, pages 1–15, 2022.

[155] Matthew Johnson, Katja Hofmann, Tim Hutton, and David Bignell. The malmo platform for artificial intelligence experimentation. In *Ijcai*, pages 4246–4247, 2016.

[156] Tejinder K Judge and Carman Neustaedter. Sharing conversation and sharing life: video conferencing in the home. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 655–658, 2010.

[157] Malte F Jung, Nik Martelaro, Halsey Hoster, and Clifford Nass. Participatory materials: having a reflective conversation with an artifact in the making. In *Proceedings of the 2014 conference on Designing interactive systems*, DIS '14, pages 25–34, New York, NY, USA, June 2014. Association for Computing Machinery.

[158] Yasmin Kafai, Chris Proctor, and Debora Lui. From theory bias to theory dialogue: embracing cognitive, situated, and critical framings of computational thinking in k-12 cs education. *ACM Inroads*, 11(1):44–53, 2020.

[159] Yasmin B Kafai and Quinn Burke. *Connected code: Why children need to learn programming*. Mit Press, 2014.

[160] K Megasari Kahn, Rani Megasari, Erna Piantari, and Enjun Junaeti. Ai programming by children using snap! block programming in a developing country. 2018.

[161] Peter H Kahn, Takayuki Kanda, Hiroshi Ishiguro, Brian T Gill, Solace Shen, Jolina H Ruckert, and Heather E Gary. Human creativity can be facilitated through interacting with a social robot. In *2016 11th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pages 173–180. ieeexplore.ieee.org, March 2016.

[162] Takayuki Kanda, Takayuki Hirano, Daniel Eaton, and Hiroshi Ishiguro. Interactive robots as social partners and peer tutors for children: A field trial. *Human-computer interaction*, 19(1):61–84, 2004.

[163] Hyeonsu Kang and Philip J Guo. Omnicode: A novice-oriented live programming environment with always-on run-time value visualizations. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology*, pages 737–745, 2017.

[164] Pegah Karimi, Kazjon Grace, Mary Lou Maher, and Nicholas Davis. Evaluating creativity in computational Co-Creative systems. July 2018.

[165] Pegah Karimi, Jeba Rezwana, Safat Siddiqui, Mary Lou Maher, and Nasrin Dehbozorgi. Creative sketching partner: an analysis of human-AI co-creativity. In *Proceedings of the 25th International Conference on Intelligent User Interfaces*, IUI '20, pages 221–230, New York, NY, USA, March 2020. Association for Computing Machinery.

[166] Maciej Karwowski. Development of the creative self-concept. *Creativity. Theories–Research-Applications*, 2(2):165–179, 2015.

[167] Caitlin Kelleher and Randy Pausch. Lowering the barriers to programming: A taxonomy of programming environments and languages for novice programmers. *ACM Computing Surveys (CSUR)*, 37(2):83–137, June 2005.

[168] James Kennedy, Séverin Lemaignan, Caroline Montassier, Pauline Lavalade, Bahar Irfan, Fotios Papadopoulos, Emmanuel Senft, and Tony Belpaeme. Child speech recognition in human-robot interaction: evaluations and recommendations. In *Proceedings of the 2017 ACM/IEEE international conference on human-robot interaction*, pages 82–90, 2017.

[169] Ada S Kim and Amy J Ko. A pedagogical analysis of online coding tutorials. In *Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education*, pages 321–326, 2017.

[170] Joy Kim, Maneesh Agrawala, and Michael S Bernstein. Mosaic: Designing online creative communities for sharing Works-in-Progress. In *Proceedings of the 2017 ACM Conference on Computer Supported Cooperative Work and Social Computing*, CSCW '17, pages 246–258, New York, NY, USA, February 2017. Association for Computing Machinery.

[171] Seonghun Kim, Yeonju Jang, Woojin Kim, Seongyune Choi, Heeseok Jung, Soohwan Kim, and Hyeoncheol Kim. Why and what to teach: Ai curriculum for elementary

school. In *proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 15569–15576, 2021.

[172] Jerome Kirk, Marc L Miller, and Marc L Miller. *Reliability and validity in qualitative research*, volume 1. Sage, 1986.

[173] Jill Klefstad. Focus on family: Environments that foster inquiry and critical thinking in young children: Supporting children's natural curiosity. *Childhood Education*, 91(2):147–149, 2015.

[174] Amy J. Ko, Robin Abraham, Laura Beckwith, Alan Blackwell, Margaret Burnett, Martin Erwig, Chris Scaffidi, Joseph Lawrance, Henry Lieberman, Brad Myers, et al. The state of the art in end-user software engineering. *ACM Computing Surveys (CSUR)*, 43(3):1–44, 2011.

[175] Amy J Ko, Alannah Oleson, Neil Ryan, Yim Register, Benjamin Xie, Mina Tari, Matthew Davidson, Stefania Druga, and Dastyni Loksa. It is time for more critical CS education. *Commun. ACM*, 63(11):31–33, October 2020.

[176] Amy J Ko, Alannah Oleson, Neil Ryan, Yim Register, Benjamin Xie, Mina Tari, Matthew Davidson, Stefania Druga, and Dastyni Loksa. It is time for more critical cs education. *Communications of the ACM*, 63(11):31–33, 2020.

[177] Matthew Koehler and Punya Mishra. What is technological pedagogical content knowledge (tpack)? *Contemporary issues in technology and teacher education*, 9(1):60–70, 2009.

[178] Clifford Konold, Traci Higgins, Susan Jo Russell, and Khalimahtul Khalil. Data seen through different lenses. *Educational Studies in Mathematics*, 88(3):305–325, 2015.

[179] Jan-Peter Kramer, Joachim Kurz, Thorsten Karrer, and Jan Borchers. How live coding affects developers' coding behavior. In *2014 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC)*, pages 5–8. IEEE, 2014.

[180] Todd Kulesza, Simone Stumpf, Margaret Burnett, and Irwin Kwan. Tell me more? the effects of mental model soundness on personalizing an intelligent agent. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1–10, 2012.

[181] Chronis Kynigos et al. Half-baked logo microworlds as boundary objects in integrated design. *Informatics in Education-An International Journal*, 6(2):335–359, 2007.

[182] Micheal Lanham. *Learn Unity ML-Agents–Fundamentals of Unity Machine Learning: Incorporate new powerful ML algorithms such as Deep Reinforcement Learning for games*. Packt Publishing Ltd, 2018.

[183] Lotta C Larson and Teresa Northern Miller. 21st century skills: Prepare students for the future. *Kappa Delta Pi Record*, 47(3):121–123, 2011.

[184] Jong-Eun Lee, Jeanne D Day, Naomi M Meara, and Scott Maxwell. Discrimination of social knowledge and its flexible application from creativity: a multitrait–multimethod approach. *Pers. Individ. Dif.*, 32(5):913–928, April 2002.

[185] Michael J Lee, Faezeh Bahmani, Irwin Kwan, Jilian LaFerte, Polina Charters, Amber Horvath, Fanny Luor, Jill Cao, Catherine Law, Michael Beswetherick, et al. Principles of a debugging-first puzzle game for computing education. In *2014 IEEE symposium on visual languages and human-centric computing (VL/HCC)*, pages 57–64. IEEE, July 2014.

[186] Michael J Lee, Faezeh Bahmani, Irwin Kwan, Jilian LaFerte, Polina Charters, Amber Horvath, Fanny Luor, Jill Cao, Catherine Law, Michael Beswetherick, Sheridan Long, Margaret Burnett, and Amy J Ko. Principles of a debugging-first puzzle game for computing education. In *2014 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC)*, pages 57–64, July 2014.

[187] Michael J Lee and Amy J Ko. Personifying programming tool feedback improves novice programmers' learning. In *Proceedings of the seventh international workshop on Computing education research*, ICER '11, pages 109–116, New York, NY, USA, August 2011. Association for Computing Machinery.

[188] Yong Jae Lee, C Lawrence Zitnick, and Michael F Cohen. Shadowdraw: real-time user guidance for freehand drawing. *ACM Transactions on Graphics (TOG)*, 30(4):1–10, 2011.

[189] Yong Jae Lee, C Lawrence Zitnick, and Michael F Cohen. ShadowDraw: real-time user guidance for freehand drawing. *ACM Trans. Graph.*, 30(4):1–10, July 2011.

[190] Shane Legg and Marcus Hutter. Universal intelligence: A definition of machine intelligence. *Minds and machines*, 17(4):391–444, 2007.

[191] Golan Levin and Tega Brain. *Code as creative medium: a handbook for computational art and design*. MIT Press, 2021.

[192] Sharona T Levy and David Mioduser. Does it ”want” or ”was it programmed to...”? kindergarten children’s explanations of an autonomous robot’s adaptive functioning. *International Journal of Technology and Design Education*, 18(4):337–359, 2008.

[193] Chen-Chu Liang and Yu-Hsi Yuan. Exploring children’s creative self-efficacy affected by after-school program and parent–child relationships. *Frontiers in Psychology*, 11:2237, 2020.

[194] Phoebe Lin, Jessica Van Brummelen, Galit Lukin, Randi Williams, and Cynthia 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.

[195] Thomas R Lindlof and Bryan C Taylor. *Qualitative communication research methods*. Sage publications, 2017.

[196] Annabel Lindner, Stefan Seegerer, and Ralf Romeike. Unplugged activities in the context of ai. In *International Conference on Informatics in Schools: Situation, Evolution, and Perspectives*, pages 123–135. Springer, 2019.

[197] Duri Long, Takeria Blunt, and Brian Magerko. Co-designing ai literacy exhibits for informal learning spaces. *CSCW ’21: ACM Conference on Computer Supported Cooperative Work and Social Computing*, 10 2021.

[198] Duri Long and Brian 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.

[199] Duri Long and Brian 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.

[200] Duri Long, Anthony Teachey, and Brian Magerko. Family learning talk in ai literacy learning activities. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*, pages 1–20, 2022.

[201] Shi-Jer Lou, Yung-Chieh Chou, Ru-Chu Shih, and Chih-Chao Chung. A study of creativity in cac2 steamship-derived stem project-based learning. *Eurasia Journal of Mathematics, Science and Technology Education*, 13(6):2387–2404, 2017.

[202] Pedro Lucas and Carlos Martinho. Stay awhile and listen to 3buddy, a co-creative level design support tool. In *ICCC*, pages 205–212. computationalcreativity.net, 2017.

[203] Dorian Lynskey. 'alexa, are you invading my privacy?' – the dark side of our voice assistants. https://www.theguardian.com/technology/2019/oct/09/alexa-are-you-invading-my-privacy-the-dark-side-of-our-voice-assistants/, 2019. [Online; accessed 19-January-2020].

[204] Mary Madden, Michele Gilman, Karen Levy, and Alice Marwick. Privacy, poverty, and big data: A matrix of vulnerabilities for poor americans. *Washington University Law Review*, 95(1):053–125, 2017.

[205] Katie Makar and Andee Rubin. A framework for thinking about informal statistical inference. *Statistics Education Research Journal*, 8(1), 2009.

[206] John Maloney, Mitchel Resnick, Natalie Rusk, Brian Silverman, and Evelyn Eastmond. The scratch programming language and environment. *ACM Transactions on Computing Education (TOCE)*, 10(4):1–15, 2010.

[207] Jane Margolis. *Stuck in the shallow end: Education, race, and computing*. MIT press, 2010.

[208] Radu Mariescu-Istodor and Ilkka Jormanainen. Machine learning for high school students. In *Proceedings of the 19th Koli Calling International Conference on Computing Education Research*, pages 1–9, 2019.

[209] Andrew McNutt, Anton Outkine, and Ravi Chugh. A study of editor features in a creative coding classroom. *arXiv preprint arXiv:2301.13302*, 2023.

[210] Emily McReynolds, Sarah Hubbard, Timothy Lau, Aditya Saraf, Maya Cakmak, and Franziska Roesner. Toys that listen: A study of parents, children, and internet-connected toys. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, pages 5197–5207. ACM, 2017.

[211] Pekka Mertala. The pedagogy of multiliteracies as a code breaker: A suggestion for a transversal approach to computing education in basic education. *British Journal of Educational Technology*, 2021.

[212] Rebecca Michelson, Akeiylah Dewitt, Ria Nagar, Alexis Hiniker, Jason Yip, Sean Munson, and Julie Kientz. Parenting in a pandemic: Juggling multiple roles and managing technology use in family life during covid-19 in the united states. *Article*, 5, 10 2021.

[213] Rebecca Michelson, Akeiylah DeWitt, Ria Nagar, Alexis Hiniker, Jason Yip, Sean A Munson, and Julie A Kientz. Parenting in a pandemic: Juggling multiple roles and

managing technology use in family life during covid-19 in the united states. *Proceedings of the ACM on Human-Computer Interaction*, 5(CSCW2):1–39, 2021.

[214] Microsoft. Github next — hey, github! https://githubnext.com/projects/hey-github/, 2023 2023. (Accessed on 02/15/2023).

[215] GitHub Microsoft. Github copilot · your ai pair programmer. https://github.com/features/copilot, February 2023. (Accessed on 02/12/2023).

[216] Matthew B Miles and A Michael Huberman. Drawing valid meaning from qualitative data: Toward a shared craft. *Educational researcher*, 13(5):20–30, 1984.

[217] David Mioduser and Sharona T Levy. Making sense by building sense: Kindergarten children's construction and understanding of adaptive robot behaviors. *International Journal of Computers for Mathematical Learning*, 15(2):99–127, 2010.

[218] LLK MIT. Scratch statistics - imagine, program, share. https://scratch.mit.edu/statistics/, February 2023. (Accessed on 02/15/2023).

[219] Dana L Mitra. Youth as a bridge between home and school: Comparing student voice and parent involvement as strategies for change. *Education and Urban Society*, 38(4):455–480, 2006.

[220] Luis C Moll and James B Greenberg. Creating zones of possibilities: Combining social contexts for instruction. *Vygotsky and education: Instructional implications and applications of sociohistorical psychology*, pages 319–348, 1990.

[221] Luis Morales-Navarro, Naomi Thompson, Yasmin Kafai, Mia Shaw, and Nichole Pinkard. Reimagining and co-designing with youth an hour of code activity for critical engagement with computing. In *Interaction Design and Children*, pages 288–296, 2022.

[222] Mozilla. Common Voice Initiative. https://voice.mozilla.org/, 2019. [Online; accessed 2-December-2019].

[223] Geoff Musick, Guo Freeman, and Nathan J McNeese. Gaming as family time: Digital game co-play in modern parent-child relationships. *Proceedings of the ACM on Human-Computer Interaction*, 5(CHI PLAY):1–25, 2021.

[224] Clifford Nass and Youngme Moon. Machines and mindlessness: Social responses to computers. *Journal of social issues*, 56(1):81–103, 2000.

[225] Greg L Nelson, Benjamin Xie, and Amy J Ko. Comprehension first: evaluating a novel pedagogy and tutoring system for program tracing in cs1. In *Proceedings of the 2017 ACM Conference on International Computing Education Research*, pages 2–11, 2017.

[226] Michelle M. Neumann. Parent scaffolding of young children's use of touch screen tablets. *Early Child Development and Care*, 188(12):1654–1664, December 2018.

[227] Peter Nikken and Jeroen Jansz. Parental mediation of children's videogame playing: A comparison of the reports by parents and children. *Learning, media and technology*, 31(2):181–202, 2006.

[228] Philip Oltermann. German parents told to destroy doll that can spy on children. *The Guardian*, 2017.

[229] Cathy O'neil. *Weapons of math destruction: How big data increases inequality and threatens democracy*. Broadway Books, 2016.

[230] Anastasia K Ostrowski, Cynthia Breazeal, and Hae Won Park. Long-term co-design guidelines: empowering older adults as co-designers of social robots. In *2021 30th IEEE International Conference on Robot & Human Interactive Communication (RO-MAN)*, pages 1165–1172. IEEE, 2021.

[231] Kate Pahl and Sally Kelly. Family literacy as a third space between home and school: Some case studies of practice. *Literacy*, 39(2):91–96, 2005.

[232] Stamatis Papadakis. The impact of coding apps to support young children in computational thinking and computational fluency. a literature review. In *Frontiers in Education*, volume 6, page 657895. Frontiers Media SA, 2021.

[233] Seymour Papert. *Mindstorms: Children, computers, and powerful ideas*. Basic Books, Inc., 1980.

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

[235] Hae Won Park, Rinat B Rosenberg-Kima, Maor Rosenberg, Goren Gordon, and Cynthia Breazeal. Growing growth mindset with a social robot peer. In *HRI*, pages 137–145, 2017.

[236] Michael Quinn Patton. *Qualitative evaluation and research methods*. SAGE Publications, inc, 1990.

[237] Sharoda A Paul and Madhu C Reddy. Understanding together: sensemaking in collaborative information seeking. In *Proceedings of the 2010 ACM conference on Computer supported cooperative work*, pages 321–330, 2010.

[238] Blakeley H Payne. An ethics of artificial intelligence curriculum for middle school students. *MIT Media Lab Personal Robots Group. Retrieved Oct*, 10:2019, 2019.

[239] William Christopher Payne, Yoav Bergner, Mary Etta West, Carlie Charp, R Benjamin Benjamin Shapiro, Danielle Albers Szafir, Edd V Taylor, and Kayla DesPortes. danceON: Culturally responsive creative computing. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, number Article 96 in CHI '21, pages 1–16, New York, NY, USA, May 2021. Association for Computing Machinery.

[240] William Christopher Payne, Yoav Bergner, Mary Etta West, Carlie Charp, R Benjamin Benjamin Shapiro, Danielle Albers Szafir, Edd V Taylor, and Kayla DesPortes. danceON: Culturally responsive creative computing. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, number Article 96 in CHI '21, pages 1–16, New York, NY, USA, May 2021. Association for Computing Machinery.

[241] Rachel Rabkin Peachman. Mattel pulls aristotle children's device after privacy concerns. "https://www.nytimes.com/2017/10/05/well/family/mattel-aristotle-privacy.html", year = 2017, note = "[Online; accessed 19-January-2020]".

[242] Michael A Peters, Simon Marginson, and Peter Murphy. *Creativity and the global knowledge economy*. Peter Lang, 2009.

[243] Martin Porcheron, Joel E Fischer, Stuart Reeves, and Sarah Sharples. Voice interfaces in everyday life. In *proceedings of the 2018 CHI conference on human factors in computing systems*, page 640. ACM, 2018.

[244] Tom Pringle and Eden Zoller. How to achieve ai maturity and why it matters. *Ovum, June (2018), Available on Internet: https://www. amdocs. com (22/02/2019)*, 2018.

[245] Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. Robust speech recognition via large-scale weak supervision. *arXiv preprint arXiv:2212.04356*, 2022.

[246] Hayes Raffle, Rafael Ballagas, Glenda Revelle, Hiroshi Horii, Sean Follmer, Janet Go, Emily Reardon, Koichi Mori, Joseph Kaye, and Mirjana Spasojevic. Family story play: reading with young children (and elmo) over a distance. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1583–1592, 2010.

[247] Yim Register and Amy J. Ko. Learning machine learning with personal data helps stakeholders ground advocacy arguments in model mechanics. In *Proceedings of the 2020 ACM Conference on International Computing Education Research*, ICER '20, page 67–78, New York, NY, USA, 2020. Association for Computing Machinery.

[248] Replit. Ghostwriter - code faster with ai - replit. https://replit.com/site/ghostwriter, February 2023. (Accessed on 02/12/2023).

[249] Replit. Introducing,ghostwriter,chat. https://introducing-ghostwriter-chat.util.repl.co/, February 2023. (Accessed on 02/15/2023).

[250] Lauren B Resnick, Merrilee Salmon, Colleen M Zeitz, Sheila Haley Wathen, and Mark Holowchak. Reasoning in conversation. *Cognition and instruction*, 11(3-4):347–364, 1993.

[251] Mitchel Resnick, Robbie Berg, and Michael Eisenberg. Beyond black boxes: Bringing transparency and aesthetics back to scientific investigation. *The Journal of the Learning Sciences*, 9(1):7–30, 2000.

[252] Mitchel Resnick, Amy Bruckman, and Fred Martin. Pianos not stereos: Creating computational construction kits. 1996.

[253] Dean Riddlesden and Alex D Singleton. Broadband speed equity: A new digital divide? *Applied Geography*, 52:25–33, 2014.

[254] Gregorio Robles, Jesús Moreno-León, Efthimia Aivaloglou, and Felienne Hermans. Software clones in scratch projects: On the presence of copy-and-paste in computational thinking learning. In *2017 IEEE 11th International Workshop on Software Clones (IWSC)*, pages 1–7. IEEE, 2017.

[255] Barbara Rogoff, Behnosh Najafi, and Rebeca Mejía-Arauz. Constellations of cultural practices across generations: Indigenous american heritage and learning by observing and pitching in. *Human Development*, 57(2-3):82–95, 2014.

[256] Ricarose Roque, Karina Lin, and Richard Liuzzi. Engaging parents as creative learning partners in computing. *Exploring the Material Conditions of Learning*, 2:687–688, 2015.

[257] Ricarose Roque, Karina Lin, and Richard Liuzzi. "i'm not just a mom": Parents developing multiple roles in creative computing. Singapore: International Society of the Learning Sciences, 2016.

[258] Ricarose Roque, Karina Lin, and Richard Liuzzi. “I’m Not Just a Mom”: Parents Developing Multiple Roles in Creative Computing. Singapore: International Society of the Learning Sciences, 2016.

[259] Steven I Ross, Michael Muller, Fernando Martinez, Stephanie Houde, and Justin D Weisz. A case study in engineering a conversational programming assistant’s persona. *arXiv preprint arXiv:2301.10016*, 2023.

[260] Sherry Ruan, Jiayu He, Rui Ying, Jonathan Burkle, Dunia Hakim, Anna Wang, Yufeng Yin, Lily Zhou, Qianyao Xu, Abdallah AbuHashem, et al. Supporting children’s math learning with feedback-augmented narrative technology. In *Proceedings of the Interaction Design and Children Conference*, pages 567–580, 2020.

[261] Sherry Ruan, Jiayu He, Rui Ying, Jonathan Burkle, Dunia Hakim, Anna Wang, Yufeng Yin, Lily Zhou, Qianyao Xu, Abdallah Abu Hashem, Griffin Dietz, Elizabeth L Murnane, Emma Brunskill, and James A Landay. Supporting children’s math learning with feedback-augmented narrative technology. In *Proceedings of the Interaction Design and Children Conference*, IDC ’20, pages 567–580, New York, NY, USA, June 2020. Association for Computing Machinery.

[262] Sherry Ruan, Liwei Jiang, Justin Xu, Bryce Joe-Kun Tham, Zhengneng Qiu, Yeshuang Zhu, Elizabeth L Murnane, Emma Brunskill, and James A Landay. Quizbot: A dialogue-based adaptive learning system for factual knowledge. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, pages 1–13, 2019.

[263] Andee Rubin. Learning to reason with data: How did we get here and what do we know? *Journal of the Learning Sciences*, 00(00):1–11, 2019.

[264] Rosemary S Russ, Bruce L Sherin, and Miriam Gamoran Sherin. What constitutes teacher learning. *Handbook of research on teaching*, pages 391–438, 2016.

[265] Heleen Rutjes, Martijn Willemsen, and Wijnand IJsselsteijn. Considerations on explainable ai and users’ mental models. 2019.

[266] Druga S. and Verzhbinsky D. stefania11/pippisynth: class project for aut21 507. https://github.com/stefania11/pippisynth, September 2022. (Accessed on 02/15/2023).

[267] Alpay Sabuncuoglu. Designing one year curriculum to teach artificial intelligence for middle school. In *Proceedings of the 2020 ACM Conference on Innovation and Technology in Computer Science Education*, pages 96–102, 2020.

[268] Herman Saksono, Ashwini Ranade, Geeta Kamarthi, Carmen Castaneda-Sceppa, Jessica A Hoffman, Cathy Wirth, and Andrea G Parker. Spaceship launch: Designing a collaborative exergame for families. In *Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing*, pages 1776–1787, 2015.

[269] Donald P Sanders and Gail McCutcheon. The development of practical theories of teaching. *Journal of Curriculum and Supervision*, 2(1):50–67, 1986.

[270] Ismaila Temitayo Sanusi, Kissinger Sunday, Solomon Sunday Oyelere, Jarkko Suhonen, Henriikka Vartiainen, and Markku Tukiainen. Learning machine learning with young children: exploring informal settings in an african context. *Computer Science Education*, pages 1–32, 2023.

[271] Michael Scaife and Mike Duuren. Do computers have brains? what children believe about intelligent artifacts. *British Journal of Developmental Psychology*, 13(4):367–377, 1995.

[272] Baruch B Schwarz. Argumentation and learning. In *Argumentation and education*, pages 91–126. Springer, 2009.

[273] Fiona Louise Scott. Family mediation of preschool children's digital media practices at home. *Learning, Media and Technology*, pages 1–16, July 2021.

[274] Serenade. Serenade — code with voice. https://serenade.ai/, February 2023. (Accessed on 02/15/2023).

[275] David Williamson Shaffer and Mitchel Resnick. " thick" authenticity: New media and authentic learning. *Journal of interactive learning research*, 10(2):195–216, 1999.

[276] Xiaojing Shang, Zhujun Jiang, Feng-Kuang Chiang, Yicong Zhang, and Dan Zhu. Effects of robotics stem camps on rural elementary students' self-efficacy and computational thinking. *Educational technology research and development*, pages 1–26, 2023.

[277] Rashi Shrivastava. Amazon alexa wants to put your child to bed with generative ai storytelling. https://www.forbes.com/sites/rashishrivastava/2022/12/07/amazon-alexa-wants-to-put-your-child-to-bed-with-generative-ai-storytelling/?sh=2844aa87b5b9, January 2023. (Accessed on 02/15/2023).

[278] Lee S Shulman. Pck: Its genesis and exodus. In *Re-examining pedagogical content knowledge in science education*, pages 13–23. Routledge, 2015.

[279] Robert S Siegler. Improving the numerical understanding of children from low-income families. *Child Development Perspectives*, 3(2):118–124, 2009.

[280] Roger Silverstone and Eric Hirsch. *Consuming technologies*. Taylor & Francis, 1994.

[281] Rishabh Singh, Sumit Gulwani, and Armando Solar-Lezama. Automated feedback generation for introductory programming assignments. In Hans-Juergen Boehm and Cormac Flanagan, editors, *ACM SIGPLAN Conference on Programming Language Design and Implementation, PLDI '13, Seattle, WA, USA, June 16-19, 2013*, pages 15–26. ACM, 2013.

[282] Bryan A Sisk, Alison L Antes, Sara Burrous, and James M DuBois. Parental attitudes toward artificial intelligence-driven precision medicine technologies in pediatric healthcare. *Children*, 7(9):145, 2020.

[283] Sinem Siyahhan, Sasha A Barab, and Michael P Downton. Using activity theory to understand intergenerational play: The case of family quest. *International Journal of Computer-Supported Collaborative Learning*, 5(4):415–432, 2010.

[284] Allison Skerrett. Lolita, facebook, and the third space of literacy teacher education. *Educational Studies*, 46(1):67–84, 2010.

[285] Zoe Skinner, Stacey Brown, and Greg Walsh. Children of color's perceptions of fairness in ai: An exploration of equitable and inclusive co-design. In *Extended Abstracts of the 2020 CHI Conference on Human Factors in Computing Systems*, pages 1–8, 2020.

[286] Anna Smith. From mimeographs to new learning ecologies. In *Critical Digital Literacies: Boundary-Crossing Practices*, pages 224–230. Brill, 2021.

[287] David M Sobel, Joshua B Tenenbaum, and Alison Gopnik. Children's causal inferences from indirect evidence: Backwards blocking and bayesian reasoning in preschoolers. *Cognitive science*, 28(3):303–333, 2004.

[288] Elliot M Soloway and Beverly Woolf. Problems, plans, and programs. *ACM SIGCSE Bulletin*, 12(1):16–24, 1980.

[289] Papadakis Stamatios. Can preschoolers learn computational thinking and coding skills with scratchjr? a systematic literature review. *International Journal of Educational Reform*, page 10567879221076077, 2022.

[290] Statista. Digital assistant and voice assistant adoption 2019. 2019.

[291] Reed Stevens and L. Takeuchi. *The New Coviewing: Designing for Learning through Joint Media Engagement*. The Joan Ganz Cooney Center, 2011.

[292] Jiahong Su and Yuchun Zhong. Artificial intelligence (ai) in early childhood education: Curriculum design and future directions. *Computers and Education: Artificial Intelligence*, 3:100072, 2022.

[293] Yingze Sun, Matthew Aylett, and Yolanda Vazquez-Alvarez. e-seesaw: A tangible, ludic, parent-child,awareness system. In *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems- CHI EA '16*, pages 1821–1827, 2016.

[294] Lori Takeuchi, Reed Stevens, et al. The new coviewing: Designing for learning through joint media engagement. In *New York, NY: The Joan Ganz Cooney Center at Sesame Workshop*, 2011.

[295] Fumihide Tanaka, Aaron Cicourel, and Javier R Movellan. Socialization between toddlers and robots at an early childhood education center. *Proceedings of the National Academy of Sciences*, 104(46):17954–17958, 2007.

[296] Chaoying Tang, Qihong Duan, and Haiying Long. How do parents influence student creativity? evidence from a large-scale survey in china. *Thinking Skills and Creativity*, 46:101134, 2022.

[297] Steven L Tanimoto. A perspective on the evolution of live programming. In *2013 1st International Workshop on Live Programming (LIVE)*, pages 31–34. IEEE, 2013.

[298] Emmeline Taylor and Katina Michael. Smart toys that are the stuff of nightmares. *IEEE Technology and Society Magazine*, 35(1):8–10, 2016.

[299] Tactical Tech. Tactical tech – [open call for partnerships] organizations working on media and digital literacy initiatives for youth across europe. https://tacticaltech.org/news/call-partnerships-digital-literacy-youth, February 2023. (Accessed on 02/10/2023).

[300] Technovation. Ai education for families — technovation families. https://www.curiositymachine.org/, February 2023. (Accessed on 02/11/2023).

[301] Matti Tedre, Tapani Toivonen, Juho Kaihila, Henriikka Vartiainen, Teemu Valtonen, Ilkka Jormanainen, and Arnold Pears. Teaching machine learning in k-12 computing education: Potential and pitfalls. *arXiv preprint arXiv:2106.11034*, 2021.

[302] RC Teepe, Inge Molenaar, and Ludo Verhoeven. Technology-enhanced storytelling stimulating parent–child interaction and preschool children's vocabulary knowledge. *Journal of Computer Assisted Learning*, 33(2):123–136, 2017.

[303] Pamela Tierney and Steven M Farmer. Creative Self-Efficacy: Its potential antecedents and relationship to creative performance. *Acad. Manage. J.*, 45(6):1137–1148, 2002.

[304] David Touretzky, Christina Gardner-McCune, Fred Martin, and Deborah Seehorn. Envisioning ai for k-12: What should every child know about ai? In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 9795–9799, 2019.

[305] David Touretzky, Christina Gardner-McCune, Fred Martin, and Deborah Seehorn. Envisioning ai for k-12: What should every child know about ai? In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 9795–9799, 2019.

[306] Giovanni Maria Troiano, Qinyu Chen, Ángela Vargas Alba, Gregorio Robles, Gillian Smith, Michael Cassidy, Eli Tucker-Raymond, Gillian Puttick, and Casper Harteveld. Exploring how game genre in Student-Designed games influences computational thinking development. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, CHI '20, pages 1–17, New York, NY, USA, April 2020. Association for Computing Machinery.

[307] Moran Tsur and Natalie Rusk. Scratch microworlds: Designing Project-Based introductions to coding. In *Proceedings of the 49th ACM Technical Symposium on Computer Science Education*, SIGCSE '18, pages 894–899, New York, NY, USA, February 2018. Association for Computing Machinery.

[308] S Turkle. Tamagotchi diary. *The London Review of Books, April*, 20, 2006.

[309] Sherry Turkle. *Evocative Objects: Things We Think With.* MIT Press, September 2011.

[310] Sherry Turkle, Cynthia Breazeal, Olivia Dasté, and Brian Scassellati. Encounters with kismet and cog: Children respond to relational artifacts. *Digital media: Transformations in human communication*, pages 1–20, 2006.

[311] Sherry Turkle and Seymour Papert. Epistemological pluralism and the revaluation of the concrete. *Journal of Mathematical Behavior*, 11(1):3–33, 1992.

[312] Priyan Vaithilingam, Tianyi Zhang, and Elena L Glassman. Expectation vs. experience: Evaluating the usability of code generation tools powered by large language models. In *Chi conference on human factors in computing systems extended abstracts*, pages 1–7, 2022.

[313] Junia Valente and Alvaro A Cardenas. Security & privacy in smart toys. In *Proceedings of the 2017 Workshop on Internet of Things Security and Privacy*, pages 19–24, 2017.

[314] Jan AGM Van Dijk. Digital divide research, achievements and shortcomings. *Poetics*, 34(4-5):221–235, 2006.

[315] Mike van Duuren and Michael Scaife. How do children represent intelligent technology? *European Journal of Psychology of Education*, 10(3):289–301, 1995.

[316] Henriikka Vartiainen, Matti Tedre, and Teemu Valtonen. Learning machine learning with very young children: Who is teaching whom? *International journal of child-computer interaction*, 25:100182, September 2020.

[317] Anu Vazhayil, Radhika Shetty, Rao R Bhavani, and Nagarajan Akshay. Focusing on teacher education to introduce ai in schools: Perspectives and illustrative findings. In *2019 IEEE Tenth International Conference on Technology for Education (T4E)*, pages 71–77. IEEE, 2019.

[318] Caitlyn Vergara. Tech layoffs affecting your kid's digital ecosystem — by @altbriefs — feb, 2023 — medium. https://medium.com/@ALTbriefs/tech-layoffs-affecting-your-kids-digital-ecosystem-4483d7abce36, February 2023. (Accessed on 02/10/2023).

[319] Sara Vogel, Christopher Hoadley, Ana Rebeca Castillo, and Laura Ascenzi-Moreno. Languages, literacies and literate programming: can we use the latest theories on how bilingual people learn to help us teach computational literacies? *Computer Science Education*, 30(4):420–443, 2020.

[320] Anna-Lisa Vollmer, Robin Read, Dries Trippas, and Tony Belpaeme. Children conform, adults resist: A robot group induced peer pressure on normative social conformity. *Science Robotics*, 3(21):eaat7111, 2018.

[321] Christiane Gresse von Wangenheim, Jean CR Hauck, Fernando S Pacheco, and Matheus F Bertonceli Bueno. Visual tools for teaching machine learning in k-12: A ten-year systematic mapping. *Education and Information Technologies*, pages 1–46, 2021.

[322] Shirin Vossoughi, Paula K Hooper, and Meg Escudé. Making through the lens of culture and power: Toward transformative visions for educational equity. *Harvard Educational Review*, 86(2):206–232, 2016.

[323] Lev Vygotsky. Interaction between learning and development. *Readings on the development of children*, 23(3):34–41, 1978.

[324] Lev Semenovich Vygotsky. Imagination and creativity in childhood. *Journal of Russian & East European Psychology*, 42(1):7–97, January 2004.

[325] Sandra Wachter, Brent Mittelstadt, and Chris Russell. Counterfactual explanations without opening the black box: Automated decisions and the gdpr. *Harv. JL & Tech.*, 31:841, 2017.

[326] Lawrence J Walker, Karl H Hennig, and Tobias Krettenauer. Parent and peer contexts for children's moral reasoning development. *Child development*, 71(4):1033–1048, 2000.

[327] G Walsh, E Foss, J Yip, and A Druin. FACIT PD: a framework for analysis and creation of intergenerational techniques for participatory design. *of the SIGCHI Conference on Human ...*, 2013.

[328] Greg Walsh, Elizabeth Foss, Jason Yip, and Allison Druin. Facit pd: a framework for analysis and creation of intergenerational techniques for participatory design. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 2893–2902, 2013.

[329] Mark Warschauer and Morgan Ames. Can one laptop per child save the world's poor? *Journal of international affairs*, pages 33–51, 2010.

[330] Karl E Weick. *Sensemaking in organizations*, volume 3. Sage, 1995.

[331] David Weintrop and Uri Wilensky. To block or not to block, that is the question: students' perceptions of blocks-based programming. In *Proceedings of the 14th international conference on interaction design and children*, pages 199–208, 2015.

[332] Michelle Hoda Wilkerson-Jerde and Uri J Wilensky. Patterns, probabilities, and people: Making sense of quantitative change in complex systems. *Journal of the Learning Sciences*, 24(2):204–251, 2015.

[333] Randi Williams, Christian Vázquez Machado, Stefania Druga, Cynthia Breazeal, and Pattie Maes. My doll says it's ok: a study of children's conformity to a talking doll. In *Proceedings of the 17th ACM Conference on Interaction Design and Children*, pages 625–631. ACM, 2018.

[334] Simon Willison. Bing: "i will not harm you unless you harm me first". https://simonwillison.net/2023/Feb/15/bing/, February 2023. (Accessed on 02/15/2023).

[335] Julia Woodward, Zari McFadden, Nicole Shiver, Amir Ben-hayon, Jason C Yip, and Lisa Anthony. Using co-design to examine how children conceptualize intelligent interfaces. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, pages 1–14, 2018.

[336] Lu Xiao and Jennifer Martin. Supporting parent-young child activities with interactive tabletops: a conceptual analysis. In *Proceedings of the ACM 2012 conference on Computer Supported Cooperative Work Companion*, pages 301–310, 2012.

[337] Kun Xu and Matthew Lombard. Persuasive computing: Feeling peer pressure from multiple computer agents. *Computers in Human Behavior*, 74:152–162, 2017.

[338] Weixiang Yan and Yuanchun Li. Whygen: explaining ml-powered code generation by referring to training examples. In *Proceedings of the ACM/IEEE 44th International Conference on Software Engineering: Companion Proceedings*, pages 237–241, 2022.

[339] Sarita Yardi and Amy Bruckman. Income, race, and class: exploring socioeconomic differences in family technology use. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 3041–3050, 2012.

[340] Svetlana Yarosh. Supporting long-distance parent-child interaction in divorced families. In *CHI'08 Extended Abstracts on Human Factors in Computing Systems*, pages 3795–3800. 2008.

[341] YING.H. Ai knitwear - ying.h. https://yunyingh.com/AI-Knitwear, 2023. (Accessed on 02/15/2023).

[342] Jason C Yip, Kiley Sobel, Caroline Pitt, Kung Jin Lee, Sijin Chen, Kari Nasu, and Laura R Pina. Examining adult-child interactions in intergenerational participatory design. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, pages 5742–5754. ACM, 2017.

[343] J Yu, C Bai, and R Roque. Considering parents in coding kit design: Understanding parents' perspectives and roles. *Proceedings of the 2020 CHI Conference on*, 2020.

[344] Junnan Yu, Chenke Bai, and Ricarose Roque. Considering Parents in Coding Kit Design: Understanding Parents' Perspectives and Roles. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, pages 1–14, Honolulu HI USA, April 2020. ACM.

[345] Daoguang Zan, Bei Chen, Fengji Zhang, Dianjie Lu, Bingchao Wu, Bei Guan, Yongji Wang, and Jian-Guang Lou. When neural model meets nl2code: A survey. *arXiv preprint arXiv:2212.09420*, 2022.

[346] Eric Zeng, Shrirang Mare, and Franziska Roesner. End user security and privacy concerns with smart homes. In *Thirteenth Symposium on Usable Privacy and Security ({SOUPS} 2017)*, pages 65–80, 2017.

[347] Chao Zhang, Cheng Yao, Jianhui Liu, Zili Zhou, Weilin Zhang, Lijuan Liu, Fangtian Ying, Yijun Zhao, and Guanyun Wang. StoryDrawer: A Co-Creative agent supporting children's storytelling through collaborative drawing. In *Extended Abstracts of the 2021 CHI Conference on Human Factors in Computing Systems*, pages 1–6. Association for Computing Machinery, New York, NY, USA, May 2021.

[348] Xiaofei Zhou, Jessica Van Brummelen, and Phoebe Lin. Designing ai learning experiences for k-12: Emerging works, future opportunities and a design framework. *arXiv preprint arXiv:2009.10228*, 2020.

[349] Barry J Zimmerman. Self-Regulated learning and academic achievement: An overview. *Educ. Psychol.*, 25(1):3–17, January 1990.

[350] Abigail Zimmermann-Niefield, Makenna Turner, Bridget Murphy, Shaun K Kane, and R Benjamin 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.

Chapter 10

APPENDIX