High School Youth PEER AUDITING of Machine Learning Applications to Promote Computational Literacies

Yasmin B. Kafai & Luis Morales-Navarro University of Pennsylvania





Youth interact with AI/ML applications everyday

- Through their daily interactions they develop some understandings about how that ML applications learn from data and how applications recognize patterns
- Their everyday ideas have implications for the design of tools and activities to introduce youths to ML





Morales-Navarro, L., & Kafai, Y. B. (2024). Investigating Youths' Everyday Understanding of Machine Learning Applications: a Knowledge-in-Pieces Perspective. Proceedings of the 18th International Conference of the Learning Sciences - ICLS 2024. Buffalo, NY: International Society of the Learning Sciences.

SEE ALSO:

Coenraad, M. (2022).

Salac, J., Oleson, A., Armstrong, L., Le Meur, A., & Ko, A. J. (2023, August). Solyst, J., Xie, S., Yang, E., Stewart, A. E., Eslami, M., Hammer, J., & Ogan, A. (2023a). Solyst, J., Yang, E., Xie, S., Ogan, A., Hammer, J., & Eslami, M. (2023b







OPACITY of how models work

Empirical and inductive NATURE of development

Challenge

"The challenge facing many young people today is that they generally speaking have a limited understanding of technology and computing, not only in terms of its construction but how it affects their lives. Hence young people have very limited capacity to pose demands for technology, make informed choices about technology in their lives, and take part in the development of technology and the cultures that surround it." Dindler, Iverson Caspersen & Smith, 2022, p. 120



AI Literacies

Algorithm Auditing

PEER AUDITING WORKSHOP & FINDINGS

Discussion

Next Steps

Computational Literacies

REVIEWS/ESSAYS

A Revaluation of Computational Thinking in K-12 Education: Moving Toward Computational Literacies Yasmin B. Kafai O and Chris Proctor 0

One the past decide initiatives around the world have introduced computing into K-12 education under the under the C sempetational thirding. Write in tail implementations focused on abilis and to overadge for college and caver readiness. more recent memiring include shaded computetional trinking (identity, participation, contive expression) and official computer and trinking politics and othical impacts of contracting, periody. This repare in others a multiplice of whith monitories represent for computationally, herein in the 21st contrary. We chick the purper landscape of 6, 12 computing education chease interactions between of Yearshift analogs of comparational duriangs and schools how on an compare high Transverb or computations. There is a during the importance of computing for broader K-12 educational prior like serve ne ony attend out issues.

Reveards: computer and learning: computer science education: in tool theory; identity, literaty

In the January 2015 issue of Videouband Massacher, Vienner K-12 education, noting that it was builded whose time had 36. Competational thinking defined by Jamene Wing (2003) as "unobeing robeing problems, designing systems and understanding furnish behavior that draws on consepts furdamental to computing" (p. 33)-we described to a key moreutter for bringing componentationer (CN) back into schools. Comparational denient's monum mon centrally came has a month later when the orders "What Most Schriek Durit" Jeach" (2013) was released on VeriTube, informing millions of sleavers that today's children need to beam about CS. A large end of colorities, enough two Viewahl bunder Sil Cave, bekehall your Chris Bosch, and reak moletim will tare percend the came. In the sideo, a quote by the late Sixye Jobs, colounder, of Arek, made the connection to computational thinking by demanding that Asseybody in this country should lister how toregram a comprise ... hactuse it reaches you have to think." Take the same year the Hoar of Code wis handled during Comparer Science Istingation Work, going in Bors of 8, 12, stations that fire cars of programming. Fight years have, over one billion students around the world inverprising of the COLUMN AVAIL

and Perroversial the same of comparational dataking within a lack of diversity in the field (Wilson et al., 2010). With CS to an decay art conting used high school gadaction. Icy sea dents were tabling AP occurs when they were offered or all (Magolis et al. 2012). Neiller ander preparation program out petitionians were available, and school districts had on financeously or soundards to choose Not Jacoward to 2021, the number of crudents taking AP courses and participating in the scan has dynificantly increased through atlinos couching parin potent comparable to other popular AP once such as state too, mother contribution standards and programs are being catal/field, and many school districts new dours CS courses as guaration reprisentant. By all massives, this is egolicant proyeet even theory's much work still remains to be done in induity CS arrentinged part of the K-12 education system lags. Deleter in al., 2018). lasked, concrutational limiting his become part of a global

effort under which the introduction of CS education in K. 1204 premiered maker different chreetivers, most poormieredy to mainain each country's economic competitivoness (1 issiend , 2013) has also to develop a new cross-disciplinary its range "... to saidup young and infinities we should add computational

> Educational Researcher, Web 51, Vol. 2, pp. 145-151 Article realise guide her regappiblicer (our mil

Does descriptions developments gives value reports distat. Encourse of Parasitions, 20 dedgin, 25, rate district states of K. 12, CS, effective, which dearmented. Discussion will be 2019, 2019, 2019 a St.

146 PERMITTING BUILDING

We define *computational literacies* as a set of practices situated in a sociocultural context which utilize external computational media to support cognition and communication. Computational literacies encompass phenomena at scales from the individual to the societal, as well as connections between these phenomena and the media which supports and shapes them....

In our view, the question of what to teach in K-12 CS need not have a single answer, but could instead have many answers grounded in the computational literacy practices of diverse communities and cultures

Kafai & Proctor, 2022, p.148



Kafai, Y. B., & Proctor, C. (2022). A revaluation of computational thinking in K-12 education: Moving toward computational literacies. Educational Researcher, 51(2), 146-151.

Five Big Ideas

Object ID:

COCIETAL IMO

mpact socie the and negative

3 - LEARNING

Five Big Ideas in Artificial Intelligence

5. Societal Impact

AI can impact society in both positive and negative ways. Al technologies are changing the ways we work, travel, communicate, and care for each other. But we must be mindful of the harms that can potentially occur. For example, biases in the data used to train an AI system could lead to some people being less well served than others. Thus, it is important to discuss the impacts that AI is having on our society and develop criteria for the ethical design and deployment of Al-based systems.

4. Natural Interaction

HATURAL INTERACTION Humans are among the hardest things for AI agents to understand. Intelligent agents require many kinds of knowlege to interact naturally with humans. Agents must be able to converse in human languages, recognize facial expressions and emotions, and draw Computers can learn from data upon knowledge of culture and social conventions to infer intentions from observed behavior. Today's Al systems can use language to a limited extent, but lack the general reasoning and conversational capabilities of even a child.

The Al for K-12 in Rightwe is a joint project of the Association for the Advancement of Artificial Intel Rigence (AAA) and the Computer Science Teachers Association (CSIR), funded by National Science Foe edution several DRL-1846073

1. Perception

Accuracy

Computers perceive the world using sensors. Perception is the process of extracting meaning from sensory signals. Making computers "see" and "hear" well enough for practical use is one of the most significant achievements of AI to computers perceive the world using sense date.

2. Representation & Reasoning

Agents maintain representations of the world and use them for reasoning. Representation is one of the fundamental problems of intelligence, both natural and artificial, Computers construct representations using data structures, and these representations support reasoning algorithms that derive new information from what is already known. While Al agents can reason about very complex problems, they do not think the way a human does.

Learning

REPRESENTATION & REACONING Computers can learn from data, Machine learning is a kind of statistical inference that finds patterns in data. Many areas of Al have progressed significantly in recent years thanks to learning algorithms that create new representations. For the 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.

AI4K12



Computational Thinking 2.0

Koli Calling '21, November 18-21, 2021, Joensuu, Finland

Tedre, Denning & Toivonen

Table 1: Comparison of educational concerns in traditional programming education vs. education for creating ML and other data-driven models.

	CT1.0	CT 2.0	
Problem solving: Stage 1	Formalize the problem	Collect data from the intended context	
Problem solving: Stage 2	Design a solution	Filter and clean data. Label data.	
Problem solving: Stage 3	Implement the solution in a stepwise program	Train a model from the available data	
Problem solving: Stage 4	Compile and execute the program	Evaluate and use the model	
Universality of solution	Weakly context-dependent	Strongly context-dependent	
Goodness of solution	In some cases clearly works or doesn't. Can be for-	Models may display higher or lower confidence. F	
	mally proven to be either correct or incorrect (at	ficiency can be established through testing. Statisti-	
	advanced levels). Effectiveness can be proven.	cally better or worse (at advanced levels).	
Testing	Black-boxed or glass-boxed cross checking of the	Evaluate the model against predictions, completely	
	outputs and the program code	black boxed	
Debugging	Tracking and tracing program states and code for	Experimenting with data, parameters, and hyperpa-	
	error.	rameters, based on trial and error	
Philosophy of problem	Deductive	Inductive	
solving			
Structure	Transparent. Visualization tools available.	Black boxed	
Notional machines	Stepwise, deterministic, discrete flow of program	Parallel, possibly nondeterministic, passing data	
	through states (as contents of memory locations).	through a network	
Complexity concerns	Prepare for worst case, optimize for average case	No time / space variance between passing data	
		through a network	
Portability	Tedious to make portable to different platforms.	Straightforwardly portable	
Trial and error	Discouraged	Necessary	
Software life cycle	Traditional, well established life cycle. Clear version-	More data create new "versions". Documenting is	
	ing.	based on empiricism and reporting of training data.	
Syntax and semantics	Syntactically strict, highly structured	Data can be unstructured, loose semantics	

Matti Tedre University of Eastern Finland

CT 2.0 Peter J. Denning. pide)reprocedu Neval Postgraduate School



Figure 1 Children consting a gesture receptation model on an image recognition system. (CDFR compliant photo.) ABSTRACT training data, and minferoment learning that are not part of CT at all. All these changes challenge the traditional views scherel to maching CT in K-12. ML is not the only emerging technology appending in the con-

CT has been the central software available F. U computing education. at loss energies the radio limits of the scheme scheme independent states, and policy makers have pointed the movement, A consensus has enough over the conceptual landscape of $C\Sigma$. Meanwhile, machine beausing the l, his triggered scase stope

States rates, sancher maring (16.), et al. program and operating and the second states of the - Social and prefessional topics -- Computational thinking: K-12 education. KEYWORDS Compension. Look at series of the changes ML has silvenly triansred in comput-

Competitional thinking, E-12, Marline Intening, Artificial Intelliarray, School

Computing education: Com-

puting landscope. Quantum computing and biological computing are not far behind. We need to start exhibiting how CT must evolve to anticipate and most these challenges.

Tamani Toivonani

University of Eastern Pinland

taparal.tole onem@act.ft

Look a new of the charges AD, has should triggered in compan-ing. It has cauled gravity improved speech and image recognition, purseful recommunications on stranding services, automatance rangestion of cars, napse-imana performance in board and com-puter games, and seven directative-reality "deepfide" videos. Most advances in topics aloves are the to hardware cerelation to aco-ACM Enforcement Research Interf Tolon, Denn J., Danning, and Tapinal Tolevanna. 2021. CT 201. In 23:07 Exhibit Calling Marinesine Conference on Computing Education Research & Ed-Calling '259 Non-online 17:42, 2021, Jonaton, Furdand. ACM, New York, NY, USA, 5 Pages 1: Improvidence of a 11:1051000602 20080051

the rules for competing a function, machine loarning OID gets a

Advances strapping dama and a fair fair fair strapping dama and a strapp The second secon

16 Competencies

CHI 2020 Paper

CHI 2020, April 25-30, 2020, Honolulu, HI, USA

What is AI Literacy? Competencies and Design Considerations

 Duri Long
 Brian Magerko

 Georgia Institute of Technology
 Georgia Institute of Technology

 Adlanta, GA, USA
 Atlanta, GA, USA

 duri@gatech.edu
 magerko@gatech.edu

ABSTRACT

Artificial intelligence (AI) is becoming increasingly integrated in user-facing technology, but public understanding of these technologies is often limited. There is a need for additional HCI research investigating a) what competencies users need in order to effectively interact with and critically evaluate AI and b) how to design learnercentered AI technologies that foster increased user understanding of AI. This paper takes a step towards realizing both of these goals by providing a concrete definition of AI literacy based on existing research. We synthesize a variety of interdisciplinary literature into a set of core competencies of AI literacy and suggest several design considerations to support AI developers and educators in creating learner-centered AI. These competencies and design considerations are organized in a conceptual framework thematically derived from the literature. This paper's contributions can be used to start a conversation about and guide future research on AI literacy within the HCI community.

Author Keywords

AI literacy, AI education; AI for K-12; artificial intelligence; machine learning, computing education

CSS CONCEPTS

General and reference-Surveys and overviews
 Social and professional topics-Computing literacy
 Computing methodologies-Artificial intelligence

INTRODUCTION

CHI '20, April 25–90, 2029, Honolulin, HL USA © 2020 Copyright is held by the owner function 'Publication rights licensed to ACM ACM 978-1-4030-6706-020 04, 351:00 https://doi.org/10.2114/15513831.3376227 Design and education both play a role in contributing to public missinderstandings about AI. Black-box algorithms to e. algorithms with obscured mone-workings) can cause misunderstandings about AI [55]. On the other hand—even knowledge on the part of the user can lead to miscenceptions [25]. There is a clear need for a better understanding of this space from the propertieves to buth learners and designers.

Researchers in the HCI community have begun to address public micrococytons of AI by unvertigating how people make sense of AI (e.g. 4(d)) and exploring how to design more understandable technology (e.g. 4(2)). However, there is a need for additional research unvestigating what new transforms the way that we communicate, work, and hive with each other and with machines. We refer to this set of competencies as AI *literary*.

Emerging research is exploring how to foster AI literacy in audiences without technical backgrounds. Within the past year, companies have pursued initiatives to broaden AI education to underrepresented auchences in an effort to increase workforce diversity [5,148], educators have published guides on how to incorporate AI into K-12 curricula [145], and researchers are exploring how to engage young learners in creative programming activities involving AI [45,79,132,146,149]. The "AI for K12" working group is currently developing a set of standards for K-12 classrooms to determine what each grade band should know about AI [130]. The group has also identified five "big ideas" of AI to guide the standards development: 1) "Computers perceive the world using sensors"; 2) "Agents maintain models/representations of the world and use them for reasoning"; 3) "Computers can learn from data"; 4) "Making agents interact with humans is a substantial challenge for AI developers"; and 5) "AI applications can impact society in both positive and negative ways" [130].

The five 'big idea' of AI provide a strong fromdation for future research on fostering AI littency. However, most of the research on AI education for non-technical learners has just been published writin the last year. In contrast, AI as a field has been active since the 1950s, and there a a variety that could contrabute to understanding that competencies should be included in a definition of AI literary and how to better design elucational appeareaces that fowler AI literary.

Page 1

Competency 1 (Recognizing AI) Distinguish between technological artifacts that use and do not use AI.

Competency 2 (Understanding Intelligence) Critically analyze and discuss features that make an entity "intelligent", including discussing differences between human, animal, and machine intelligence.

Competency 3 (Interdisciplinarity) Recognize that there are many ways to think about and develop "intelligent" machines. Identify a variety of technologies that use AI, including technology spanning cognitive systems, robotics, and ML.

Competency 4 (General vs. Narrow) Distinguish between general and narrow AI.

Competency 5 (AI's Strengths & Weaknesses) Identify problem types that AI excels at and problems that are more challenging for AI. Use this information to determine when it is appropriate to use AI and when to leverage human skills.

Competency 6 (Imagine Future AI) Imagine possible future applications of AI and consider the effects of such applications on the world.

Competency 7 (Representations) Understand what a knowledge representation is and describe some examples of knowledge representations.

Competency 8 (Decision-Making) Recognize and describe examples of how computers reason and make decisions.

Competency 9 (ML Steps) Understand the steps involved in machine learning and the practices and challenges that each step entails.

Competency 10 (Human Role in AI) Recognize that humans play an important role in programming, choosing models, and fine-tuning AI systems.

Competency 11 (Data Literacy) Understand basic data literacy concepts such as those outlined in [107].

Competency 12 (Learning from Data) Recognize that computers often learn from data (including one's own data).

Competency 13 (Critically Interpreting Data) Understand that data cannot be taken at face-value and requires interpretation. Describe how the training examples provided in an initial dataset can affect the results of an algorithm.

Competency 14 (Action & Reaction)

Understand that some AI systems have the ability to physically act on the world. This action can be directed by higher-level reasoning (e.g. walking along a planned path) or it can be reactive (e.g. jumping backwards to avoid a sensed obstacle).

Competency 15 (Sensors)

Understand what sensors are, recognize that computers perceive the world using sensors, and identify sensors on a variety of devices. Recognize that different sensors support different types of representation and reasoning about the world.

Competency 16 (Ethics)

Identify and describe different perspectives on the key ethical issues surrounding AI (i.e. privacy, employment, misinformation, the singularity, ethical decision making, diversity, bias, transparency, accountability).

Computational Literacies

Scribner (1984) which outlines three core dimensions of literacy: **FUNCTIONAL, CRITICAL AND PERSONAL.**

Literacies of any kind always address a *functional* dimension of "the level of proficiency necessary for effective performance in a range of settings" (p. 9).

The *critical* dimension of literacy equips individuals with critical consciousness to examine conditions in their community and lead effective action for a just society.

The *personal* dimension frames becoming literate as a form of self-enhancement to better communicate, interact with others, and build relationships.

ALGORITHM AUDITING

"Repeatedly querying an algorithm and observing its output in order to draw conclusions about the algorithm's opaque inner workings and possible external impact."

Metaxa et al., 2021, p. 10

ALGORITHM AUDITING as DECODING

Deconstruction involves describing, evaluating and reflecting on the values and intentions embedded in sociotechnical systems and considering their possible implications (Dindler et al., 2020; Schaper et al., 2022). This involves

(1) Assessing the properties of a system in terms of its inputs, outputs and materials,

(2) Inquiring on the intended use and actual use of a system,

(3) Foregrounding the values, worldviews, and assumptions embedded in the system, and

(4) Impacting individuals, communities and the environment.

We argue that algorithm auditing is a method that can support learners in deconstructing AI/ML applications.

Examples Of Algorithm Auditing



housing | employment | product pricing | health | search

Algorithm Auditing in Five Steps

- Generating a hypothesis
- 2

3

4

5

1

- Generating systematic, thorough, and thoughtful inputs to test the hypothesis
- Running the test and keeping track of the inputs and output pairs
- Analyzing the data
- **Reporting findings**

Everyday audits

Toward User-Driven Algorithm Auditing: Investigating users' strategies for uncovering harmful algorithmic behavior

Alicia DeVos Carnegie Mellon University Pittsburgh, PA, USA adevos@andrew.cmu.edu Aditi Dhabalia Carnegie Mellon University Pittsburgh, PA, USA aditidhabalia@gmail.com

Kenneth Holstein* Carnegie Mellon University Pittsburgh, PA, USA kjholste@andrew.cmu.edu

ABSTRACT

Recent work in HCI suggests that users can be powerful in surfacing harmful algorithmic behaviors that formal auditing approaches fail to detect. However, it is not well understood how users are often able to be so effective, nor how we might support more effective user-driven auditing. To investigate, we conducted a series of think-aloud interviews, diary studies, and workshops, exploring how users find and make sense of harmful behaviors in algorithmic systems, both individually and collectively. Based on our findings, we present a process model capturing the dynamics of and influences on users' search and sensemaking behaviors. We find that 1) users' search strategies and interpretations are heavily guided by their personal experiences with and exposures to societal bias; and 2) collective sensemaking amongst multiple users is invaluable in user-driven algorithm audits. We offer directions for the design of future methods and tools that can better support user-driven auditing.

CCS CONCEPTS

Human-centered computing → Empirical studies in collaborative and social computing; Computer supported cooperative work; Empirical studies in HCI; + Information systems → Web searching and information discovery.

ersity Carnegie Mellon University A Pittsburgh, PA, USA .com hongs@andrew.cmu.edu Motahhare Eslami*

Hong Shen

Carnegie Mellon University Pittsburgh, PA, USA meslami@andrew.cmu.edu

on Human Factors in Computing Systems (CHI '22), April 29-May 5, 2022, New Orleans, LA, USA. ACM, New York, NY, USA, 19 pages. https://doi.org/10. 1145/3491102.3517441

1 INTRODUCTION

The presence of biases and inequities in algorithmic systems has led researchers to develop new approaches for algorithm auditing to detect biased, discriminatory, or otherwise harmful behaviors1 (e.g., [17, 22, 31, 46, 53, 59, 62, 66, 74, 82]). Typically these auditing techniques are led by experts such as researchers, activists, industry practitioners, and government agencies [20]. For example, in "scraping audit" techniques, experts query an algorithmic system and investigate and compare the outputs [66]. As another example, in a "sock puppet audit", experts use computer programs to impersonate different types of users, inject fake data into the system, and evaluate the results [66]. Although expert-led auditing approaches have been greatly impactful, the absence of the actual context of use and everyday users in the auditing process can still result in major blindspots in practice. As well as experts' cultural blindspots, social dynamics, changing norms, and new circumstances can hinder experts' detection of many types of algorithmic biases and harms [22, 35, 45, 69, 71, 81]. In contrast, recent years have seen many cases in which users uncover and raise awareness about harmful algorithmic behaviors that they encounter while interacting with .

Everyday Algorithm Auditing: Understanding the Power of Everyday Users in Surfacing Harmful Algorithmic Behaviors

HONG SHEN* and ALICIA DEVOS*, Carnegie Mellon University, USA MOTAHHARE ESLAMI[†] and KENNETH HOLSTEIN[†], Carnegie Mellon University, USA

A growing body of literature has proposed formal approaches to audit algorithmic systems for biased and harmful behaviors. While formal auditing approaches have been greatly impactful, they often suffer major blindspots, with critical issues surfacing only in the context of everyday use once systems are deployed. Recent years have seen many cases in which *everyday users* of algorithmic systems detect and raise awareness about harmful behaviors that they encounter in the course of their everyday interactions with these systems. However, to date little academic attention has been granted to these bottom-up, user-driven auditing processes. In this paper, we propose and explore the concept of *everyday algorithm auditing*, a process in which users detect, understand, and interrogate problematic machine behaviors via their day-to-day interactions with algorithmic systems. We argue that everyday users are powerful in surfacing problematic machine behaviors that may elude detection via more centrally-organized forms of auditing, regarding audivelde a behaviors. Finally, we discuss work that lies ahead, toward bridging the gaps between formal auditing approaches and the organic auditing behaviors that a energe in everyday use of algorithmic systems.

 $\label{eq:CCS} Concepts: \bullet \textbf{Human-centered computing} \rightarrow \textbf{Human computer interaction (HCI)}; \textit{Empirical studies in HCI}.$

Additional Key Words and Phrases: Everyday Algorithm Auditing; Auditing Algorithms; Algorithmic Bias; Everyday Users; Fair Machine Learning

ACM Reference Format:

Hong Shen, Alicia DeVos, Motahhare Eslami, and Kenneth Holstein. 2021. Everyday Algorithm Auditing: Understanding the Power of Everyday Users in Surfacing Harmful Algorithmic Behaviors. Proc. ACM Hum.-Comput. Interact. 5, CSCW2, Article 433 (October 2021), 29 pages. https://doi.org/10.1145/3479577

Expert and user-involved audits



Encarnación et al., under review

Everyday emergent audits



9:28 AM · Sep 21, 2020 · Twitter Web App



Youth's perspectives towards algorithmic justice

- Research on learning and algorithmic justice has focused on rights and protections (Ito et al., 2023) and high-stake issues such as police surveillance (Vakil & McKinney de Royston, 2022)
- Youth's ideas about algorithmic justice are grounded in their lived experiences with technologies (Coenraad, 2022; Salac et al., 2023; Solyst et al., 2023)

Testing in AI/ML Education

- Some studies mention testing but provide little to no details about how students test models and what they learn from testing activities.
- The studies that address testing show interesting findings: testing helps learners build hypotheses about model performance



Morales-Navarro, L., Shah, M., & Kafai, Y. B. (2024, March). Not Just Training, Also Testing: High School Youths' Perspective-Taking through Peer Testing Machine Learning-Powered Applications. In *Proceedings of the 55th ACM Technical Symposium on Computer Science Education V. 1* (pp. 881-887).

Auditing ≠ Testing

- Auditing emphasizes the system rather than user interaction/reaction
- Auditing is **systematic and iterative** process with the goal of drawing conclusions at the level of the system rather than about individual test cases
- Audits are generally **external evaluations** done by independent third parties from the outside-in

Metaxa et al., 2021

Peer Auditing



TITLE: OBJECTIVE: LABELS/CLASSES:





2. Peer auditors receive project with factsheet and audit it. Every five minutes a new group of auditors evaluate the system by comparing expected outcomes to system outcomes.



3. Peer auditors read through all evaluation instances and write an auditing report.

PEER AUDITING WORKSHOP

Youth as Peer Auditors: Engaging Teenagers with Algorithm Auditing of Machine Learning Applications

LUIS MORALES-NAVARRO, University of Pennsylvania, United States YASMIN B. KAFAI, University of Pennsylvania, United States VEDYA KONDA, University of Pennsylvania, United States DANAÉ METAXA, University of Pennsylvania, United States

As artificial intelligence/machine learning (Al/AL) applications become more pervasive in youth lives, supporting them to interact, design, and evaluate applications is crucial. This paper positions youth as auditors of their peers' ML-powered applications to better understand algorithmic systems' opaque inner workings and external impacts. In a two-week workshop, 13 youth (ages 14-15) designed and audited ML-powered applications. We analyzed pre/post clinical interviews in which youth were presented with auditing tasks. The analyses show that after the workshop all youth identified algorithmic biases and inferred dataset and model design issues. Youth also discussed algorithmic justice issues and ML model improvements. Furthermore, youth reflected that auditing provided them new perspectives on model functionality and ideas to improve their own models. This work contributes (1) a conceptualization of algorithm auditing for youth; and (2) empirical evidence of the potential benefits of auditing. We discuss potential uses of algorithm auditing in learning and child-computer interaction research.

 $CCS Concepts: \bullet Human-centered computing \rightarrow Empirical studies in HCL \bullet Social and professional topics \rightarrow K-12 education; Computing literacy.$

Additional Key Words and Phrases: youth, algorithm auditing, algorithmic justice, machine learning, child-computer interaction, artificial intelligence

ACM Reference Format:

Luis Moralee-Navarro, Yasmin B. Kafai, Vedya Konda, and Danaë Metaxa. 2024. Youth as Peer Auditors: Engaging Teenagers with Algorithm Auditing of Machine Learning Applications. In Interaction Design and Children (IDC '24), June 17–20, 2024, Delf, Netherlands. ACM, New York, NY, USA, 22 pages. https://doi.org/10.1145/3028516.3653752 Luis Morales-Navarro, Yasmin B. Kafai, Vedya Konda, and Danaë Metaxa. 2024. Youth as Peer Auditors: Engaging Teenagers with Algorithm Auditing of Machine Learning Applications. In Interaction Design and Children (IDC '24), June 17–20, 2024, Delft, Netherlands. ACM, New York, NY, USA, 22 pages



Context of the study

W1: ML classifiers, physical computing, ML pipeline, Auditing session.

W2: Working on ML-powered electronic textile (e-textile) physical computing projects. Auditing session.

Pseudonym	Age	Gender	Race & Ethnicity	Previous CS experience
Kayla	14	Female	Black	Yes
Lou	15	Female	Black	No
Jerome	15	Male	Native American & Black	Yes
Bryan	15	Male	Asian & White	Yes
Jackie Star	15	Female	White	Yes
Fatimah	14	Female	Black	Yes
Andrés	14	Male	Latinx	Yes
Richard	14	Male	White	Yes
Iván	14	Male	Latinx & White	No
Emily	14	Female	Black	Yes
Luke	15	Male	Black & Latinx	Yes
Stephanie	15	Female	Black & White	Yes
Walter	15	Male	Asian	Yes

Youth projects



Interview protocol

Classifier tasks





How did youth's identification of potential algorithmic biases and harm change?

All participants identified potential algorithmic biases in post (compared to 9 in pre). Biases related to body shapes, breed (in the case of animals), color, size, shape, and context/location, race, and relevancy.



How did youth's identification of potential algorithmic biases and harm change?

In post they reflected on personal and societal biases.

"from **my personal experience**, teaching as a very female-dominated profession." Iván

"A lot of YouTube channels it has... I feel like it's mainly run by White guy gamers." Iván

How did youth's identification of potential algorithmic biases and harm change?

Considering harm and justice (7 in pre to 12 in post).

Diverse ideas about harm and justice.



"For the scientists, like kids saying they want to be scientists, **if nobody that looks like me is a** scientist, then should I really become one?" Luke



"Lizzo, everyone calls her beautiful but none of the generated pictures looked like her. <mark>If you look like her</mark> it can make you feel bad" Andrés



"Yeah, **it just excludes**. Like in this **context**, with just generating pictures. I don't know if it's really impactful, it could in other contexts." Jackie Star



"I don't think it can be harmful. I do think it's discriminatory. You're not gonna, like, get offended by the AI." Richard

How did youth's inferences about data and model design change?

Youth made more inferences to data and model design issues in post from 6.9 to 12.8 average inferences per participant.

Increased in post:

- Model features
- Data composition
- Data diversity
- Data context
- Data sources
- Class balance

Decreased in post:

• Data quantity

How did youth's inferences about data and model design change?



"provide more features to the model so that it would know what to look for" Fatimah



"yeah, definitely a bias towards sharks if it was close up to a face, because that's probably all that it really is like taught on" Jackie Star What benefits did youth find in auditing applications and having their applications audited?

Auditing provided with new perspectives about model functionality and performance.



"not just getting more diverse user input, but feedback from people that don't think like you." Iván



"you also get different standpoints because people think in so many different ways that, like, you wouldn't have thought of something and now you can incorporate that." Lou What benefits did youth find in auditing applications and having their applications audited?

Auditors provided helpful feedback



"people were like, well, you could have added more variety to this class" Jackie Star



"helped me humble myself, helped me realize, okay, there are changes I can make, or actually my project is doing much better than I thought it would" Fatimah What benefits did youth find in auditing applications and having their applications audited?

Looking at projects from new perspectives



"you can turn around and improve that yourself" lván



"I use the logic that I use in their project of challenging it to see what would break it on our project." Jerome

DISCUSSION

ALGORITHM AUDITING

- AS A SOCIO-TECHNICAL PROCESS
- FOR ALGORITHMIC JUSTICE
- AND COMPUTATIONAL EMPOWERMENT

AUDITING AS SOCIO-TECHNICAL PROCESS

- Youth benefited from cognitive distance and being able to "take perspective" of their own applications and those of their peers. This enabled them to provide recommendations for their peers and to apply what they saw as auditors to their own projects.
- Youth took a **more adversarial approach**, describing how, for some of them, the goal was to try to "break" the applications or find "all the problems". This approach differs from the stance of expert auditors

AUDITING FOR ALGORITHMIC JUSTICE

PRE **SOME** youth were able to identify potential biases.

Previous research shows that both adults and teenagers participating in cooperative inquiry sessions and think-aloud interviews can engage with these topics by building on their rich experiences as users of AI/ML-powered applications [13, 49, 56].

POST ALL youth identified potential biases.

AUDITING AS COMPUTATIONAL EMPOWERMENT

"We define computational empowerment as a concern for the method used by students, as individuals and groups, to develop the capacity to understand digital technology and its effect on their lives and society at large and their ability to engage critically and curiously with the **construction** and **deconstruction** of technology." Dindler, Iverson Caspersen & Smith, 2022, p. 121

AUDITING FOCUSES ON **DECONSTRUCTION** OF TECHNOLOGY:

Supporting youth in analyzing applications designed by other people to **interrogate the values encoded in them and their impact on society** (Schaper et al., 2022).

ALGORITHM AUDITING as DECODING

Deconstruction involves describing, evaluating and reflecting on the values and intentions embedded in sociotechnical systems and considering their possible implications (Dindler et al., 2020; Schaper et al., 2022). This involves

(1) Assessing the properties of a system in terms of its inputs, outputs and materials,

(2) Inquiring on the intended use and actual use of a system,

(3) Foregrounding the values, worldviews, and assumptions embedded in the system, and

(4) Impacting individuals, communities and the environment.

Algorithm auditing is a method that can support learners in deconstructing AI/ML applications.

NEXT STEPS

- **DESIGNING AND AUDITING ML APPLICATIONS**
- INTRODUCING TEACHERS TO AUDITING ML APPLICATIONS
- INTEGRATING AUDITING ACTIVITIES IN CS CLASSROOMS



Our team:

Yasmin

Danaë

Lauren

Luis Vedya Phillip

Thank you!

