The Promise and Peril of Algorithm Audits for Increasing Transparency and Accountability of Donated Datasets

Christo Wilson
Khoury College of Computer Sciences, Northeastern University
cbw@ccs.neu.edu

2019

INTRODUCTION

Online platforms contain some of the largest and most valuable datasets ever assembled in human history. Extensive records of what people say, what they read, what they like and dislike, where they go and when, etc. are all routinely collected and stored by a variety of private companies. Opening up access to these datasets so that researchers, journalists, and policy makers can leverage them for science and for social good is one of the great challenges of this century.

Unfortunately, the barriers around these datasets keep getting higher. For example, in 2018 Facebook and Twitter both changed their APIs to make it more difficult to gather data from their respective platforms (Schroepfer 2018; Roth and Johnson 2018). On one hand, these changes were an understandable reaction to the public outcry following well-publicized privacy breaches like Cambridge Analytica (Rosenberg, Confessore, and Cadwalladr 2018). On the other hand, these changes were heavy-handed, in that they do not distinguish between researchers acting in good faith and malicious parties.

One viable path forward is partnership between researchers and companies. For example, Social Science One is pioneering a model where researchers apply for access to massive-scale data from Facebook (Social Science One 2018). This structure ensures that only qualified researchers with peer-reviewed research plans are granted access to sensitive data, thus helping to mitigate concerns that control of the data will be lost, or that it will be misused. A data sharing organization with broad buy-in from companies and researchers could, in theory, scale this model up to tackle larger data sets and more concurrent research projects.

However, even if companies can be convinced to donate datasets to a centralized organization, challenges remain with respect to this data. *First*, donated data is often stripped of the context in which it was produced, making it difficult to interpret. This is especially true when the data is question is collected by or generated from a socio-technical system that combines various kinds of algorithms with input from human beings. Algorithmic systems that rank, sort, recommend, filter, and otherwise

permute data influence the behavior of users, thus creating artifacts in the data. Unless these algorithms are understood, it can be difficult to separate the "signal" in the data corresponding to human behavior, from the "noise" created by algorithmic interfaces. *Second*, there are fundamental questions of trust with respect to donated data. How can researchers be sure that data provided by private entities is complete and representative when these datasets are locked up inside opaque systems that hinder attempts at independent validation?

Algorithm auditing may offer solutions to these challenges. Algorithm auditing is an emerging field that focuses on increasing the *transparency* and *accountability* of opaque, "black-box" algorithmic systems. Algorithm audits are the modern incarnation of classic methods pioneered by social scientists and economists for investigating real-world discrimination. For example, in a "correspondence study" the auditor mails fake resumes to a company that are identical except for carefully chosen attributes (e.g., race and gender) to measure who gets invited for interviews. This may reveal discrimination in the company's hiring process. Algorithm audits operate in the same spirit: by carefully supplying input to an algorithm of interest, the outputs can be collected and analyzed to determine how the algorithm functions, whether it exhibits specific behaviors, and ultimately what impact this algorithm may have on human users.

In this paper, I discuss the opportunities and challenges of integrating algorithm auditing into a data sharing organization. I begin by briefly presenting background information on algorithm auditing as a field, examples of successful audits, and methods that are commonly leveraged to conduct audits. Next, I discuss how algorithm audits could be used to enrich donated datasets by (1) analyzing the algorithmic contexts that may have influenced the collection and production of the data, and (2) enabling independent validation that the data is complete and representative. Finally, I discuss several key challenges, including: tradeoffs between cooperating with and adversarially auditing data partners; maintaining the privacy of data collected during audits; and legal risks associated with auditing.

BACKGROUND

Computer scientists have been probing the internal workings of opaque, corporate systems for many years, although it was not until 2014 that the term "algorithm auditing" was applied to this work (Diakopoulos 2014; Sandvig et al. 2014). The goals of algorithm auditing are to increase *transparency* and *accountability* around so-called "black-box" algorithmic systems by answering basic questions about how these systems operate. Is there an algorithm that is shaping users' experience in a given context? What variables or features are used as input by this algorithm, and how are they related to the algorithm's outputs? What are the privacy implications of the data being used by an algorithm in a particular context? Are the outputs of the algorithm biased or discriminatory against specific (possibly legally protected) classes of people? Does the algorithm ever produce repugnant output (e.g., labeling images of dark-skinned people as gorillas (Dougherty 2015))? By critically examining consequential systems that are operating in the public-sphere, algorithm audits enable to people to have informed discussions about how these systems are implemented, their social consequences, and possibly put pressure on operators and designers to change undesirable or unacceptable behaviors.

Like social science audits and correspondence studies, algorithm audits are temporally fixed. The goal of an audit is to determine the existence of an algorithm and analyze its behavior at a single point in time. An algorithm audit does not necessarily tell us about the behavior of a target algorithm in the past or future (unless the audit is done longitudinally), nor does it necessarily tell us about the behavior of other (possibly related) algorithms.

Similar again to the classic studies, algorithm audits are outcome-oriented. Algorithm auditors eschew the term "reverse-engineering" since it implies that the goal of an audit is to derive a complete understanding of a black-box system, possibly down to the source code-level. This is an explicit non-goal. Some audit designs may be able to answer the question "why did an algorithm behave a specific way," but this is not true in all cases, and is not a fundamental requirement for a successful audit. Instead, algorithm audits are primarily focused on the outputs and outcomes of systems, i.e., did an algorithm exhibit problematic behavior at the time of the audit?

The outcome-orientation of algorithm audits completely obviates the criticism that systems may be "too complicated" to audit, or that machine learning-based systems are "uninterpretable". Just as a correspondence study does not attempt to answer the question of why a person or organization is discriminatory, algorithm audits can be explicitly designed to avoid disentangling why an algorithm exhibits specific behaviors; it is often sufficient to simply demonstrate that a behavior existed at some point in time.

Finally, just like the classic methods, algorithm audits are designed as deception studies. The service that is being audited is not informed ahead of time, as this might cause them to (temporarily) alter their platforms behavior. Instead, we follow the principle of *responsible disclosure*, and debrief the targets of our audits after the studies are complete, but before the results are publicized.

Examples. Algorithm auditing techniques have been successfully applied to systems in many domains, with online targeted advertising perhaps being the first. In a seminal study, Latanya Sweeney found racial discrimination in ads served by Google's AdSense (Sweeney 2013). Other studies have found that Facebook and Google's platforms target users based on sensitive attributes (e.g., gender, sexual orientation) without their knowledge (Guha, Cheng, and Francis 2010; Datta, Tschantz, and Datta 2015). Auditors are now critically examining the tools provided by advertising platforms to determine the impact on user privacy (Venkatadri et al. 2018).

Another active area is audits of algorithms on e-commerce platforms. Investigative journalists and academics have documented cases of online price discrimination (Valentino-DeVries, Singer-Vine, and Soltani 2012; Mikians et al. 2012, 2013; Hannak et al. 2014). Chen et al. investigated dynamic pricing algorithms and the "Buy Box" on Amazon (Chen, Mislove, and Wilson 2016), while Eslami et al. investigated how user's reacted to a skewed feedback system on Booking.com (Eslami et al. 2017). Two studies have critically examined ridesharing systems, focusing on their implementation of dynamic "surge" pricing (Chen, Mislove, and Wilson 2015), as well as the equitability of these services (Jiang et al. 2018).

A third area that has received attention is search engines, typically motivated by fears of partisan "echo chambers" and "filter bubbles" (Pariser 2011). Audits have extensively documented how Google Search implements personalization (Hannak et al. 2013; Robertson, Lazer, and Wilson 2018) and localization (Kliman-Silver et al. 2015), as well as the impact of these design decisions on the partisan-lean of political information that is surfaced by the search engine (Robertson et al. 2018). Auditors have begun branching out to other impactful search interfaces, such as Twitter (Kulshrestha et al. 2017).

Lastly, there is a growing literature around audits that examine gender and racial discrimination by algorithms. Hannak et al. examined bias in the TaskRabbit and Fiverr gig-economy marketplaces (Hannák et al. 2017), while Chen et al. examined the relationships between gender and search rank on the recruitment websites Monster, CareerBuilder, and Indeed (Chen et al. 2018). Angwin et al. used public records requests to investigate the COMPAS recedivism prediction tool, and found that it discriminated against African Americans by erroneously classifying them as "high risk" more often than

Whites (Angwin et al. 2016).

ALGORITHM AUDITING METHODS

In this section, I briefly introduce methods for auditing algorithms. This includes "adversarial" audits, that are conducted at arms length from a target system, as well as tools that are meant to be used by developers to audit their own algorithms.

Sandvig et al. propose five methods for conducting audits of black-box algorithms that are designed to address specific technical and legal challenges (Sandvig et al. 2014):

1. In a **Code Audit**, researchers directly examine the source code of a system. While conceptually simple, this type of audit is unlikely to succeed in practice. Companies are typically reluctant to disclose source code that may contain trade secrets. Furthermore, source code alone is insufficient to understand algorithms that leverage machine learning; in these cases the underlying data is also essential.

2. In a **Noninvasive User Audit**, researchers recruit participants to gather samples of input and output data from a given algorithm. This method has the advantage of placing minimal burden on the target service (since the participants typically use the system anyway). However, the use of real people introduces confounding variables into the input data that makes it difficult to derive detailed conclusions about the behavior of the algorithm.

3. In a **Scraping Audit**, the researchers simply download information from the target service, either directly from webpages or via an API. No attempt is made to masquerade as a legitimate user. This method is not viable in situations where user characteristics impact the output of the algorithm.

4. In a **Sock Puppet Audit**, researchers use fake accounts that they control to probe an algorithm. This method is most similar to a traditional audit study, and has the advantage of giving researchers the greatest level of control over experimental variables.

5. A **Crowdsourced Audit** is the same as a sock puppet audit, except that the user accounts are created and controlled by real people (crowdworkers) in the employ of the researchers.

In prior work, my collaborators and I developed a *hybrid* methodology for auditing algorithms that combines **noninvasive user** and **sock puppet audits** (Hannak et al. 2013; Hannak et al. 2014). At a high-level, we seek to answer two questions about a target algorithm: *how does the algorithm behave under real-world conditions?*, and *what specific features influence the algorithm's output*? The user audit allows us to answer the first question by observing the algorithm's behavior as it interacts with a large number of real people. The sock puppets allow us to answer the second question, since we can precisely control all of the features of the puppets and observe how the algorithm reacts to each one in isolation.

Tools. Sandvig et al.'s methods are designed for adversarial algorithm audits, i.e., the audit is conducted by an outsider with no privileged access to the target system. However, as concerns about fairness and interpretability of machine learning algorithms have grown, there has been a push in the research community to build tools that developers can use to audit their own algorithms. Examples tools include FairML (Adebayo 2016), Lime (Ribeiro, Singh, and Guestrin 2016), Aequitas (Saleiro

et al. 2018), and FairTest (Tramer et al. 2017). In some cases, these tools are meant to help developers identify biases in datasets before they are used in applications or to train models. In other cases, these tools "probe" trained models to determine if the they exhibit patterns of classification or prediction errors that are biased, e.g., the distribution of errors is strongly correlated with membership in a legally protected class.

While these "white-box" auditing tools are useful for developers, they have limited utility for "black-box" adversarial audits. Since these tools assume complete access to datasets and trained models, they are typically only useful for insiders. In contrast, external auditors rarely have access to raw, internal datasets, and their ability to probe machine learning models are restricted by the interface offered by the service (e.g., via the functionality offered by a corresponding website or app).

AUDITING FOR TRANSPARENCY

Data is not generated in a vacuum: the design of socio-technical systems has an enormous impact on the shape of data collected by these systems. User interface and platform design can incentivize, or disincentive, peoples' behavior, leading to various artifacts in data itself. For example, interfaces that present information in rank-ordered lists privilege the content at the top (Granka, Joachims, and Gay 2004; Guan and Cutrell 2007; Lorigo et al. 2008); this can lead to rich-get-richer effects where data that is not objectively "better" gains the appearance of being so due to a feedback loop between human attention, clicks, and algorithmic sorting based on engagement (Salganik, Dodds, and Watts 2006). In another complex example, we found that users exhibited racial and gender discrimination in social feedback on the gig-economy marketplace TaskRabbit (Hannák et al. 2017); this biased feedback then impacted the ordering of workers in search results, which could cause white, male workers to receive more jobs in the future, thus skewing the employment data in the system.

Artifacts in datasets that are caused by the design of platforms are particularly pernicious because these designs are typically opaque. With respect to a given dataset, it may be unclear what, if any, algorithmic systems shape the user experience and thus the data generated by users, let alone how these designs impact the data quantitatively. Even in cases where data is freely released by an organization, this additional context about the manner in which the data was originally generated and collected is often missing.

Algorithm auditing is one potential tool for attempting to address these concerns about the lack of transparency around socio-technical systems and their associated data. Algorithm audits offer the possibility of determining what algorithms, if any, are operative behind-the-scenes in a system that collects or produces a dataset. In some cases, it may even be possible to critically interrogate these algorithms to quantify their effect, e.g., to determine how information is ranked, filtered, or otherwise permuted before being presented to users.

Case study. Twitter vividly illustrates the necessity of audits to increase transparency of algorithmic systems. There is an enormous amount of literature that is based on data from Twitter, analyzing the social graph, the ebb and flow of trends, the content of tweets, the links shared by users, etc. However, this literature almost entirely fails to account for the various algorithmic systems that shape the user experience on Twitter, and consequently shape the resulting data. Prominent examples include the "Who to follow" recommendation system; the "In case you missed it" timeline ranking system; "For you", "Moments", and Twitter's other various content-recommendation systems; and tweet search. All of these algorithms alter the flow of information received by users, privileging some at the expense of

others. It is totally unclear how these algorithms shape the social structure of Twitter, which topics and hashtags trend, etc. I argue that all of these separate algorithms should be audited, given their prominence on the platform, and given the critical importance that Twitter (and data from Twitter) has taken on in the public sphere.

Despite all of the active algorithms on Twitter, it is still an "easy case" with respect to the impact of algorithms on datasets. Twitter is known for their relatively unfiltered user-interface, and attempts to meddle with the status quo have been met with swift backlash from users (Statt 2018). Contrast this with more complex systems like Facebook, which is an agglomeration of separate products that all get forced through the single, highly-curated bottleneck of the news-feed. If Facebook data becomes more widely available within the research community (Social Science One 2018) the impact of these various algorithms on the resulting data will need to be assessed and grappled with.

AUDITING FOR ACCOUNTABILITY

One serious consequence of data being centralized inside major companies is that the public is totally beholden to these entities for information. When data is released by a company, it is not immediately apparent that the data is complete, representative, or error free, since it is difficult to independently verify the data's quality. Ultimately, I frame this issue as one of *accountability*: how can the research community ensure that datasets provided by closed entities are given in good faith, and are not riddled with undetectable errors?

Malice need not be the cause of fidelity problems with released data: there are any number of non-malicious reasons why a dataset might have systematic problems. For example, suppose that a company releases a dataset that is sampled, supposedly uniformly at random. Unfortunately, we know that truly random sampling is hard to achieve in practice on very large datasets (Morstatter et al. 2013), which may impact the representativeness of the released data. Alternatively, consider a system that uses personalization to tailor content. A snapshot of data from this system may fail to capture what users actually experience in practice (if the data is non-personalized), or it may capture a view of the system that only corresponds to a subset of the overall userbase (if a skewed sample of personalized data is captured).

Another example worth highlighting is the complex interaction between content moderation and data preservation. Suppose a social media company releases a purportedly complete sample of messages from their platform over some time period. We might reasonable ask: is this data truly complete? Does it include messages that were automatically deleted by spam-prevention or security systems? What about messages that were taken down due to copyright complaints? What about messages that were moderated after eliciting complaints from other users (e.g., for including hate speech or pornography)? For some research uses, the answers to these questions may not matter. But for others, these issues may be consequential, and the company in question may be unwilling or unable to answer these questions.

Algorithm auditing may offer solutions to these problems. Using algorithm auditing techniques, independent experts can critically interrogate a platform by creating and crawling content. These tools allow the auditor to collect their own data samples, which can be compared to the official samples to assess whether they appear to be complete and representative. Further, by injecting new, carefully controlled content into a system, an auditor can test mechanisms and policies that govern data on the platform, such as content moderation, filtering, and takedown systems.

Ridesharing Example. Our prior work offers an illustrative example. We set out to study the equitability of the ridesharing service Uber by collecting vehicle data from its smartphone app. By default, the app claims to show the real-time locations of the eight closest available vehicles to the user's current GPS coordinates. By leveraging many user accounts, each claiming to be at a different location, we were able to "blanket" major cities and collect data on all vehicles that could be observed in the Uber app (Chen, Mislove, and Wilson 2015; Jiang et al. 2018).

However, we were concerned that our data was not representative. What if the Uber app was not presenting data on real vehicles? To assess this, we leveraged a dataset of Uber trips that was obtained by the city of New York directly from the company, and was subsequently publicly released. We compared the geospatial and temporal statistics of Uber vehicles we observed in New York City using our data collection method to the vehicles in the data provided by Uber. We found that both sets of vehicles exhibited statistically identical geospatial and temporal distributions, which provided strong evidence that the data we collected from the Uber app was valid.

Now consider the reverse situation. What if a ridesharing company publicly released a dataset of rides, and the scientific community wanted to assess if the data was truly representative? We could use auditing techniques to surreptitiously collect data from the ridesharing app and use statistical tests to verify that the released dataset was drawn from the same distributions as the crawled data. This general approach could be applied in any situation where an auditor had the ability to independently collect data from the target system.

CHALLENGES

There are clear benefits towards transparency and accountability that arise from including a robust algorithmic auditing program alongside an effort to collect and share large datasets. However, algorithm auditing also raises many thorny challenges that need to be carefully considered and grappled with before any community fully commits to integrating these techniques into their repertoire.

Cooperation Versus Adversity

Since the first audits and correspondence studies in social science and economics, the field has relied on deception as an essential facet of study designs. My own algorithm auditing work also relies on deception: we responsibly disclose our methods and findings to subjects (typically companies) only after the audit is complete. These disclosures have been met with a variety of reactions, ranging from total indifference, to mild annoyance, to open hostility.

The necessity for deception creates a fundamental tension between algorithm auditing and data sharing as a cooperative endeavor. On one hand, a truly effective organization dedicated to collecting and sharing large datasets must be able to cooperate with many organization, e.g., government agencies and companies. On the other hand, auditing those same data partners may hinder, or even entirely prevent, this cooperation. Indeed, companies may be incentivized to share datasets specifically because it allows them to manufacture good will while simultaneously carefully controlling the dissemination of curated data without having to reveal problematic data, secretive practices, or proprietary algorithms. Algorithm auditing may obviate some of these benefits for data partners, by uncovering inconsistencies in data or hidden details about underlying algorithmic processes.

One potential mitigation for this issue is to offer data partners a range of transparency options with respect to the data they share. Partners that are "fully" transparent might need to share data as well as details about any algorithmic systems that impact the collection, presentation, or permutation of

this data (e.g., any algorithm that might induce artifacts in the data or impact its completeness). The details about algorithms could be held in confidence, i.e., not released publicly. Alternatively, partners opting for less transparency might need to agree a priori to an audit.

Ultimately, however, the tension between cooperation and adversity boils down to trust, and there is no easy way to resolve this. If the goal of the data sharing organization is to maximize partnership and cooperation, algorithm auditing may be incompatible with this mission. Alternatively, if the organization adopts a more cautious (or cynical) "trust, but verify" world-view, then the costs of algorithm auditing may be more acceptable versus the possibility of reduced partnership.

Secrecy Versus Transparency

A second fundamental contradiction with respect to algorithm audits is that their goal is to increase transparency, but the audit itself (and the data collected during the audit) may be less than transparent, depending on the methods that are used. This may stand in stark contrast to the overall objective of a data sharing organization, which may strive for complete openness for all datasets. Can an organization dedicated to sharing data reconcile the need for secrecy when it comes to the data generated during algorithm audits?

The need for data secrecy arises from privacy issues that are inextricable from some algorithm auditing methods. Some algorithm audit designs rely solely on publicly available data, e.g., from crawls or from sock puppets accounts that are created by the auditor. In these designs, there is often little-to-no risk of exposing private data that belongs to other, human users of the target system.[1] Formally, these design should be exempt from institutional review requirements concerning human subjects (although we encourage all auditors to seek out formal exemptions regardless).

However, audit designs that rely on real people to help collect data, such as noninvasive user and crowdsourced audits (Sandvig et al. 2014), very much run the risk of collecting private data from participants. For example, in our audits of Google Search we have relied on a browser extension installed by real people to run search queries on our behalf (Robertson, Lazer, and Wilson 2018; Robertson et al. 2018). Even though the queries were chosen by us (i.e., we did not collect participants' personal queries), the search engine result pages may include personal data like the participant's real name and Google Account username, if they were logged-in to Google at the time of our data collection. Entirely expunging this personal data from the collected HTML is impossible, and thus we cannot release this raw dataset (these restrictions are further laid out in the IRB approval for our study).

On one hand, issues related to participant privacy are completely expected in many scientific areas. Survey data, for example, is almost always anonymized at collection time and aggregated before release to preserve individual privacy. On the other hand, it is important to recognize that even if precautions are taken, there is often no foolproof method for anonymizing data from algorithm audits, as the Google Search example above illustrates. The best one can hope to achieve in these cases is release of highly aggregated data, which may not satisfy desires or mandates for full transparency and reproducibility.

[1] There are exceptions to this rule. For example, as of 2016 Lyft assigned unique, unchanging identifiers to all drivers, which we were able to collect during our audit of their platform (Jiang et al. 2018). We did not publicly release this data since it might violate drivers' expectations of privacy.

Legal Risks

As of the writing of this document, algorithm auditing exists in a legal gray area. Many algorithm auditing designs require collecting data from services using methods that violate these services' Terms of Use or Terms of Service (ToS). For example, many services prohibit the crawling of data, the analysis of crawled data, and/or the creation of sock puppet accounts. Within the US, circuit courts are split with regards to whether breaches of ToS are violations of the Computer Fraud and Abuse Act (CFAA), which is the primary federal anti-hacking law in the US. To a lesser extent, algorithm audits may also run afoul of the Digital Millennium Copyright Act (DMCA) if the audit "circumvents" security systems, as well as trade secrecy laws if proprietary information or algorithms are revealed during the course of an audit.

Although there are efforts underway aimed at reforming or narrowing the CFAA (such as a legal case that I am part of whose goal is to legitimize good-faith algorithm auditing (Mislove and Wilson 2017)), the timeline and outcome of these efforts remain unclear. Additionally, audit methods that are less risky from a legal standpoint, such as crowdsourced designs (Sandvig et al. 2014), are not entirely safe. For example, even though crowdsourcing may obviate the need for an auditor to make sock puppet accounts, some ToS stipulate that users may not share their accounts with third-parties. Thus, enlisting users to aid in crowdsourced audits may simply shift the legal risk from the auditor to the confederate users.

Any organization or individual that plans to engage in algorithm audits needs to seriously contemplate these legal risks. Is increased transparency and accountability worth the potential downsides? Is the organization or individual sufficiently well-resourced to defend themselves in the event that the target of an audit decides to pursue civil damages, or if a government decides to pursue a criminal case?

Safe Harbors for Audits. In the short-term, the cybersecurity community may offer lessons for how to navigate the legal issues surrounding algorithm audits. Historically, white-hat hackers (i.e., independent cybersecurity experts operating in good-faith) were often targeted with legal threats and intimidation after they revealed vulnerabilities in products and systems. White-hats were caught in a catch-22 situation: (1) reveal vulnerabilities privately to companies, and run the risk of being ignored (i.e., the vulnerability would never be addressed), or (2) go public to force the company to mitigate, and consequently run the risk of prosecution.

Over time, some companies have come to realize that white-hat hackers are an asset, not a threat, and moved to create structures that enfranchise them. These often take the form of *bug bounty programs* where companies can delineate exactly what systems they will allow white-hats to scrutinize, the methods that are permitted for scrutinizing these systems, and the process for disclosing problems (e.g., privately at first, followed by full public disclosure after the issue has been mitigated). As the name suggests, companies pay bounties to white-hats that participate in these programs to incentivize their work and encourage them to responsibly disclose vulnerabilities.

Crucially, bug bounty programs are beginning to include contractual language that exempts white-hat hackers from legal risk (e.g., from the CFAA and DMCA in the US) as long as they operate within the boundaries of the program. Amit Elizari, a legal scholar from Berkeley, has been instrumental in drafting the legal language of these *safe harbors* and promoting their adoption by major bug bounty platforms (Elazari Bar On 2018a).

Amit Elizari has proposed that safe harbors could be developed and adopted for algorithm audits as well (Elazari Bar On 2018b). This would have the effect of enfranchising good-faith auditors, while

allowing platforms and services to establish some boundaries for audits (e.g., to minimize any disruption to systems that might result from audits). Following the same principle of *responsible disclosure* from cybersecurity, platform providers would have a chance to triage the results of audits before they were made public, thus helping to balance the interests of platforms and auditors.

An organization dedicated to data sharing could be invaluable for promoting the adoption of safe harbors for algorithm audits. At one extreme, the organization could require that data partners adopt such a safe harbor framework, as a show of good-faith and to preempt legal risk if the organization chose to audit the partner's algorithms. A less extreme approach might be to encourage partners to adopt safe harbors, and perhaps offer incentives for doing so, such as greater access to datasets and researchers.

CONCLUSION

It is critical that structures be developed for data sharing between private entities and researchers, journalists, and policy makers. As it stands, almost all value from massive datasets is being captured by private entities, to the great detriment of science and society.

Algorithm audits are, potentially, a valuable tool that can be coupled with broader efforts to gather and share massive datasets. First, audits can help increase transparency by uncovering the missing context around datasets that are shaped by the interplay between algorithms and people. Second, audits can improve accountability by enabling independent "double-checking" of datasets, to ensure that they are reliable, truthful, complete, and representative.

That said, the benefits of algorithm auditing must be carefully weighed against the risks. Audits are adversarial by design, which may deter data partners from cooperating with any organization that uses them. Further, the data collected during audits may contain private information from platform users, and/or participants in the audit, which may prevent the data from being released. Finally, algorithm auditing methods are subject to legal uncertainty that must be seriously grappled with.

References

Adebayo, Julius A. 2016. "FairML: Toolbox for Diagnosing Bias in Predictive Modeling." Master's thesis, Massachusetts Institute of Technology. https://github.com/adebayoj/fairml.

Angwin, Julia, Jeff Larson, Surya Mattu, and Lauren Kirchner. 2016. *Machine Bias.* Accessed November 28, 2018. https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing.

Chen, Le, Ruijun Ma, Anikó Hannák, and Christo Wilson. 2018. "Investigating the Impact of Gender on Rank in Resume Search Engines." In *Proc. of CHI.*

Chen, Le, Alan Mislove, and Christo Wilson. 2015. "Peeking Beneath the Hood of Uber." In *Proc. of IMC.*

———. 2016. "An Empirical Analysis of Algorithmic Pricing on Amazon Marketplace." In *Proc. of WWW.*

Datta, Amit, Michael Carl Tschantz, and Anupam Datta. 2015. "Automated Experiments on Ad Privacy Settings: A Tale of Opacity, Choice, and Discrimination." In *Proc. of PETS*.

Diakopoulos, Nicholas. 2014. *Algorithmic Accountability: On the Investigation of Black Boxes.* Accessed November 28, 2018. https://www.cjr.org/tow_center_reports/algorithmic_accountability_on_the_investigation_of_black_boxes.php.

Dougherty, Conor. 2015. "Google Photos Mistakenly Labels Black People 'Gorillas'." *The New York Times.* Accessed November 28, 2018. https://bits.blogs.nytimes.com/2015/07/01/google-photos-mistakenly-labels-black-people-gorillas/.

Elazari Bar On, Amit. 2018a. "Private Ordering Shaping Cybersecurity Policy: The Case of Bug Bounties." In *Rewired: Cybersecurity Governance,* edited by Ryan Ellis and Vivek Mohan, 231–264. Hoboken, NJ: Wiley.

———. 2018b. *We Need Bug Bounties for Bad Algorithms.* https://motherboard.vice.com/en_us/article/8xkyj3/we-need-bug-bounties-for-bad-algorithms.

Eslami, Motahhare, Kristen Vaccaro, Karrie Karahalios, and Kevin Hamilton. 2017. ""Be Careful; Things Can Be Worse than They Appear": Understanding Biased Algorithms and Users' Behavior around Them in Rating Platforms." In *Proc. of ICWSM.*

Granka, Laura A., Thorsten Joachims, and Geri Gay. 2004. "Eye-Tracking Analysis of User Behavior in WWW Search." In *Proc. of SIGIR.*

Guan, Zhiwei, and Edward Cutrell. 2007. "An Eye Tracking Study of the Effect of Target Rank on Web Search." In *Proc. of CHI.*

Guha, Saikat, Bin Cheng, and Paul Francis. 2010. "Challenges in Measuring Online Advertising Systems." In *Proc. of IMC.*

Hannak, Aniko, Piotr Sapiezynski, Arash Molavi Kakhki, Balachander Krishnamurthy, David Lazer, Alan Mislove, and Christo Wilson. 2013. "Measuring Personalization of Web Search." In *Proc. of WWW.*

Hannak, Aniko, Gary Soeller, David Lazer, Alan Mislove, and Christo Wilson. 2014. "Measuring Price Discrimination and Steering on E-Commerce Web Sites." In *Proc. of IMC.*

Hannák, Anikó, Claudia Wagner, David Garcia, Alan Mislove, Markus Strohmaier, and Christo Wilson. 2017. "Bias in Online Freelance Marketplaces: Evidence from TaskRabbit and Fiverr." In *Proc. of CSCW.*

Jiang, Shan, Le Chen, Alan Mislove, and Christo Wilson. 2018. "On Ridesharing Competition and Accessibility: Evidence from Uber, Lyft, and Taxi." In *Proc. of WWW.*

Kliman-Silver, Chloe, Anikó Hannák, David Lazer, Christo Wilson, and Alan Mislove. 2015. "Location, Location, Location: The Impact of Geolocation on Web Search Personalization." In *Proc. of IMC.*

Kulshrestha, Juhi, Motahhare Eslami, Johnnatan Messias, Muhammad Bilal Zafar, Saptarshi Ghosh, Krishna P Gummadi, and Karrie Karahalios. 2017. "Quantifying Search Bias: Investigating Sources of Bias for Political Searches in Social Media." In *Proc. of CSCW.*

Lorigo, Lori, Maya Haridasan, Hrönn Brynjarsdóttir, Ling Xia, Thorsten Joachims, Geri Gay, Laura Granka, Fabio Pellacini, and Bing Pan. 2008. "Eye Tracking and Online Search: Lessons Learned and Challenges Ahead." *Journal of the Association for Information Science and Technology* 59 (7): 1041–1052.

Mikians, Jakub, László Gyarmati, Vijay Erramilli, and Nikolaos Laoutaris. 2012. "Detecting Price and Search Discrimination on the Internet." In *Proc. of HotNets.*

———. 2013. "Crowd-Assisted Search for Price Discrimination in e-Commerce: First Results." In *Proc. of ACM Conference on Emerging Networking Experiments and Technologies.*

Mislove, Alan, and Christo Wilson. 2017. *We're Suing the Federal Government to Be Free to Do Our Research.* The Conversation. Https://theconversation.com/were-suing-the-federal-government-to-be-free-to-do-our-research-74676. https://theconversation.com/were-suing-the-federal-government-to-be-free-to-do-our-research-74676.

Morstatter, Fred, Jurgen Pfeffer, Huan Liu, and Kathleen M. Carley. 2013. "Is the Sample Good Enough? Comparing Data from Twitter's Streaming API with Twitter's Firehose." In *Proc. of ICWSM.*

Pariser, Eli. 2011. *The Filter Bubble: What the Internet Is Hiding from You.* New York: Penguin Press.

Ribeiro, Marco Tulio, Sameer Singh, and Carlos Guestrin. 2016. ""Why Should I Trust You?": Explaining the Predictions of Any Classifier." In *Proc. of KDD.*

Robertson, Ronald E., Shan Jiang, Kenneth Joseph, Lisa Friedland, David Lazer, and Christo Wilson. 2018. "Auditing Partisan Audience Bias within Google Search." *Proceedings of the ACM on Human-Computer Interaction* 2, CSCW:Article 148. doi:10.1145/3274417.

Robertson, Ronald E., David Lazer, and Christo Wilson. 2018. "Auditing the Personalization and Composition of Politically-Related Search Engine Results Pages." In *Proc. of WWW.*

Rosenberg, Matthew, Nicholas Confessore, and Carole Cadwalladr. 2018. "How Trump Consultants Exploited the Facebook Data of Millions." *The New York Times.* https://www.nytimes.com/2018/03/17/us/politics/cambridge-analytica-trump-campaign.html.

Roth, Yoel, and Rob Johnson. 2018. *New Developer Requirements to Protect Our Platform.* Twitter Developer Blog.

Saleiro, Pedro, Abby Stevens, Ari Anisfeld, and Rayid Ghani. 2018. "Aequitas: Bias and Fairness Audit." https://github.com/dssg/aequitas.

Salganik, Matthew, Peter Sheridan Dodds, and Duncan J. Watts. 2006. "Experimental Study of Inequality and Unpredictability in an Artificial Cultural Market." *Science* 311:854–856.

Sandvig, Christian, Kevin Hamilton, Karrie Karahalios, and Cedric Langbort. 2014. "Auditing Algorithms: Research Methods for Detecting Discrimination on Internet Platforms." *Data and Discrimination: Converting Critical Concerns into Productive Inquiry:* 1–23.

Schroepfer, Mike. 2018. "An Update on Our Plans to Restrict Data Access on Facebook." https://newsroom.fb.com/news/2018/04/restricting-data-access/.

Social Science One. 2018.

Statt, Nick. 2018. *Twitter Will Soon Let You Switch between Chronological and Ranked Feeds.* https://www.theverge.com/2018/9/17/17872276/twitter-algorithmic-timeline-settings-change-viral-tweet-response.

Sweeney, Latanya. 2013. "Discrimination in Online Ad Delivery." *ACM Queue* 11 (3).

Tramer, Florian, Vaggelis Atlidakis, Roxana Geambasu, Daniel Hsu, Jean-Pierre Hubaux, Mathias Humbert, Ari Juels, and Huang Lin. 2017. "FairTest: Discovering Unwarranted Associations in Data-Driven Applications." In *Proc. of IEEE European Symposium on Security and Privacy.*

Valentino-DeVries, Jennifer, Jeremy Singer-Vine, and Ashkan Soltani. 2012. "Websites Vary Prices, Deals Based on Users' Information." *The Wall Street Journal.* Accessed November 28, 2018. https://www.wsj.com/articles/SB10001424127887323777204578189391813881534.

Venkatadri, Giridhari, Yabing Liu, Athanasios Andreou, Oana Goga, Patrick Loiseau, Alan Mislove, and Krishna P. Gummadi. 2018. "Privacy Risks with Facebook's PII-Based Targeting: Auditing a Data Broker's Advertising Interface." In *Proc. of IEEE Symposium on Security and Privacy.*