2020

Accountability in Computer Systems

Kroll, Joshua A.

Oxford University Press

Kroll, Joshua A., Accountability in Computer Systems (June 30, 2020). The Oxford Handbook of the Ethics of AI. Dubber, Markus D., Frank Pasquale, and Sunit Das, Eds. Oxford University Press. 2020.

http://hdl.handle.net/10945/65368

Accountability in Computer Systems

Oxford Handbook of the Ethics of Artificial Intelligence

Chapter 9

Joshua A. Kroll

Assistant Professor of Computer Science, Naval Postgraduate School[*]

Capturing human values such as fairness, privacy, and justice in software systems is challenging. Values are abstract and may be contested, or at least viewed differently by different stakeholders, meaning they resist both definition and the concrete specification necessary to build machines or engineered systems. Choices in designing systems to embody values are political, and implicate structures beyond the system in question, trading off benefits and costs for different stakeholders. But this does not place computer systems beyond governance: the creators, operators, and controllers of such systems can and must be held accountable for the outcomes their systems effect. Accountability consists of a relationship focused on answerability: one agent or entity is accountable to another for certain outcomes in certain contexts. Operationalizing that accountability relationship requires keeping records – accounts – of how systems operated and were created. The entity to which an agent is held accountable can then determine responsibility, assigning praise or blame for the relevant outcomes and allocating consequences, ascribing moral valence to the agent's actions and the resultant outcomes. Most abstractly, judgements about responsibility can serve to establish the fidelity of system behaviors to operative social, political, legal, and moral norms. Accountability is the best framework for considering the governance of values in computer systems, providing a concrete and achievable approach to engaging abstract questions around values and ideals.

Thirty-seven seconds after the launch of the first Ariane 5 rocket on 4 June 1996, a software subroutine crashed, starting a chain reaction that led the launch vehicle to self-destruct. When the software attempted to convert a 64-bit floating point number to a 16-bit unsigned integer, the too-large value of the former could not be represented in the smaller format of the latter, triggering an unhandled error condition. Fortunately, this was a "hot standby", meant to take over in the event the active copy of the software failed. Unfortunately, the active copy was running the same computation and therefore also crashed almost immediately after. The buggy subroutine existed to keep the rocket balanced while on the ground and was unnecessary after liftoff, but had been left running beyond in case the launch was delayed momentarily. Cascading failures continued as the entire inertial reference subsystem crashed, causing incorrect data to feed into the rocket's guidance software. To correct what the guidance software erroneously understood as a deviation from the rocket's planned trajectory, but which was in fact bad data, the rocket's software control ordered the guidance nozzles on the main engine and the boosters to maximum deflection. This caused the rocket to veer wildly off course and experience "high aerodynamic loads", which tore the boosters off the main rocket, (correctly) triggering the rocket's self-destruct mechanism.[1] The result of this disaster was the complete loss of the launch vehicle and the onboard Cluster atmospheric research satellites, totaling about $370 million in direct losses. The failure set back the European Space Agency's efforts to develop its next-generation launch vehicle by several years, which to that point had run for 10 years at a cost equivalent to over US$7 billion.

Yet despite the proximate cause of the accident being a failure in the rocket's software, the inquiry board convened to analyze the accident recommended spreading responsibility across several functions in the development, design, and implementation of the launcher, saying that "When taking this design decision, it was not analysed or fully understood" meaning the "possible implications of allowing [the software] to continue to function [after liftoff] […] were not realized". The natural human instinct in the face of such failure is to identify the cause, assign responsibility for that cause to a person or group of people, and to tie that responsibility to consequences – in other words, to hold someone *accountable* for the failure. But after the Ariane 5 Flight 501 total launch failure, no individual, nor any part of the development team, was held directly responsible. Responsibility fell partially on several functions within the program – programmers, designers, requirements engineers, test engineers, and project managers – many of which could have exposed the failure ahead of time, but none of which did because each function focused on their chosen framing of their part of the project.[2] Along with other high-profile early software failures such as Therac-25,[3] the Ariane 5 failure contributed to decades of reflection in the software community about what is necessary to make software systems reliable in critical applications.[4]

Such reflection must also be applied to artificial intelligence (AI), a term which here refers to any behavior embodied in a machine (usually, a software system) which a human would consider intelligent. Concerns that such systems might not be reliable have led to calls for greater governance, especially as software systems have taken over an increasing number of critical application domains in modern society. Often, such automation augments traditional human decision-makers and professionals; sometimes, it outright replaces social and economic structures formerly mediated by humans with new structures mediated by software-driven

[*] Affiliation provided for professional identification purposes only. The views expressed in this chapter are those of the author and do not reflect official positions of the United States Government or its Department of Defense. The author is grateful to helpful comments from and discussions with Deirdre Mulligan, Emmanuel Moss, Abigail Z. Jacobs, Andrew Smart and Peter J. Denning. © 2020 Joshua A. Kroll. This chapter is available for reuse under the Creative Commons Attribution-Non Commercial Share-Alike 4.0 International License (CC BY-NC-SA 4.0), https://creativecommons.org/licenses/by-sa/4.0/. The required attribution notice under the license must include the chapter's full citation information, e.g., "Kroll, Joshua A. 'Accountability in Computer Systems' in *Oxford Handbook of the Ethics of AI*. Dubber, Markus D., Frank Pasquale, and Sunit Das, Eds. Oxford University Press. 2020."

[1] Lions, Jacques-Louis, Lennart Luebeck, Jean-Luc Fauquembergue, Gilles Kahn, Wolfgang Kubbat, Stefan Levedag, Leonardo Mazzini, Didier Merle, and Colin O'Halloran. "Ariane 5 Flight 501 Failure: Report by the Inquiry Board." (1996).

[2] Dowson, Mark. "The Ariane 5 software failure." *ACM SIGSOFT Software Engineering Notes* 22, no. 2 (1997): 84.

[3] Leveson, Nancy G., and Clark S. Turner. "An investigation of the Therac-25 accidents." *Computer* 26, no. 7 (1993): 18-41.

[4] These issues are by no means software-specific, but extend to all engineering in safety-critical contexts. See, e.g., Vaughan, Diane. *The Challenger launch decision: Risky technology, culture, and deviance at NASA.* University of Chicago Press, 1996 or Leveson, Nancy G. *Engineering a safer world*. The MIT Press, 2016.

machines.[5] This chapter of the Oxford Handbook of Ethics of Artificial Intelligence examines the relationship between such AI systems and the concept of *accountability*.

Definitions and the Unit of Analysis

To understand accountability in the context of AI systems, we must begin by examining the various ways the term is used and the variety of concepts to which it is meant to refer. Further, we must examine the *unit of analysis*, or the level of abstraction at which we consider the term to apply.[6] As with many terms used in the discussion of AI, different stakeholders have fundamentally different and even incompatible ideas of what concept they are referring to, especially when they come from different disciplinary backgrounds.[7] This confusion leads to disagreement and debate in which parties disagree not on substance, but on the subject of debate itself. Here, we provide a brief overview of concepts designated by the term "accountability", covering their relationships, commonalities, and divergences in the service of bridging such divides.[8]

Artifacts, Systems, and Structures: Where does accountability lie?

Accountability is generally conceptualized with respect to some entity – a relationship that involves reporting of information to that entity and in exchange receiving praise, disapproval, or consequences when appropriate. Successfully demanding accountability around an entity, person, system, or artifact requires establishing both ends of this relationship: who or what answers to whom or to what?

Additionally, to understand a discussion of or call for accountability in an AI system or application, it is critical to determine what things the system must answer for, that is, the information exchanged. There are many ways to ground a demand for answerability and give it normative force, and commensurately many types of accountability: moral, administrative, political, managerial, market, legal & judicial, relative to constituent desires, and professional.[9] AI systems intersect with all eight types of accountability, each in different ways and depending on the specifics of the application context.

Beyond the question of the normative backing for accountability is the question of to what unit it is applied: are we considering a single component, a larger system, or the entire structure of society in determining how accountability will be operationalized? Such unit of analysis

[5] Kroll, Joshua A., Solon Barocas, Edward W. Felten, Joel R. Reidenberg, David G. Robinson, and Harlan Yu. "Accountable algorithms." *U. Pa. L. Rev.* 165 (2016): 633.

[6] Selbst et al. refer to failures to understand the appropriate unit of analysis as "abstraction error" and define five "traps" representing common pitfalls in problem framing. See Selbst, Andrew D., danah boyd, Sorelle A. Friedler, Suresh Venkatasubramanian, and Janet Vertesi. "Fairness and abstraction in sociotechnical systems." *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pp. 59-68. ACM, 2019.

[7] Deirdre Mulligan, Kroll, Joshua A., Nitin Kohli, and Richmond Wong, "This thing called 'fairness': Disciplinary confusion realizing a value in technology." *Proceedings of the Computer Supported Cooperative Work Conference*. (2019)

[8] An alternative but similar taxonomy is presented in Wieringa, Maranke. "What to account for when accounting for algorithms: a systematic literature review on algorithmic accountability". In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency (FAT* '20)*. 2020. Association for Computing Machinery, New York, NY, USA, 1–18. https://doi.org/10.1145/3351095.3372833

[9] This taxonomy is due to Jabbra, Joseph G., and Onkar Prasad Dwivedi. *Public service accountability: A comparative perspective.* Kumarian Press, 1988.

questions apply both to determining what we are holding accountable and what we are holding it accountable to.[10] For example, in considering a system that predicts credit risk, we might choose to examine the instrument itself (i.e., it adequately reflects the borrower's risk of default), the larger sociotechnical context including applicants and loan officers (i.e., it functions adequately in the administration of lending, comports with actors' understanding of how it should behave, and is not subject to gaming), or the overall structure of credit analysis and lending (i.e., it does not systematically undermine credit markets or the provisioning or distribution of goods and services and does not unduly discriminate against structurally subordinated groups). Similarly, we may wish to hold it accountable to standards at a variety of levels of abstraction: instrumentally, we may demand that the score is correct if and only if it adequately rates risk or rates risk in an equal way across demographic groups, holding the system's performance to an objective and mathematical standard of *correctness*; at a systems level, we might hold the score to a standard of defensibility in litigation or another *oversight* mechanism; at a societal level, we might ask whether the distribution of risk elucidated by the score is the correct and morally appropriate distribution, a standard of *fidelity to normative goals*. Determining the extent to which each of these standards is met requires different approaches based on the level of analysis by different actors: correctness relates to technical decisions about a system's design; oversight implicates a specific entity or policy in receiving and examining answers about how a system behaved; normative fidelity is constructed through social and political processes, and often systems affect the operative norms just as much as they are constrained. Correctness, here, has two meanings: fidelity to a specification (the usual meaning in engineering) and consonance with normative context. That is: does a system follows the rules we have laid out for it? And are those rules the right rules?

Often, the unit of analysis referenced by someone discussing accountability relates to their disciplinary training and orientation: those interested in technology development, design, and analysis are more likely to conceptualize the system-as-embodied, situating algorithms and the agency of AI systems within machines themselves, or with their designers (i.e., technologists focus on the computers, the software, and the interfaces, or the engineering process). Political, social, and legal demands for accountability often focus around higher-order units such as sociotechnical systems of artifacts interacting with people or entire paradigms of social organization (i.e., policy discussions are often focused on systemwide outcomes or the fidelity of systems to democratically determined goals, looking at the company or agency involved and operative policy rather than the specific tools in use or their performance characteristics). Often, all units of analysis inform appropriate interventions supporting accountability, as the information necessary to establish system-level accountability may depend on metrics and measures established at the technical level. Thus, accountability must be part of the design at every scale, in tandem.

Related to the unit of analysis question is the issue of causal and moral responsibility. When operationalizing accountability, it is important that the relationship of answerability corresponds either to its subject causing the condition for which it is answerable or to its being morally culpable for that condition. If no such link exists, or if the information conveyed via the accountability relationship does not establish the link, then it is difficult to find the actor

[10] Bijker, Wiebe E., Thomas Parke Hughes, and Trevor J. Pinch, eds. *The social construction of technological systems: New directions in the sociology and history of technology.* MIT Press, 1989.

accountable. Operationalizing accountability in AI systems requires developing ways to make such links explicit and communicable. For example, the scapegoating of a component or portion of the problem can impair agency of the involved actors in establishing fault. Additionally, the problem of many hands can serve as a barrier to accountability, as it did in the Ariane 5 Flight 501 failure.[11] While many hands were responsible for that failure, this need not be the case: alternative governance structures for such multifaceted, cross-functional development teams could explicitly make leaders responsible, providing an incentive for them to ensure adequate performance and the avoidance of failures across their organization, or use other mechanisms to make domains of answerability clear at the level of functions or organizations. For example, legal proceedings often hold organizations (say, corporations) accountable at an abstract level, leaving the determination of individual accountability to happen inside the organization.[12] But these as well can be their own sort of scapegoating – accidents in autonomous systems are often blamed on "human error" even when the human has little meaningful control over what the system is doing.[13]

Accountability, Oversight, and Review

If we conceptualize accountability as answerability of various kinds, and we understand who must answer, for what, and to whom the answers are intended, then we have redeveloped the concept of *oversight*, a component of governance where a designated authority holds special power to review evidence of activities and to connect them to consequences. Oversight complements regulatory methods in governance, allowing for checks and controls on a process even when the correct behavior of that process cannot be specified in advance as a *rule*. Rather, an oversight entity can observe the actions and behaviors of the process and separate the acceptable ones from the unacceptable ones *ex post*. Further, when rules exist, an oversight entity can verify that the process acted consistently within them. Even when guidance is expressed in standards or principles, oversight can apply those more abstract desiderata in a given case, weighing considerations against each other given scenario-specific facts and circumstances.

In computer science, and in engineering generally, the twin modalities of guaranteeing compliance with a formally stated policy *ex ante* and keeping records which provide for auditing *ex post* have long been recognized as the major approaches to understanding the fidelity of an artifact to goals such as correctness, security, and privacy.[14] However, the dominant modality – whether building software and hardware controllers; rockets and aircraft; or bridges and buildings – has been to decide on a rule up front, to express this rule as a set of requirements for the system, to implement the system so that it is faithful to those requirements, and to verify that the implementation comports with the requirements while also validating that the requirements

[11] Nissenbaum, Helen. "Accountability in a computerized society." *Science and engineering ethics* 2, no. 1 (1996): 25-42. In the Ariane 5 case, an inability to hold an individual or specific function accountable is not necessarily a failure – the European Space Agency was certainly responsible for the incident and able to lay blame on the entire development organization in ways that significantly changed engineering practices.

[12] A barrier to accountability specific to AI systems in this sense is that liability for software products is often disclaimed by vendors and is in any case difficult to establish in product safety law. For an overview, see Choi, Bryan H. "Crashworthy code." *Wash. L. Rev. 94* (2019): 39.

[13] Elish, Madeleine Clare. "Moral crumple zones: Cautionary tales in human-robot interaction." Engaging Science, Technology, and Society 5 (2019): 40-60.

[14] Weitzner, Daniel J., Harold Abelson, Tim Berners-Lee, Joan Feigenbaum, James Hendler, and Gerald Jay Sussman. "Information accountability." *Communications of the ACM* 51, no. 6 (2008): 82.

capture the desired high-level goals. In this way, conformance of the artifact to a rule can be known ahead of time. Such an approach is quite powerful, and is often highly desirable (for example, we wish to know that a bridge will support a certain weight across a span before materials are expended in its construction). However, it is insufficient where the rules are exceedingly complex, contested, or require interpretation in order to be enforced, features of domains where AI systems are most desirable. Such domains include the application of many laws stated as principles, including data protection rules (e.g., determining whether consent is "informed"), copyright (e.g., establishing whether copying constitutes fair use), the use of protected data by law enforcement or for intelligence activities (e.g., granting orders allowing investigators access to protected information), and situations where there exist concerns about fairness, bias, or nondiscrimination.[15] Even in the many common cases where laws are expressed as standards or duties of care, there can be "best practices" that are amenable to implementation, but the many data breaches of organizations which have been certified to comply with such best practices attest to the fact that such practices often do not speak to the equities at stake. Further, the practical enforcement of laws could be described as the process of managing exceptions without risking the rule.

Further, AI is often employed in domains where specifying what rule should apply is difficult, and so the precise contours of the rule are left up to the system. For example, it would be prohibitively difficult to define a rule deductively for identifying objects in images (however, this approach has been a focus of many decades of AI research), though methods which use pattern extraction from large volumes of annotated data have recently proved useful and achievable. Such pattern extraction methods are particularly ill suited to applications where interpretation is required. But when the details of a rule are deferred to the operation of an AI system, the rule escapes even the basic level of scrutiny, verification, or validation present when rules are carefully constructed by engineers and policymakers. Further, because the rule is not set *ex ante*, it is difficult to disclose the rule or assess whether it meets operative normative guidance.

Enabling governance beyond setting rules is critical, as many contexts resist formalization as concrete rules. The proper operationalization of certain value-sensitive concepts, such as fairness, may be contested among stakeholders. Achieving political consensus in such cases may require intentional vagueness or deferral of authority to a designated entity (for example, legislatures generally defer the specifics of rulemaking to regulatory authorities, who may be more knowledgeable and better able to react to changing circumstances, and both also defer the specifics of administering the law in particular cases to courts and judges, who can balance values which are in tension and who can review cases with more certainty as to what happened as their view is retrospective, not prospective). Again, because AI systems defer the details of shaping rules into the operation of the system, they sidestep these balancing, refinement, and consensus-building processes, which are critical to develop a system's legitimacy.

[15] Weitzner, Daniel J., Harold Abelson, Tim Berners-Lee, Joan Feigenbaum, James Hendler, and Gerald Jay Sussman. "Information accountability." *MIT Technical Report* MIT-CSAIL-TR-2007-034. June 13, 2007.

Beyond this, some concepts may be *essentially contested*,[16] meaning that while stakeholders agree on the broad outlines of the concept in question, inherent in that agreement is a disagreement about the correct way to realize it in the world. Fairness is an excellent example – although many (or all) stakeholders in a particular context may wish an AI system to behave fairly, what is fair for some may not be fair for others. Setting out rules for what constitutes fairness must, of its nature, set these stakeholders in tension with each other. Privacy has also been described as an essentially contested concept.[17] Accountability provides a framework for reorienting this problem: stakeholders may be able to agree on a process or mechanism for weighing countervailing concerns in particular cases even when they cannot agree on the proper operationalization of acceptable vs. unacceptable behavior for a system up front. That is, acknowledging that definitions may not be possible, a case-by-case process for resolution may serve the operative value well enough in practice for the purpose of all stakeholders. Further, deferring enforcement can make space for the interpretive nature of goals expressed as standards or principles, rather than via the mechanical operation of a rule.

As oversight is critical to operationalizing accountability in practice, building AI systems which facilitate accountability (to some entity, for some property) necessitates designing those systems to support robust oversight. This implies establishing evidence of how they were created and how they are operating, enabling the job of the overseer.[18] In this way, accountability is tied directly to the maintenance of records (though records alone do not provide accountability and other requirements may be present). The job of the oversight entity can be characterized as applying appropriate norms from the context of the AI system's deployment to tie the actions described in those records to consequences. Oversight is of particular importance for AI systems, as it bridges the gap between engineering capacity to model values and governance goals and the substantive requirements of those goals. In particular, oversight can establish the regularity of agreed-upon procedures even when the substance served by those procedures is subject to disagreement among stakeholders.

Accountability as Accounting, Recordkeeping, and Verifiability

The simplest definition of accountability is in terms of accounting, that is, keeping records of what a system did so that those actions can be reviewed later. It is important that such records be faithful recordings of actual behaviors, to support the reproducibility of such behaviors and their analysis. Additionally, such records must have their integrity maintained from the time they are created until the time they must be reviewed, so that the review process reliably examines (and can be seen by others to examine) faithful records that describe what they purport to describe. Finally, it is important that both the fidelity and the integrity of the records be evident both to the overseer and anyone who relies on the overseer's judgements. Oversight in which the entity being reviewed can falsely demonstrate compliance is no oversight at all.

[16] Gallie, Walter Bryce. "Essentially contested concepts." In *Proceedings of the Aristotelian Society*, vol. 56, pp. 167-198. Aristotelian Society, Wiley, 1955.

[17] Mulligan, Deirdre K., Colin Koopman, and Nick Doty. "Privacy is an essentially contested concept: a multi-dimensional analytic for mapping privacy." *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* 374, no. 2083 (2016): 20160118.

[18] Kroll, Joshua A. "The fallacy of inscrutability." *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* 376, no. 2133 (2018): 20180084.

In some cases, the causes of behaviors can be "black-boxed" – ignored for the purposes of recordkeeping. This is the case, for example, with human bureaucracies. We cannot demand a full causal explanation for the behaviors and opinions of the human functionaries in such a structure. Even if we could, an explication of their behavior in terms of neuronal activations and connections would be so complex as to be meaningless, providing little in the way of epistemic grounding for the outcome of the bureaucratic process. Instead, such processes develop explanations and justifications which are appropriately selective and contrastive, describing what needs to be known to the correct people at a useful level of abstraction.[19] Thus, determining where and how to keep records of AI system behaviors is an important design consideration. The best way to determine which records best support accountability is to determine what oversight is necessary and to determine how to facilitate that oversight. Additionally, records are often useful directly for the subjects of decisions by AI systems or the public at large. When this is the case, the system design should also involve questions of how to develop direct accountability to subjects or the public, rather than accountability which is intermediated through political trust in an oversight entity.

Recordkeeping is a common operationalization of accountability in Computer Science and other technology-oriented fields.[20] Feigenbaum et al. provide a survey, taxonomizing recordkeeping along the dimensions of time and goals (when are records kept? What sorts of violations of policy do records aim to capture?), information (what information is learned about policy violations and policy violators?), and action (what, if any, actions are taken based on records of policy violations?).[21] This approach views accountability with respect to a concretely defined policy and violations of that policy. Some authors go as far as to define accountability as the property that any policy violation can be attributed to the violator in a way that allows the assignment of blame. However, as we have seen in the concept of oversight, accountability need not depend on the existence of a prespecified, concrete policy – it may also operate by synthesizing a policy extensionally *ex post* (i.e., based on the analysis of particular cases). Additionally, the existence of records does not immediately imply that a system is truly answerable for its behaviors or for outcomes caused by those behaviors. Records which are ignored, unseen, or simply not acted upon do little to facilitate accountability. We must expand the concept of accountability to tie the content of the records to the broader principle of responsibility.

Accountability as Responsibility

Answerability includes not just the notion that answers exist, but that individuals or organizations can be made to answer for outcomes of their behavior or of the behavior of tools they make use of. Responsibility ties actions or outcomes to consequences. Authors in this space have identified three major normative bases for this connection: *causality*, *fault*, and *duty* – either the actions of the entity being held accountable caused the outcome being considered, or the entity is somehow culpable for the outcome irrespective of cause, or the entity is ascribed an

[19] Miller, Tim. "Explanation in artificial intelligence: Insights from the social sciences." *Artificial Intelligence* (2018).

[20] Espeland, Wendy Nelson, and Berit Irene Vannebo. "Accountability, quantification, and law." *Annu. Rev. Law Soc. Sci.* 3 (2007): 21-43.

[21] Feigenbaum, Joan, Aaron D. Jaggard, Rebecca N. Wright, and Hongda Xiao. "Systematizing 'accountability' in computer science." *Technical Report* YALEU/DCS/TRE1452, *Yale University* (2012).

obligation to certain behaviors. All three types of responsibility, and the relationship of any to accountability, are subtle and bear unpacking. Operationalizing any one or all three to make practical the necessary accountability mechanisms and regimes is the subject of much work across several disciplines.

The notion of causality is itself a complicated question with a rich history of inquiry in the form of metaphysics; we leave this history aside here. However, the dominance of the scientific approach to understanding causation in the development of technical artifacts and especially AI systems is relevant to our inquiry.[22] Because scientific approaches look to full, mechanistic explanations and experimentally validated knowledge to establish facts, they can struggle to establish the causes of some phenomena or to distinguish causal relationships from other relationships. For example, in situations where variables are confounded, it can be challenging to establish whether a measured effect is causal or illusory.[23] Confounding occurs when multiple factors correlate with a certain outcome, and there is confusion over which associations represent the cause, limiting the extent to which any one can be assigned responsibility. In building a machine learning system for predicting mortality risk in pneumonia patients, researchers discovered that patients previously diagnosed with asthma performed better as a group, and as a result models rated them at a lower risk of near-term death. Domain experts (doctors) disagreed, noting that asthma patients have much higher fatality risk from pneumonia than patients without an asthma diagnosis. The problem lay in a quirk of the training data: by hospital rule, patients diagnosed with pneumonia and previously diagnosed with asthma were automatically admitted to intensive care, giving that cohort more aggressive treatment and more careful monitoring, leading to better outcomes and confusing the statistical models.[24]

Further, events often have multiple causes and reasoning about an appropriate set of causes for an event is challenging. Modern mechanisms for reasoning mathematically about causality generally only reason about simple causation or causation in the context of controlled experiments (which are often not possible for questions of interest), leading to a situation where inferences about causality formalisms tell only a portion of the story.[25] Causal analysis often proceeds by reasoning about *counterfactuals*, claims about the state of the world that would have resulted if some event did not occur, if some new event did occur, or if some observable feature of the world were different. In the context of reasoning about accountability in AI systems, counterfactuals present an interesting difficulty: when we consider how a system might have behaved in a hypothetical world different from the one we inhabit, we must understand the relationship between these worlds to interpret the counterfactual. The simplest sort of counterfactual merely introduces or removes a putative cause. In practice, the situations about which we wish to reason can involve complicated interactions or implicate existing social structures, configuring the hypothetical counterfactual world in a way that is very unlikely from the perspective of our world. For example, simply changing an individual's race or gender while holding other attributes the same is unlikely to produce a counterfactual case that can be

[22] Bunge, Mario. *Causality and modern science*. Routledge, 2017.

[23] Momin Malik. "A Hierarchy of Limitations in Machine Learning". Preprint. arXiv: 2002.05193. https://arxiv.org/abs/2002.05193

[24] Caruana, Rich, Yin Lou, Johannes Gehrke, Paul Koch, Marc Sturm, and Noemie Elhadad. "Intelligible models for healthcare: Predicting pneumonia risk and hospital 30-day readmission." In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1721-1730. ACM, 2015.

[25] Pearl, Judea. *Causality*. Cambridge university press, 2009.

analyzed in a sensible manner.[26] Concepts such as race and gender are co-constructed of a number of factors, and it can be challenging to find meaning in shifting the experience of a given subject so radically.

Still, causal responsibility is a key component of accountability for the simple reason that, if a system is to answer for its behavior, it is important to understand the causal origins of that behavior when possible. However, understanding the mechanisms of causation does not answer the question of why those mechanisms function in those ways, leading to the question of fault or *moral responsibility*. As in the dichotomy for correctness mentioned above, we can ask both what the mechanism of a decision was and, separately and normatively, whether that mechanism is the right mechanism that comports with social, political, and legal context and with values such as fairness and justice. Moral responsibility ascribes moral valence to both actions and responses to those actions, such as praise for conforming to an operative norm, or blame for violating it. This valence can be inherited from moral judgements about the operative norm itself, as well. Over and above causal responsibility, moral responsibility requires *agency*, or the ability to have behaved differently in a situation where control of the operative outcome could have been effected. For example, moral blame requires both that an entity is causally related to the event to which a moral ascription is being made and that the entity's actions were in some way faulty (that is, that different actions would in a moral sense have been better). Since Aristotle, philosophers have judged the appropriateness of moral blame by making moral judgements based on traits of relevant agents, explicitly vesting moral responsibility in the voluntary nature of a moral agent's control over its actions.[27]

This notion of agency raises an important sidebar about responsibility: the agents which can be held responsible are exactly those with sufficient agency to be ascribed causal responsibility, moral responsibility, or duties and obligations. In general, this implies that, while the objects of recordkeeping and accountability are generally machines, software, or algorithms, the entity being held answerable – the subject of accountability – must be a moral agent worthy of the ascription of responsibility. The ability to be assigned responsibility is, in key ways, tied to moral "personhood". Such personhood can vest with constructed persons – corporate and socially constructed entities – as well as with natural persons. The nature of holding constructed persons accountable is different to holding natural persons responsible as responsibility can lead to punishment for natural persons in much more direct ways than it can for constructed persons.

A concept tightly bound to responsibility and yet distinct from it is *liability*, the (often legal) ascription of responsibility for the plight of the victim in a particular scenario. Unlike accountability, which is a relational concept about responsibility in the sense of answerability for an action, liability is analyzed from the perspective of a debt owed to someone who has suffered

[26] An excellent overview of counterfactual reasoning as it applies to AI systems can be found in Miller, Tim. "Explanation in artificial intelligence: Insights from the social sciences." *Artificial Intelligence* (2018). A more detailed version of the argument against counterfactual reasoning about constructed attributes can be found in Kohler-Hausmann, Issa. "Eddie Murphy and the dangers of counterfactual causal thinking about detecting racial discrimination." *Northwestern U. Law Rev.* 113(5), 2019.

[27] Eshleman, Andrew, "Moral Responsibility", *The Stanford Encyclopedia of Philosophy* (Winter 2016 Edition), Edward N. Zalta (ed.).

harm.[28] Liability underscores the third category of responsibility, that of duty or obligation. Obligations may exist outside of answerability relationships. For example, a judge could be said to be responsible (in the sense of having a duty) for instructing a jury prior to their deliberations, but because that responsibility does not cause the judge to answer to a specific entity, we would not say that the judge is accountable for this (however, the judge could be accountable to higher courts, to voters directly, to competent representative bodies with authority to impeach, or via challenges to court procedure for failing to uphold this duty). Liability is not a substitute for accountability, although it can help to enforce or encourage accountability or to reify an agent's duties to encourage that agent to act or remain answerable for outcomes related to that agent's actions by assigning a financial cost to breaches of duties. Treating liability as a substitute for accountability leads to imperfect assessments of both. For example, in the Ariane 5 case, many different functions worked on the project and many people in each of those functions, obscuring lines of accountability. Yet the European Space Agency was very clearly liable for the cost of the failure and would have been liable for any related harms (for example, if the rocket had caused harm after exploding and falling to earth). Similarly, when liability is disclaimed by organizations, as it often is in the provisioning of software and AI tools, an agent using that software may have no control over how the software behaves, yet be unable to hold the creator of that software liable, let alone responsible.

Accountability as Normative Fidelity

The most abstract way that the term "accountability" is used connects the answerability relationship to broader norms, values, and fundamental rights. That is, when a system should uphold a particular political, social, or legal norm or be held to some moral standard, that requirement is often couched in terms of accountability in the sense of moral responsibility.[29] For example, Bovens, Schillemans, and Goodin observe that, in politics, "'[a]ccountability' is used as a synonym for many loosely defined political desiderata, such as good governance, transparency, equity, democracy, efficiency, responsiveness, responsibility, and integrity."[30] Political scientists often wonder whether accountability continues to hold meaning, even when operationalizing it is straightforward in the ever-growing number of places where it is claimed as desirable.[31]

And yet, accountability provides an achievable mechanism for approaching otherwise slippery and contested normative goals. While it might not be possible to agree on definitions of "fairness" or even of "discrimination", agents and entities are still accountable for their behaviors with respect to the operative norms. Although it is noble to pursue computer systems which are "moral", "ethical", or "fair", it is not clear how to operationalize this goal or how to tell when it has been achieved. However, agents which develop or rely on these tools can be made accountable for the outcomes they bring about, enabling judgements about when and how these agents are answerable on understandings of when operative norms have been violated.

[28] These ideas owe a great debt to Nissenbaum's work separating accountability and liability in Nissenbaum, Helen. "Accountability in a computerized society." *Science and engineering ethics* 2, no. 1 (1996): 25-42.
[29] Noorman, Merel "Computing and Moral Responsibility", The Stanford Encyclopedia of Philosophy (Spring 2018 Edition), Edward N. Zalta (ed.).
[30] Bovens, Mark, Thomas Schillemans, and Robert E. Goodin. "Public accountability." The Oxford handbook of public accountability (2014): 1-22.
[31] Mulgan, Richard. "'Accountability': An ever-expanding concept?" Public administration 78, no. 3 (2000): 555-573.

Accountability as a Governance Goal

This notion of accountability as normative fidelity demonstrates that accountability can serve as a governance mechanism. Because accountability is straightforwardly achievable and enables judgements about complex and contested values, it is a useful and tractable goal for governance. Systems can be designed to meet articulated requirements for accountability, and this enables governance within companies, around governmental oversight, and with respect to the public trust. Interested parties can verify that systems meet these requirements. This verification operates along the same lines that interested parties would use to confirm that any governance is operating as intended. Establishing lines of accountability forces a governance process to reckon with the values it must protect or promote without needing a complete articulation and operationalization of those values. This makes accountability a primary value for which all governance structures should strive.

Accountability vs. Transparency

Accountability is often associated with transparency, the concept that systems and processes should be accessible to those affected either through an understanding of their function, through input into their structure, or both. For a computer system, this often means disclosure about the system's existence, nature, and scope; scrutiny of its underlying data and reasoning approaches; and connection of the operative rules implemented by the system to the governing norms of its context.[32] Yet transparency is often insufficient and undesirable on its own; it is best conceptualized as an instrument for achieving accountability. Understanding and realizing other values – such as fairness, privacy, or nondiscrimination – requires shifting the focus from transparency to accountability in order to make those values cognizable and to recognize them as reified in the system.

For example, a lottery is a perfectly transparent process in the abstract, and yet ensuring that a computerized lottery operates faithfully (i.e., picks uniformly from the set of entries a designated winner) is an exceptionally difficult and fraught task. Even physical lotteries require elaborate ceremonies to demonstrate that all possible numbers have been entered into a physical mixing device and sufficiently randomized, without any extra selections becoming possible.[33] Although the core selection algorithm of a lottery is simple to understand and easy to program correctly, it relies on random choices that, by construction, must not be repeatable, making review of a lottery outcome intrinsically difficult – because any random choice is as good as any other, random values which are predictable to the lottery operator cannot be distinguished from ones which are not. Even a correctly implemented software lottery can be run at low cost millions or billions of times, creating a set of options from which a preferred winner can be selected *ex post*. The problem of demonstrating that every entry in the lottery was considered on equal footing and that no additional illegitimate entries were added is difficult, though feasible to solve with modern computer science. Transparency alone is insufficient to ensure that a lottery effects its fairly simple goals. Instead, the entire process must make clear that the properties required of its outcomes hold, and that violations of those properties will be detectable, to know when the actors responsible have deviated from the goal or when the outcome is illegitimate for other

[32] Pasquale, Frank. The black box society. Harvard University Press, 2015.

[33] Kroll, Joshua A., Solon Barocas, Edward W. Felten, Joel R. Reidenberg, David G. Robinson, and Harlan Yu. "Accountable algorithms." *U. Pa. L. Rev.* 165 (2016): 633.

reasons and can be held accountable. Similarly, the actors can be praised if the process operates faithfully.

Beyond this insufficiency, transparency can be undesirable in certain contexts, leading to situations where the subjects of decisions can alter their behavior strategically to violate an operative norm. For example, if procedures at a military installation’s guarded gate are always the same, an adversary can establish the weaknesses in those procedures and exploit them. To prevent this, procedures are often changed regularly. Yet, if an adversary knows which procedures will be in effect on which day, they can use that knowledge to attempt to overcome the procedures when they are weakest, gaining access to the installation on days when guards are most lackadaisical. The same logic applies to employees pilfering cash from a till, to burglars approaching their target, or to smugglers crossing a border or other control point. More generally, use of some measure as a target for control often leads people to change their behavior to maximize their benefit, a phenomenon known as *Goodhart’s Law*[34]; for example, when test scores are used as a measure of educational achievement and student achievement is the core measure of teacher performance, teachers are incentivized to train students to perform well on known tests rather than to understand the underlying material, confusing the practices of education and training.[35]

Finally, full transparency often trades off with other values related to confidentiality. Whether that is expressed as the personal privacy of individuals affected by a computer system or the proprietary intellectual property interests of the system’s creators or operators, the level of transparency required for governance often trades off against the disclosure of legitimate secrets. For this reason as well, it is best to think in terms of answerability relationships and accountability when establishing computer system governance mechanisms.

Mechanisms for Accountability in AI

Of course, transparency is a useful tool in the governance of computer systems, but mostly insofar as it serves accountability. To the extent that targeted, partial transparency helps oversight entities, subjects of a computer system’s outputs, and the public at large understand and establish key properties of that system, transparency provides value. But there are other mechanisms available for building computer systems that support accountability of their creators and operators.

First, it is key to understand what interests the desired accountability serves and to establish the answerability relationships: what agents are accountable to which other agents (“accountability of what?” and “accountability to whom?”), for what outcomes, and to what purpose? Once these are established, it is clearer which records must be kept to support interrogation of this relationship and to ensure that blame and punishment can be meted out to the appropriate agents in the appropriate cases. These records must be retained in a manner that guarantees that they relate to the relevant behavior of the computer system, representing the relationship between its inputs, its logic, and its outputs faithfully. This can be accomplished with the tools of modern

[34] Goodhart, Charles AE. "Problems of monetary management: the UK experience." In Monetary Theory and Practice, pp. 91-121. Palgrave, London, 1984.

[35] Espeland, Wendy Nelson, and Michael Sauder. "Rankings and reactivity: How public measures recreate social worlds." American journal of sociology 113, no. 1 (2007): 1-40.

computer science: cryptography, software verification, and the value type systems of computer programming languages. Record fidelity can be maintained across time and space using cryptography as well.

Beyond mechanisms that apply specifically to software, however, it is important to consider accountability and governance mechanisms that relate desired accountability relationships to the process of engineering and design and the function of organizations such as the companies that create software artifacts. Such tools include practices that encourage structured reflection on needs for an engineered system and how they should be captured in design; rules demanding the documentation of requirements and specifications; rules demanding testing and acceptance validation to ensure that produced artifacts comport with their documentation; and rules demanding documentation for users, operators, and oversight entities. Additionally, organizations can structure review processes adversarially, and maintain rules requiring multiple authority to effect changes to documentation or code, documenting the change management accordingly. Organizations can (and often do) demand that requirements or specifications be reviewed by expert teams for security and privacy practices, compliance, and readiness for release. Further, organizations can demand (or be required by policy) that their staff produce documentation for regulators or the public, such as impact assessments which disclose possible adverse effects of the systems being constructed.[36] Public documentation serves its function even when, and largely because, its creation forces organizations to consider how to develop systems which can be presented in the best possible light. Organizations should also ensure that the people or functions within the organization which are responsible for particular domains are clearly articulated and that these domains of responsibility are documented and widely understood. Finally, systems generally arise from a *lifecycle*, which must truly be a cycle: performance of the final system must be measured, evaluated, and considered against initial goals for future updates, fixes to the system as deployed, or workarounds for issues not immediately addressable.

Consider the Ariane 5 failure in this framework: would thinking in terms of accountability tools have prevented the failure? The failure was caused by an explicit decision not to protect numeric conversions into certain hardware registers for the sake of efficiency, although this decision had been taken for the previous vehicle generation, the Ariane 4 and the code blindly re-used. With clearer lines of responsibility for failure, it is likely that additional preflight simulation and testing could have been demanded and the problem identified. Further, more careful systems engineering would have revealed that allowing a subroutine needed only on the ground to run after liftoff was not as harmless as was believed, or at least would have invited more careful evaluation of pre-launch processes and the best way to handle momentary launch delay. One contemporary author noted of the failure that “Ariane 5 should teach us that there are "political" facets of engineering processes. A good process needs to regulate not only how systems are designed and developed, but also how high-level decisions about that design and development are arrived at.”[37] In this light, the fact that no engineering function could be held accountable for

[36] Reisman, Dillon, Jason Schultz, Kate Crawford, and Meredith Whittaker. "Algorithmic impact assessments: A practical framework for public agency accountability." AI Now Institute (2018).
[37] Dowson, Mark. "The Ariane 5 software failure." ACM SIGSOFT Software Engineering Notes 22, no. 2 (1997): 84.

such a massive failure seems hardly surprising, even when the failure can be proximately traced to clear errors in the construction of software.

Whither Accountability in AI?

Where do these ideas lead us for accountability in AI systems? What ends does accountability serve and what are the means to achieving them? Human values are political questions, and reflecting them in AI systems is a political act with consequences on the real world. We can (and must) connect these consequences to existing political decision-making systems by viewing the gap between system behaviors and contextual norms in terms of accountability. For example, if we want to know that an AI system is performing "ethically", we cannot expect to "implement ethics in the system" as is often suggested. Rather, we must design the system to be functional in context, including contexts of oversight and review. Only then will we be able to establish trust in AI systems, leveraging existing infrastructures of trust among people and in institutions to new technologies and tools. Thus, the prime focus of building ethical AI systems must be building AI into human systems in a way that supports effective accountability for the entire assemblage.

While the need for such practices is great, and while it is critical to establish what engineered objects are supposed to do, including what is necessary to satisfy articulated accountability relationships, the actual reduction to practice of such tools in a way that demonstrably supports accountability and other human values remains an important open question for research. While many tools and technologies exist, only now are we beginning to understand how to compose them to serve accountability and other values.

A Brief Bibliography on Accountability in AI:

1. Kroll, Joshua A., Solon Barocas, Edward W. Felten, Joel R. Reidenberg, David G. Robinson, and Harlan Yu. “Accountable algorithms.” *U. Pa. L. Rev.* 165 (2016): 633.
2. Nissenbaum, Helen. “Accountability in a computerized society.” *Science and engineering ethics* 2, no. 1 (1996): 25-42.
3. Weitzner, Daniel J., Harold Abelson, Tim Berners-Lee, Joan Feigenbaum, James Hendler, and Gerald Jay Sussman. “Information accountability.” MIT Technical Report MIT-CSAIL-TR-2007-034. June 13, 2007.
4. Wieringa, Maranke. “What to account for when accounting for algorithms: a systematic literature review on algorithmic accountability.” *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, pp. 1-18. 2020.
5. Feigenbaum, Joan, Aaron D. Jaggard, Rebecca N. Wright, and Hongda Xiao. “Systematizing ‘accountability’ in computer science.” Technical Report YALEU/DCS/TRE1452, *Yale University* (2012).
6. Kroll, Joshua A. “The fallacy of inscrutability.” *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* 376, no. 2133 (2018): 20180084.
7. Miller, Tim. “Explanation in artificial intelligence: Insights from the social sciences.” *Artificial Intelligence* (2018).
8. Reisman, Dillon, Jason Schultz, Kate Crawford, and Meredith Whittaker. “Algorithmic impact assessments: A practical framework for public agency accountability.” AI Now Institute (2018).
9. Desai, Deven R., and Joshua A. Kroll. “Trust but verify: A guide to algorithms and the law.” *Harv. J. L & Tech.* 31 (2017): 1.
10. Wachter, Sandra, and Brent Mittelstadt. “A right to reasonable inferences: re-thinking data protection law in the age of big data and AI.” *Columbia Business Law Review* (2019).
11. Breaux, Travis D., Matthew W. Vail, and Annie I. Anton. “Towards regulatory compliance: Extracting rights and obligations to align requirements with regulations.” In *14th IEEE International Requirements Engineering Conference* (RE'06), pp. 49-58. IEEE, 2006.