Towards Ethical Data-Driven Software: Filling the Gaps in Ethics Research & Practice

Brittany Johnson
Department of Computer Science
George Mason University
Email: johnsonb@gmu.edu

Justin Smith
Department of Computer Science
Lafayette College
Email: smithjus@lafayette.edu

***Abstract*—More and more, data is being used to drive automated decision making through the integration of machine learning technologies into software. With these advances comes new potential for unexpected, undesirable, and possibly dangerous outcomes for end users. This has led to an increased focus on ethics in technology in both research and practice. Much of the work in ethical practices has been centered on ethics in machine learning and little has been validated for effectiveness in practice. In this paper, we outline the existing work in ethical computing with a focus on efforts that are relevant to data-driven software development. Based on existing work, we identify gaps in our understanding of ethical software development practices and suggestions for future work to help close these gaps.**

I. Introduction

Data-driven software is increasingly used to make decisions that affect our day-to-day lives. From influencing the products we buy [1] or who gets hired for a job [2], to diagnosing and treating medical patients [3] or policing neighborhoods [4], data-driven software has the potential to have a major impact on our quality of life.

Advances in software development, such as the integration of machine learning that uses real world data for decision making, have changed how we build software systems [5], [6] and forced developers to account for additional concerns [7]. In particular, as data-driven software is becoming more ubiquitous, software teams are increasingly concerned with ethics [8]. This concern is validated by several troubling examples of detrimental societal outcomes caused by the integration of machine learning technologies in software, such as racial bias in criminal justice [9], [10] and medical software [11], gender bias in hiring [12], safety concerns in self-driving cars [13], and bots propagating misleading and potentially dangerous information [14], [15].

The prevalence, severity of, and continued potential for these detrimental societal outcomes motivates us to better understand how ethics are (or are not) practiced in data-driven software development. As a first step toward this goal, this paper presents a preliminary survey of the literature on ethics in computing. To scope this review, we are specifically interested in work that has been done to understand and improve the state of the practice in artificial intelligence, machine learning, and software that integrates the former.

While we do discuss several important theoretical works, our primary focus in this paper is on actionable interventions and empirical research studies. In this paper, we define an *intervention* as anything that can be applied or used to alter or adapt the software development process; this includes tools, frameworks, strategies, and techniques. Empirical research, particularly in the context of software engineering, consists of studies based on observation and experimentation that allow for the assessment of software technologies and artifacts [16]. Empirical studies help dispel strongly held beliefs and misconceptions, as well as provide evidence-based suggestions for changes and improvements in research and practice [17].

Based on our survey of existing work, we also identify several gaps (Section V) in our understanding of ethical software development and ideas for future research that can fill these gaps. In summary, we argue that the gaps in existing research and practice are:

- The need for additional studies to confirm whether limited preliminary findings generalize and scale to different settings and contexts (Section V-A).
- Few interventions that take into account the internal and external support that practitioners seek (Section V-B).
- Little understanding of how ethical concerns manifest throughout the *entire* data-driven software pipeline (Section V-C).
- Relatively few interventions that account for ethical concerns beyond fairness (Section V-D).
- The need for empirical validations of proposed ethics interventions (Section V-E).

The contributions of this work are:

1) A survey of existing research, literature and interventions related to ethics in technology, ranging from theoretical contributions to empirical evaluations of proposed interventions.
2) An enumeration of gaps in existing work on ethical technology as it relates to data-driven software, with a focus on empirical user studies, along with ideas for how future work can help fill those gaps.

II. Background

Ethical considerations, including concerns like fairness, safety, and trust, are wide-ranging and vary depending on the software or context [18]. In this section, we expand on what this paper means by ethics, specifically in the context of

data-driven software development. We also summarize existing efforts in ethical development practices.

A. Defining Ethics

Broadly speaking, ethics refers to the idea that something is acceptable or right in a given context; this includes whether something is legal. For example, it is unethical, as well as illegal, to discriminate against someone based on their race or gender when considering them for a job. Over the years there have been many discussions on what ethics means and how it works in the context of computing and software development [19], [20]. The increased use of machine learning technologies in software has diversified the definition of ethics and made it a more prevalent issue. Rather than focus on any one definition of ethics, in this paper we discuss existing work that attempts to address ethical issues in technology and the potential for future work based on what has been done. In the next section, we discuss the focus of our paper, ethical concerns in the context of data-driven software.

B. Ethics in Data-driven Software Development

More and more, software decision-making is being driven by data from the real world through the integration of machine learning technologies. In the context of this paper, we refer to this kind of software as data-driven software [6]. These changes in how software works adds an additional layer of complexity to how software is built. Regardless of the approach, software development generally includes certain steps, such as requirements gathering, implementation, and testing [21]. The development and evaluation of machine learning technologies has its own set of steps, as shown in Figure 1. Data-driven software development involves integrating traditional software development with machine learning technology development.

Along with changes to the process of developing software, data-driven software comes with a need to further emphasize and incorporate ethical decision-making. Furthermore, there are some considerations that are unique to (or more greatly emphasized) in this context. More specifically, data-driven decision making raises concerns regarding bias, safety, and integrity in software outcomes that can have unintended, and even detrimental, effects on its users [6], [22].

Recent work has found that merely having a "code of ethics" may not be enough to affect software development practice with respect to ethical concerns [23]. Furthermore, most codes of ethics are not exhaustive of all the various ethical concerns in data-driven software. There exists a variety of efforts at providing interventions for integrating ethical considerations in practice in the context of machine learning technologies and software, which we discuss next.

C. Existing Ethics Interventions

Researchers and practitioners have proposed numbers interventions to better support the ethical development of machine learning technologies. Some of these contributions take the form of actionable frameworks and guidelines while others are tools that can be used in the development process. In this section, we outline the many existing contributions to ethical AI and software development practices.

1) Ethics frameworks: There have been numerous contributions to ethics in the form of frameworks, principles, or guidelines. Some of the frameworks or guidelines that have been proposed are specific to certain technologies or domains. Leidner and Plachouras propose best practices, inspired by previous work in privacy, for ethically designing natural language processing (NLP) systems [24]. Their work outlines the different ethical concerns relevant to NLP, ranging from fairness to unethical research methods. Based on these concerns, they propose a process centered around an Ethics Review Board whose purpose is to either approve or deny plans and propositions made regarding NLP research and development. Char and colleagues explored ethical decision making and consequences of those decisions in machine learning healthcare applications (ML-HCA), with an emphasis on supporting ethics throughout the pipeline of developing and evaluating these systems [25]. To this, they proposed a pipeline framework to help identify ethical considerations before, during, and after ML-HCA development and provide insights into how the framework could be used in practice.

Some existing frameworks are designed to be applicable to any technology with the potential for ethical concerns. Most relevant is work done by Vakkuri and colleagues, who propose a research framework for empirical studies on AI ethics in the context of software development practice [26]. This framework, built based on existing work in AI ethics, is centered around the ART principles (accountability, responsibility, and transparency) and how these relate to *Ethics in Design*, or software development interventions for supporting the implementation of ethics, and *Ethics for Design*, or standards and principles that ensure the integrity of both developers and users. These concepts are made actionable by using a commitment net model that focuses on developers' *concerns* and *actions* with respect to each component of the framework.

Also relevant to software development is the analytical framework for ethics-aware software engineering proposed by Aydemir and Dalpiaz [27]. Their framework helps analyze ethical issues in terms of stakeholder values, the process and artifacts involved in the software being built, and the potential effect on users. Along with a framework, this paper proposes a research roadmap to help bridge various gaps that exist in ethical software development, with a focus on capturing, specifying, and validating ethics-related requirements.

Mulvenna and colleagues propose the establishment of an ethical by design manifesto, starting with a set of principles that can support various software stakeholders in integrating ethical considerations into the design process [28]. In an attempt to bridge the gap between principles and practice in Human-centered AI (HCAI), Shneiderman proposed fifteen team, organization, and industry level governance guidelines for building reliable, safe, and trustworthy HCAI systems [29]. Presented as governance structures for HCAI, the proposed guidelines covers using software engineering practice to build

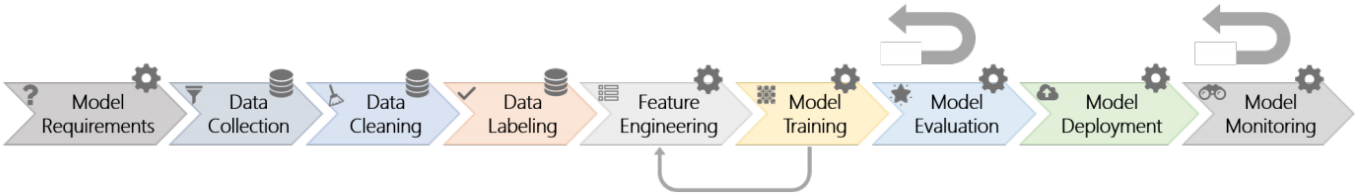

Fig. 1. The 9 stages of the machine learning workflow, as outlined by Amershi and colleagues [6].

reliable systems, using business management strategies to develop an organizational safety culture, and using independent oversight by external organizations to certify trustworthiness.

While some interventions are designed to address a variety of ethical considerations, others are designed to address a specific ethical consideration, such as fairness. Thomas and colleagues propose a Seldonian framework for addressing fairness concerns by allowing for simplified specification and regulation of undesirable behavior in machine learning algorithms [30]. Their framework works by making the designer of the algorithm responsible for ensuring expected behavior, allowing the designer to define a goal, define the interface, and create a Seldonian algorithm. Chakraborty and colleagues developed a new algorithm, called Fairway, designed to mitigate ethical bias in data and models via pre- and in-processing methods [31]. A performance evaluation of Fairway found that the combination of pre-processing and in-processing techniques improves outcomes in comparison to using one or the other and that Fairway can help achieve fairness with little sacrifice to performance.

2) Ethics tools: Numerous tools have been developed to improve ethical software practices, many of which focus on increasing fairness in AI software. One of the first published efforts at providing fairness tooling was by Adebayo, who developed FairML, a toolbox aimed at mitigating fairness in black box machine learning models [32]. FairML focuses on the effects of inputs on a model's decision making to determine the effects on fairness.

In 2018, IBM introduced AI Fairness 360 (AIF360), a Python toolkit for measuring and mitigating bias in machine learning models [33]. The toolkit provides an exhaustive and extensible set of open source models and algorithms, along with fairness metrics for models and datasets. Similar to AIF360, Saleiro and colleagues developed Aequitas, a Python toolkit for systematic auditing of model fairness [34]. As with most fairness tools, Aequitas was designed to be used by data scientists, however, it was also designed for use by policy makers. It also provides tooling for analyzing bias in datasets and determining optimal metrics for a given situation.

In the same year, Angell and colleagues presented Themis, the first tool for testing software for discrimination [35]. Similar to existing fairness tooling, Themis works based on common definitions of fairness. In contrast, Themis allows for measuring and detecting bias in software separate from any model that may (or may not) be integrated into the software.

The contributions to this space continued in the years to come. Cabrera and colleagues presented FairVis, yet another fairness auditing tool [36]. Unlike prior tools, FairVis uses visualizations to support the ability to explore and discover biases in machine learning models. Schelter and colleagues developed FairPrep, a framework that builds on AIF360 and scikit-learn [37], [38]. Based on gaps in existing work, FairPrep was designed to support best practices for data cleaning, model configuration, and selection.

To address the ability to measure fairness in large machine learning systems, Vasudevan and Kenthapadi presented the LinkedIn Fairness Toolkit (LiFT) [39]. Based on experiences deploying LiFT at LinkedIn, the paper provides insights into their framework's effectiveness as well as challenges faced and lessons learned pertaining to the adoption and usage of fairness tooling in an industry setting.

Bird and colleagues introduced Microsoft's Fairlearn, a Python fairness toolkit with a goal of supporting data scientists in training fair models [40]. Similar to FairVis, Fairlearn uses an interactive visualization to support model exploration and understanding with respect to various fairness metrics. Fairlearn makes an additional contribution by supporting the ability to explore trade-offs between fairness and performance.

Yet another recent contribution to this space is FAT Forensics, a Python toolbox developed by Sokol and colleagues [41]. FAT Forensics goes beyond just fairness evaluation, supporting the inspection of accountability and transparency aspects of machine learning software as well. By design, FAT Forensics allows for the implementation, testing, and deployment of algorithms along with the evaluation and comparison against other datasets, models, and predictions.

As we have outlined, there is no shortage of interventions available for attempting to address ethical concerns in data-driven software. While some of these contributions provide performance evaluations on real-world datasets or provide functionality based on perceived needs from existing literature, none have been empirically evaluated to determine a) which tool is best to use in a given scenario or context, b) whether they provide the support they claim to provide in practice, and c) whether users find them usable and useful.

As stated by Vakkuri and colleagues, "most of the research on AI ethics has been conceptual and theoretical in nature." [42]. Before we outline empirical user research in software ethics, we discuss existing theoretical research.

III. Theoretical Research on Software Ethics

Existing research has exhaustively discussed and explored potential theoretical foundations for ethics in machine learning

software. Some of these discussions happen outside of there research community [43], [44], however, there exist numerous research articles devoted to the theoretical side of this topic.

A. Ethical Governance

One issue commonly discussed in existing work is how governance of ethics in artificial intelligence technology should or could work in practice. Yu and colleagues developed a taxonomy based on recent work in AI governance [45]. Their taxonomy is divided into four areas: *exploring ethical dilemmas*, *individual ethical decision frameworks*, *collective decision frameworks*, and *ethics in human-AI interactions*. Bannister and colleagues pose that organizations and their leaders should be more involved and invested in building ethical technology and suggest what they call an "ethical tech mindset" for building and disseminating technologies that can have disruptive effects on society, such as those that integrate artificial intelligence [46]. The main characteristics of this mindset are *shared, inclusive responsibility*, *being proactive rather than reactive*, *integration of tech ethics into existing ethical considerations*, *relevance and flexibility*, *resources for response to ethical challenges*, and *approaches that can evolve*.

Winfield and Jirotka propose an agile framework for guiding ethical governance in intelligent autonomous systems (IAS), such as driver-less cars and data-driven medical diagnosis systems [47]. This framework, presented as a roadmap for responsible research and innovation, emphasizes the importance of building public trust to reduce public fears of IAS, standards and regulation of IAS, verification and validation of safety-critical IAS decision-making, transparency, and moral machines as forms of ethical governance. In an attempt to operationalize this roadmap, the paper presents five pillars for good ethical governance: *publishing an organizational ethical code of conduct*, *providing training on ethics and responsible research and innovation*, *practicing responsible innovation*, *transparency in ethical governance*, and *sincere efforts towards ethical governance*.

Rashid and colleagues propose a preliminary framework for handling ethical considerations that can emerge when building software engineering in society projects [48]. Derived based on experiences managing ethical concerns in real-world projects and inspired by Boehm's Spiral Model of software development, the proposed framework attempts to address the inherent fluidity of ethics across various software systems and the contexts in which they may be deployed. The paper discusses four categories of emergent ethical concerns: *ethical misuse cases*, *unintended consequences*, *micro-ethics of emergent content, contact, and conduct risks*, and *differential vulnerability across user groups*. Their framework attempts to highlight when it would be most suitable to address each of these categories of emergent ethical concerns.

Borenstein and Howard propose a re-evaluation of how we educate future developers, designers, and professionals with respect to AI and ethical concerns that come with its integration into our day-to-day lives [49]. They argue that while providing solutions to ethical challenges is important, another important piece of the puzzle is creating a "professional mindset" around the ethical dimensions of building AI technologies and integrating lessons on professional and technical ethical decision making into the STEM curriculum. Bond and colleagues outline various Human-Centered AI (HAI) ethical challenges that can drive future efforts in improving interactions between humans and AI [50]. Their work outlines how explainable AI and ethically-aligned design can support the responsible development of AI technologies, but argues that there is a need for tools to help with integrating these concepts into existing processes.

B. Components of Ethics

Some existing work has attempted to build theoretical foundations for considerations related to ethics, such as fairness, explainability, and trust, rather than the concept of ethics as a whole. As suggested by the numerous tools we discussed in Section II-C, one of the most commonly explored aspects of ethical AI is fairness. Selbst and colleagues argue that foundational computer science concepts, like abstraction, render many technical interventions, such as fairness tooling and algorithms, ineffective and even misleading once the software it was used on enters the social context they are used in [51]. Their paper outlines five "abstraction traps" that they believe fairness centered AI work can fall into and, based on various studies of sociotechical systems, explains why these traps occur and how to potentially avoid them. All of these traps, such as the *framing trap*, center around consideration of abstractions that include social and societal factors.

Related to fairness is the notion of explainability. There have been various attempts at providing tools and techniques that can support improved explainability [52]. In contrast, Rudin argues that our focus should be on building interpretable models rather than finding ways to explain black box models, especially in software that makes "high-stakes decisions" [53]. This work outlines why explainability should be avoided along with challenges to fostering interpretable AI, and use examples, such as criminal justice software, to show how interpreable models could replace traditional black box models.

Another commonly explored component of ethics, which often ties together notions like explainability and fairness, is trustworthy AI. Kumar and colleagues target the development stage of building AI systems by proposing a way of formalizing requirements for trustworthy-by-design AI, with a focus on the ethics of algorithms and data [54]. They use a concrete use-case of data in smart cities to show how their proposed framework can be beneficial. Similarly, Jacovi and colleagues explore the formalization of trust in AI systems, but via a model of trust that is inspired by social theory on trust between people [55]. Their model is centered on two key properties: user vulnerability and anticipating the impact of AI model decisions. To strengthen their model and address the issue of when and whether an AI model has achieved its goal, they incorporate notions of contractual trust along with warranted and unwarranted trust.

C. Closing the Gap Between Theory and Practice

There has been some work that begins to explore how we might close the gap between theoretical ethics conversations and practice. Chang poses that one problem is a disjoint between realizing the importance of ethics and taking ethical actions [56]. Focusing on data privacy and protection, Chang explores existing work around data ethics, makes suggestions on why the disjoint exists, and proposes ways we can bridge the gap through *ethical education*, *highlighting desired behaviors*, and *cross-functional support*. In a similar vein, Mittelstadt argues that principles surrounding AI ethics, which align closely with principles of medical ethics, may not be a sustainable approach for guaranteeing ethics in AI systems [57]. The paper cites key differences between medicine and AI development, such as legal accountability and the establishment of professional history and norms, and provides some suggestions for moving beyond a principled approach to AI ethics. Building on a conceptual map from a 2016 review of algorithm ethics, Tsamados and colleagues discuss previous and updated ethical concerns and provide "actionable guidance" for algorithm governance based on recent works [58]. Through a review of updated literature, they found that the conceptual map is still relevant and suggest ways to address problems related to each component of the original map.

While these works begin to explore the connection between theory and practice with respect to ethical software practices, most of these focus on the field of artificial intelligence, with little mention of implications and interventions at the software level. Furthermore, similar to many of the interventions we discussed in Section II-C, none of the proposed interventions have been evaluated (empirically or otherwise) to instill confidence in the ability for them to be effective in practice. In the following sections, we will discuss interventions that have been evaluated and what we can glean from their findings with respect to ethical software development practices.

IV. Empirical Studies on Ethics in Software Engineering

Research suggests that empirical studies on interventions can provide an evidence-based foundation for improving practice [17]. In this section we discuss empirical studies that were conducted to evaluate specific interventions and their potential for improving ethical software development practices.

A. What do we know: validation of interventions

A handful of studies have proposed and empirically evaluated ethical interventions to provide insights on their potential benefits in practice. Amershi and colleagues created a set of 18 guidelines for human-AI interaction [59]. These guidelines include broader suggestions that are not necessarily directly linked to ethics, such as "Learn from User Behavior" and "Remember Recent Interactions." However, some guidelines clearly are targeted at helping improve the ethics of AI systems, such as "Match Relevant Social Norms" and "Mitigate Social Biases." To validate these findings, Amershi and colleagues conducted a series of empirical evaluations. Based on the results of their evaluations, which included a heuristic evaluation and user study, they refined their guidelines.

Lee and Singh empirically evaluated six prominent open source fairness toolkits [60] using exploratory focus groups, semi-structured interviews, and a survey. Their work identifies gaps between tools' capabilities and practitioner needs. For instance, they found that the evaluated toolkits require a high level of expertise in fairness and do not necessarily communicate effective mitigation strategies.

Vakkuri and colleagues adapted the RESOLVEDD strategy, a popular tool from the field of business ethics that enables ethically aligned design in a decision-making process, and applied it in the context of software development [61]. The intervention was evaluated empirically using a case study of five student projects. The findings of this study suggest that the presence of an ethical tool has effect on ethical consideration (even when tool use is not intrinsically motivated). However, in the context of AI ethics, the RESOLVEDD strategy is limited, because it does not consider the technical aspects of a system, only its overall design.

Madaio and colleagues examine and improve on a familiar tool, AI ethics checklists [62]. They argue that existing checklists are not empirically grounded in practitioners' needs. To address this shortcoming, Madaio and colleagues paired a series of empirical explorations alongside the checklist design process. Their design process included semi-structured interviews and co-design workshops. Their study found that while checklists may be beneficial, practitioners felt that checklists are not sufficient, citing needs for "organizational change" and "additional resources." Participants also mentioned concerns around checklists oversimplify fairness concerns and make them seem guaranteeable.

B. What do we know: State of practice

In this section we summarize previous studies that empirically examine the state of how practitioners view ethics. As opposed to the previous section, in which the prior work empirically evaluated specific AI ethics tools or interventions, the studies in this section approach the question more broadly.

Much of the limited work in this space has been conducted by Vakkuri and colleagues [63], [64], [42], [65]. In a series of studies, this research group has conducted case studies, ([63], [64]) semi-structured interviews, ([42]) and a survey of practitioners ([65]). Their findings characterize how software developers implement (or disregard) ethics in some types of AI-driven systems. For instance, in startup-like environments Vakkuri and colleagues report that developers take responsibility for issues related to software development, such as finding bugs, and they generally care about ethics on a personal level. However, little is done to tackle ethical concerns that arise during product development [42]. A separate study [64] reveals a similar disconnect in the development of autonomous cyber-physical systems—developers unanimously indicated that they

consider ethics useful to their organization, but also unanimously report that their practices do not account for ethics.

A small number of studies have examined the practices of developing ML applications [66], [67], however, these studies do not focus on ethics. Zhang and colleagues interviewed eight and surveyed 195 developers to better understand the process of developing ML applications. Their focus covers much of the data-driven software development pipeline, ranging from requirements analysis to deployment and maintenance, but none of their questions touch on ethics. Similarly, Nascimento and colleagues conducted interviews with seven developers at small companies to understand how they build ML systems. In this study, participants described how they built ML systems at various stages of the development process, from problem understanding to model monitoring. Participants also discussed the challenges they perceived during development. Again, ethics was not discussed as a concern during the development process or as a challenge that developers faced.

V. Towards Ethical Data-Driven Software

So far we have discussed the limited empirical research that has been done pertaining to ethics in AI technologies and software. The work that has been done thus far has ranged from studies specific to AI to studies of ethics in the context of software development. However, there are gaps in our understanding of ethics in practice especially in the context of data-driven software development. Below we discuss what is missing and how research can help fill the gaps in our understanding of ethical software development practices.

A. Generalizability and Scale

Though few in number, there exist empirical studies on ethical software development practices (Section IV-B). These provide valuable insights into what it may look like to integrate ethical considerations into software development, however, it is not clear whether the knowledge attained from these studies would scale across software companies and domains.

First, more studies are needed to explore the potential for ethical interventions scale. This includes not only studying companies of different sizes but also more mature and established software companies, given much of the work that has been done has focused on smaller and start-up companies. This is particularly important given the differences between startups and established software teams, both small and large [68], [69].

Existing work has also targeted specific projects at specific software companies, or specific software domains. Therefore, another aspect to increasing scalability, as well as generalizability, of findings regarding ethical software development practices is data collection and analysis across software companies that develop different kinds of software in different domains. This will not only provide insights into what can generalize, but also where there are ethical concerns that are specific to a given software domain. This includes studies that explore the differences in ethical concerns in different geographic locations that may have different ideas of what is or is not ethical.

B. Internal and External Support

Many of the recommendations regarding ethical technology development include the notion that the responsibility is distributed from the individual up to the organizational level. While the realization that the organization should play a role in providing and enforcing ethical interventions is an important one, more important is understanding any existing attempts at incorporating ethics. A better understanding of existing or past efforts would help identify potential gaps, challenges, and blockers to successful integration of ethical concerns at the organizational level.

While internal support is important, research has also found that software development teams often seek support from outside resources, such as those found in online programming communities, in their efforts at building high quality software [70], [71], [72]. Another useful direction for research would be to explore external support software teams look to when wanting to learn more about or improve their ethical software development practices.

C. Ethics Throughout the Pipeline

As we discussed in Section II-B, data-driven software development involves the integration of the machine learning development workflow with traditional software development. However, much the existing work in ethical technology focuses on integrating ethics into the machine learning workflow. Furthermore, little has been done to support ethics throughout the entire data-driven software development pipeline.

Therefore, one important research direction is gaining a more robust understanding of how we integrate ethical concerns into the various stages of data-driven software development. For example, rather than thinking about the fairness problem from a machine learning algorithm perspective, Themis provides support for analyzing software for discriminatory behavior [35]. This is a first glimpse into integrating ethical tooling at the software level, however, more work is needed to understand how this connects to the big picture that is data-driven software.

Much of the work that has been contributed to the ethics in computing landscape has been in the form of principles, guidelines, and codes. However, previous works suggest the existence of these kinds of interventions is just not enough [23], [57], [50]. Therefore, another important research direction is to understand how existing principles can be integrated into practice and the kind of support needed to make it successful.

D. Going Beyond Fairness

One of the most commonly explored area of ethics in computing is fairness. This is particularly true with respect to advances in tooling for ethical software development, as we have outlined in Section II-C2. However, the exhaustive work and discussion on ethics in computing has shown that there are many other ethical concerns outside of fairness. While there exist ethical guidelines and principles outside of fairness, research has shown that tooling plays an important role in developer productivity and software quality [73], [74].

Therefore, for ethical data-driven software development to become a reality, it is important we investigate the ability to develop and integrate tooling that supports addressing ethical concerns outside of fairness. This means that we will need first an understanding of what the major ethical concerns are in the context of data-driven software and how we can identify and address them in practice. Much of this will build on existing work in, for example, AI ethics, but as we discuss in Section V-C, an important consideration is the integration of AI and software development and any unique concerns that may arise or interventions that may be needed.

E. Empirical Validation of Ethics Interventions

Previous work in ethical technology has made many suggestions for improving the state-of-practice. However, many of these suggestions have not been evaluated by the scientific community. This leads to many important questions regarding suggested interventions, such as *do they work as intended?*, *do they help or hinder the development process?*, and *are they usable?*. Therefore, there is a need for more research that attempts to validate the potential for ethical software development in practice. For instance, consider interventions that are developed and disseminated in the medical research community; would you take a vaccine that has not been tested and evaluated on real people just because the doctor said it would work? The same should apply for software interventions. The more we know about how these interventions work, and what is needed by the populations that will use them, the more likely we are to see real change in practice.

VI. Conclusion

We have outlined existing work in ethical computing, including theoretical and empirical contributions to our understanding of how ethical concerns can be integrated into the development of modern technology. Based on work that has been done, we described a number of open research problems and challenges to integrating ethical concerns into the process of developing, evaluating, and maintaining data-driven software. To improve the state-of-practice, future work in the area of ethical software development practices should consider empirical evaluations to validate existing interventions as well as ways to develop interventions that go beyond fairness concerns and can be incorporated into the various stages of the data-driven software development pipeline.

References

[1] D. Mattioli, "On Orbitz, Mac users steered to pricier hotels," *The Wall Street Journal*, vol. August 23, 2012, wsj.com/articles/SB10001424052702304458604577488822667325882.

[2] M. Raghavan, S. Barocas, J. Kleinberg, and K. Levy, "Mitigating bias in algorithmic hiring: Evaluating claims and practices," *CoRR*, vol. 1906.09208, 2019. [Online]. Available: arxiv.org/abs/1906.09208

[3] A. Rajkomar, J. Dean, and I. Kohane, "Machine learning in medicine," *New England Journal of Medicine*, vol. 380, no. 14, pp. 1347–1358, 2019.

[4] E. E. Joh, "Feeding the machine: Policing, crime data, & algorithms," *Wm. & Mary Bill Rts. J.*, vol. 26, p. 287, 2017.

[5] Z. Wan, X. Xia, D. Lo, and G. C. Murphy, "How does machine learning change software development practices?" *Transactions on Software Engineering*, 2019.

[6] S. Amershi, A. Begel, C. Bird, R. DeLine, H. Gall, E. Kamar, N. Nagappan, B. Nushi, and T. Zimmermann, "Software engineering for machine learning: A case study," in *International Conference on Software Engineering: SEIP*. IEEE, 2019, pp. 291–300.

[7] H. Liu, S. Eksmo, J. Risberg, and R. Hebig, "Emerging and changing tasks in the development process for machine learning systems," in *International Conference on Software and System Processes*, 2020, pp. 125–134.

[8] A. D. Carleton, E. Harper, T. Menzies, T. Xie, S. Eldh, and M. R. Lyu, "The ai effect: Working at the intersection of ai and se," *IEEE Software*, vol. 37, no. 4, pp. 26–35, 2020.

[9] J. Angwin, J. Larson, S. Mattu, and L. Kirchner, "Machine bias," *ProPublica*, vol. May 23, 2016, propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing.

[10] K. Hill, "Wrongfully accused by an algorithm," *The New York Times*, vol. August 3, 2020, nytimes.com/2020/06/24/technology/facial-recognition-arrest.html.

[11] H. Ledford, "Millions of black people affected by racial bias in health-care algorithms," *Nature*, vol. 574, no. 7780, pp. 608–610, 2019.

[12] N. Martin, "Are ai hiring programs eliminating bias or making it worse?" *Forbes*, vol. December 13, 2018, forbes.com/sites/nicolemartin1/2018/12/13/are-ai-hiring-programs-eliminating-bias-or-making-it-worse.

[13] T. Mohn, "Self-driving cars may not be the game changer for safety we think," *Forbes*, vol. June 10, 2020, forbes.com/sites/tanyamohn/2020/06/10/self-driving-cars-may-not-be-the-game-changer-for-safety-we-think.

[14] G. Caldarelli, R. De Nicola, F. Del Vigna, M. Petrocchi, and F. Saracco, "The role of bot squads in the political propaganda on twitter," *Communications Physics*, vol. 3, no. 1, pp. 1–15, 2020.

[15] E. Ferrara, H. Chang, E. Chen, G. Muric, and J. Patel, "Characterizing social media manipulation in the 2020 us presidential election," *First Monday*, 2020.

[16] L. Zhang, J.-H. Tian, J. Jiang, Y.-J. Liu, M.-Y. Pu, and T. Yue, "Empirical research in software engineering—a literature survey," *Journal of Computer Science and Technology*, vol. 33, no. 5, pp. 876–899, 2018.

[17] P. Devanbu, T. Zimmermann, and C. Bird, "Belief & evidence in empirical software engineering," in *International Conference on Software Engineering*. IEEE, 2016, pp. 108–119.

[18] J. Morley, L. Floridi, L. Kinsey, and A. Elhalal, "From what to how: an initial review of publicly available ai ethics tools, methods and research to translate principles into practices," *Science and engineering ethics*, vol. 26, no. 4, pp. 2141–2168, 2020.

[19] H. T. Tavani, *Ethics and technology: Controversies, questions, and strategies for ethical computing*. John Wiley & Sons, 2011.

[20] B. C. Stahl, J. Timmermans, and B. D. Mittelstadt, "The ethics of computing: A survey of the computing-oriented literature," *ACM Computing Surveys*, vol. 48, no. 4, pp. 1–38, 2016.

[21] M. Stoica, M. Mircea, and B. Ghilic-Micu, "Software development: Agile vs. traditional." *Informatica Economica*, vol. 17, no. 4, 2013.

[22] I. Ozkaya, "Ethics is a software design concern," *IEEE Software*, vol. 36, no. 3, pp. 4–8, 2019.

[23] A. McNamara, J. Smith, and E. Murphy-Hill, "Does acm's code of ethics change ethical decision making in software development?" in *Foundations of Software Engineering*, 2018, pp. 729–733.

[24] J. L. Leidner and V. Plachouras, "Ethical by design: Ethics best practices for natural language processing," in *ACL Workshop on Ethics in Natural Language Processing*, 2017, pp. 30–40.

[25] D. S. Char, M. D. Abràmoff, and C. Feudtner, "Identifying ethical considerations for machine learning healthcare applications," *The American Journal of Bioethics*, vol. 20, no. 11, pp. 7–17, 2020.

[26] V. Vakkuri, K.-K. Kemell, and P. Abrahamsson, "Ai ethics in industry: a research framework," *arXiv:1910.12695*, 2019.

[27] F. B. Aydemir and F. Dalpiaz, "A roadmap for ethics-aware software engineering," in *Software Fairness (FairWare)*. IEEE, 2018, pp. 15–21.

[28] M. Mulvenna, J. Boger, and R. Bond, "Ethical by design: A manifesto," in *European Conference on Cognitive Ergonomics*, 2017, pp. 51–54.

[29] B. Shneiderman, "Bridging the gap between ethics and practice: Guidelines for reliable, safe, and trustworthy human-centered ai systems," *Interactive Intelligent Systems*, vol. 10, no. 4, pp. 1–31, 2020.

[30] P. S. Thomas, B. C. da Silva, A. G. Barto, S. Giguere, Y. Brun, and E. Brunskill, "Preventing undesirable behavior of intelligent machines," *Science*, vol. 366, no. 6468, pp. 999–1004, 2019.

[31] J. Chakraborty, S. Majumder, Z. Yu, and T. Menzies, "Fairway: A way to build fair ML software," in *Foundations of Software Engineering*, 2020.

[32] J. A. Adebayo *et al.*, "Fairml: Toolbox for diagnosing bias in predictive modeling," Ph.D. dissertation, Massachusetts Institute of Technology, 2016.

[33] R. K. Bellamy, K. Dey, M. Hind, S. C. Hoffman, S. Houde, K. Kannan, P. Lohia, J. Martino, S. Mehta, A. Mojsilovic *et al.*, "Ai fairness 360: An extensible toolkit for detecting, understanding, and mitigating unwanted algorithmic bias," *arXiv:1810.01943*, 2018.

[34] P. Saleiro, B. Kuester, L. Hinkson, J. London, A. Stevens, A. Anisfeld, K. T. Rodolfa, and R. Ghani, "Aequitas: A bias and fairness audit toolkit," *arXiv:1811.05577*, 2018.

[35] R. Angell, B. Johnson, Y. Brun, and A. Meliou, "Themis: Automatically testing software for discrimination," in *Foundations of Software Engineering*, 2018, pp. 871–875.

[36] Á. A. Cabrera, W. Epperson, F. Hohman, M. Kahng, J. Morgenstern, and D. H. Chau, "Fairvis: Visual analytics for discovering intersectional bias in machine learning," in *Conference on Visual Analytics Science and Technology*, 2019, pp. 46–56.

[37] "scikit-learn: Machine learning in Python," scikit-learn.org/stable/.

[38] S. Schelter, Y. He, J. Khilnani, and J. Stoyanovich, "Fairprep: Promoting data to a first-class citizen in studies on fairness-enhancing interventions," *arXiv:1911.12587*, 2019.

[39] S. Vasudevan and K. Kenthapadi, "Lift: A scalable framework for measuring fairness in ml applications," in *International Conference on Information & Knowledge Management*, 2020, pp. 2773–2780.

[40] S. Bird, M. Dudík, R. Edgar, B. Horn, R. Lutz, V. Milan, M. Sameki, H. Wallach, and K. Walker, "Fairlearn: A toolkit for assessing and improving fairness in ai," MSR-TR-2020-32, Tech. Rep.

[41] K. Sokol, A. Hepburn, R. Poyiadzi, M. Clifford, R. Santos-Rodriguez, and P. Flach, "Fat forensics: A python toolbox for implementing and deploying fairness, accountability and transparency algorithms in predictive systems," *Journal of Open Source Software*, vol. 5, no. 49, p. 1904, 2020.

[42] V. Vakkuri, K.-K. Kemell, M. Jantunen, and P. Abrahamsson, ""this is just a prototype": How ethics are ignored in software startup-like environments," in *International Conference on Agile Software Development*. Springer, 2020, pp. 195–210.

[43] R. Hermon, "Being an ethical software engineer," *InfoQ*. [Online]. Available: infoq.com/articles/ethical-software-engineer

[44] J. Shapiro and R. Blackman, "Four steps for drafting an ethical data practices blueprint," *TechCrunch*. [Online]. Available: techcrunch.com/2020/07/24/four-steps-for-an-ethical-data-practices-blueprint

[45] H. Yu, Z. Shen, C. Miao, C. Leung, V. R. Lesser, and Q. Yang, "Building ethics into artificial intelligence," *arXiv:1812.02953*, 2018.

[46] C. Bannister, B. Schniderman, and N. Buckley, "Ethical tech: Making ethics a priority in today's digital organization," *Deloitte Review*, no. 27, 2020.

[47] A. F. Winfield and M. Jirotka, "Ethical governance is essential to building trust in robotics and artificial intelligence systems," *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 376, no. 2133, p. 20180085, 2018.

[48] A. Rashid, K. Moore, C. May-Chahal, and R. Chitchyan, "Managing emergent ethical concerns for software engineering in society," in *International Conference on Software Engineering*, vol. 2. IEEE, 2015, pp. 523–526.

[49] J. Borenstein and A. Howard, "Emerging challenges in ai and the need for ai ethics education," *AI and Ethics*, pp. 1–5, 2020.

[50] R. Bond, M. D. Mulvenna, H. Wan, D. D. Finlay, A. Wong, A. Koene, R. Brisk, J. Boger, and T. Adel, "Human centered artificial intelligence: Weaving ux into algorithmic decision making." in *RoCHI*, 2019, pp. 2–9.

[51] A. D. Selbst, D. Boyd, S. A. Friedler, S. Venkatasubramanian, and J. Vertesi, "Fairness and abstraction in sociotechnical systems," in *Fairness, Accountability, and Transparency*, 2019, pp. 59–68.

[52] A. Holzinger, "From machine learning to explainable ai," in *World Symposium on Digital Intelligence for Systems and Machines*. IEEE, 2018, pp. 55–66.

[53] C. Rudin, "Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead," *Nature Machine Intelligence*, vol. 1, no. 5, pp. 206–215, 2019.

[54] A. Kumar, T. Braud, S. Tarkoma, and P. Hui, "Trustworthy ai in the age of pervasive computing and big data," *arXiv:2002.05657*, 2020.

[55] A. Jacovi, A. Marasović, T. Miller, and Y. Goldberg, "Formalizing trust in artificial intelligence: Prerequisites, causes and goals of human trust in ai," *arXiv:2010.07487*, 2020.

[56] H. Chang, "Ethics in artificial intelligence: A disjoint between knowing and acting," *Journal of Data Protection & Privacy*, vol. 3, no. 3, pp. 244–249, 2020.

[57] B. Mittelstadt, "Principles alone cannot guarantee ethical ai," *Nature Machine Intelligence*, pp. 1–7, 2019.

[58] A. Tsamados, N. Aggarwal, J. Cowls, J. Morley, H. Roberts, M. Taddeo, and L. Floridi, "The ethics of algorithms: key problems and solutions," *Available at SSRN 3662302*, 2020.

[59] S. Amershi, D. Weld, M. Vorvoreanu, A. Fourney, B. Nushi, P. Collisson, J. Suh, S. Iqbal, P. N. Bennett, K. Inkpen *et al.*, "Guidelines for human-ai interaction," in *Conference on Human Factors in Computing Systems*, 2019, pp. 1–13.

[60] M. S. A. Lee and J. Singh, "The landscape and gaps in open source fairness toolkits," *SSRN*, 2020.

[61] V. Vakkuri, K.-K. Kemell, and P. Abrahamsson, "Ethically aligned design: an empirical evaluation of the resolvedd-strategy in software and systems development context," in *Conference on Software Engineering and Advanced Applications*. IEEE, 2019, pp. 46–50.

[62] M. A. Madaio, L. Stark, J. Wortman Vaughan, and H. Wallach, "Co-designing checklists to understand organizational challenges and opportunities around fairness in ai," in *Conference on Human Factors in Computing Systems*, 2020, pp. 1–14.

[63] V. Vakkuri, K.-K. Kemell, and P. Abrahamsson, "Implementing ethics in ai: Initial results of an industrial multiple case study," in *International Conference on Product-Focused Software Process Improvement*. Springer, 2019, pp. 331–338.

[64] V. Vakkuri, K.-K. Kemell, J. Kultanen, M. Siponen, and P. Abrahamsson, "Ethically aligned design of autonomous systems: Industry viewpoint and an empirical study," *arXiv:1906.07946*, 2019.

[65] V. Vakkuri, K.-K. Kemell, J. Kultanen, and P. Abrahamsson, "The current state of industrial practice in artificial intelligence ethics," *IEEE Software*, 2020.

[66] X. Zhang, Y. Yang, Y. Feng, and Z. Chen, "Software engineering practice in the development of deep learning applications," *arXiv:1910.03156*, 2019.

[67] E. de Souza Nascimento, I. Ahmed, E. Oliveira, M. P. Palheta, I. Steinmacher, and T. Conte, "Understanding development process of machine learning systems: Challenges and solutions," in *International Symposium on Empirical Software Engineering and Measurement*. IEEE, 2019, pp. 1–6.

[68] E. Klotins, M. Unterkalmsteiner, and T. Gorschek, "Software engineering in start-up companies: An analysis of 88 experience reports," *Empirical Software Engineering*, vol. 24, no. 1, pp. 68–102, 2019.

[69] M. Unterkalmsteiner, P. Abrahamsson, X. Wang, A. Nguyen-Duc, S. Shah, S. S. Bajwa, G. H. Baltes, K. Conboy, E. Cullina, D. Dennehy *et al.*, "Software startups–a research agenda," *e-Informatica Software Engineering Journal*, vol. 10, no. 1, 2016.

[70] B. Vasilescu, V. Filkov, and A. Serebrenik, "Stackoverflow and github: Associations between software development and crowdsourced knowledge," in *International Conference on Social Computing*. IEEE, 2013, pp. 188–195.

[71] Z. Iskoujina and J. Roberts, "Knowledge sharing in open source software communities: motivations and management," *Journal of Knowledge Management*, 2015.

[72] S.-F. Wen, "Learning secure programming in open source software communities: a socio-technical view," in *International Conference on Information and Education Technology*, 2018, pp. 25–32.

[73] T. Bruckhaus, N. Madhavii, I. Janssen, and J. Henshaw, "The impact of tools on software productivity," *IEEE Software*, vol. 13, no. 5, pp. 29–38, 1996.

[74] D. L. Atkins, T. Ball, T. L. Graves, and A. Mockus, "Using version control data to evaluate the impact of software tools: A case study of the version editor," *Transactions on Software Engineering*, vol. 28, no. 7, pp. 625–637, 2002.