2020

On Using Model For Downstream Responsibility

Frances S. Grodzinsky

Marty J. Wolf

Keith W. Miller

ON USING A MODEL FOR DOWNSTREAM RESPONSIBILITY

Frances S. Grodzinsky, Marty J. Wolf, Keith W. Miller

Sacred Heart Univ. (USA), Bemidji State Univ. (USA), Univ. of Missouri – St. Louis (USA)
grodzinskyf@yahoo.com; mjwolf@bemidjistate.edu; millerkei@umsl.edu

EXTENDED ABSTRACT

In "On the Responsibility for Uses of Downstream Software," (Wolf, Miller, and Grodzinsky 2019), the authors identify features of software and the software development process that may contribute to the differences in the level of responsibility assigned to the software developers when they make their software available for others to use as a tool in building a second piece of software. They call this second use of the software "downstream use." The features they identified that impact assigning responsibility to the developer of the original software for the social and ethical impacts of the downstream use include closeness to the hardware, risk, sensitivity of data, degree of control over or knowledge of the future population of users, and the nature of the software (general vs. special purpose). Close analysis of these features led the authors to develop two different analysis models that might be used to assign responsibility in the use of downstream software: the Fixed History Model and the Chained History Model.

In the Fixed History Model there is an assumption that within the system of events that led to an ethical breach, there are certain inputs that are immune to the assignment of any moral responsibility. This model does not consider assigning any portion of the Distributed Moral Responsibility (DMR) to those who produced the inputs. The Fixed History Model is appropriate for certain types of software. For example, the developers of database software are rarely considered for the assignment of moral responsibility in the event of a breach. Typically, in such a case, responsibility stops at the database implementers. The Chained History Model, however, applies in cases where one of the inputs to the system is a piece of software and the attribution of moral responsibility propagates back to the developers of that software.

In this paper, we will review a selection of recent papers (2017-2019) on the attribution of responsibility in emerging technologies. Our analysis will determine situations when responsibility attribution for a moral action could have been clarified by using either the Fixed History Model or the Chained History Mode. We will demonstrate how applying these models might help clarify ethical issues associated with distributed responsibility for software developers in some complex and interesting cases. Our analysis will show both the usefulness of and some of the shortcomings of the models proposed by Wolf, Miller, and Grodzinsky. From the analysis of the papers, we will make suggestions for a revision of the models that might be more robust when applied to cases on responsibility for downstream uses of software.

As an example, we have considered the paper "Digital health fiduciaries: protecting user privacy when sharing health data" where Chirag Arora (2019) argues that when it comes to privacy concerns surrounding health data, it is the responsibility of the digital health data controllers to take steps to protect the privacy of those whose data is being collected and stored. Arora argues for a fiduciary relationship between data subjects and the data controller. Arora's argument uses "security, anonymization, and data minimization as examples of contextualization and flexibility required to deal with privacy issues." Even though Arora brings up the WannaCry ransomware attack on the UK's National Health Service, the responsibility for the system failure is not mapped back to flaws in the

software that was infected by WannaCry, but rather to the failure to upgrade. Arora places the ethical breach at the feet of the data controller and makes no attempt to push any responsibility back to those who created the software with the flaw in it. In terms of our two models, Arora uses the Fixed History Model.

This analysis stands in contrast to the argument presented by Wolf et al. They argue that "the more sensitive the data accessed [are], the more responsibility [that] can be ascribed to the developer for [the software's] downstream use." Using this line of reasoning, Wolf et al. would likely use the Chained History Model in this case. Our analysis will compare and contrast these two opposing views.

As a second example, in the article" First steps towards an ethics of robots and artificial intelligence (RAI)," John Tasioulas (2019) investigates the problem of trying to build moral norms into RAIs. He distinguishes between RAIs that follow top down algorithms that are prescriptive and closed-rule and bottom up or stochastic algorithms that use machine learning. In the first case, the RAI is largely functional and failure to accomplish its task can be attributed back to the developer. "In .... RAIs operating on the basis of top-down algorithms that render their behavior highly predictable, the argument for attributing legal responsibility to manufacturers, owner, or users seems compelling (Tasioulas, 2019:70). Responsibility analysis can be served by the Fixed Model. In RAIs that use machine learning, the cases are more varied and complex. The author asks "...whether a good case exists for attributing legal personality to RAIs with corresponding legal rights and responsibilities ..." (Tasioulas, 2019:70). The European Union and UNESCO have been grappling with this issue and it is beyond the scope of our paper to delineate the various arguments. However, Tasioulas does raise the question of traceability as particularly difficult with RAIs using bottom-up algorithms. He asks "... how do we ensure the 'traceability' of RAIs in order to be able to assign moral or legal responsibility in relation to them? Traceability involves being able to determine the causes that led an RAI to behave in the way that it did..." (Tasioulas, 2019:71). Applying the Chained Model might aid in the analysis of what Tasioulas calls one of the biggest challenges in the deployment of machine learning RAIs.

As these two examples show, the two models for downstream responsibility attribution can clarify thinking about different kinds of software. In the full paper, we will show where and why each of the two models has been used and argue whether doing do was appropriate. Additionally, we will identify cases where the choice between the models is not clear. We will also identify revisions to the models to make them more versatile.

KEYWORDS: responsibility, software developer responsibility, models of responsibility, ethical analysis.

REFERENCES

Arora, C. (2019). Digital health fiduciaries: protecting user privacy when sharing health data, Ethics Inf Technol 21: 181. DOI: 10.1007/s10676-019-09499-x.

Tasioulas, J. (2019). First steps towards an ethics of robots and artificial intelligence, Journal of Practical Ethics. 7(1), 49-83.

Wolf, M. J., Miller, K. W., & Grodzinsky, F. S. (2019). On the responsibility for uses of downstream software," Computer Ethics - Philosophical Enquiry (CEPE) Proceedings. *2019*, Article 3. DOI: 10.25884/7576-wd27, from https://digitalcommons.odu.edu/cepe_proceedings/vol2019/iss1/3.