BLOG @CACM

The *Communications* Web site, <http://cacm.acm.org>, features more than a dozen bloggers in the BLOG@CACM community. In each issue of *Communications*, we'll publish selected posts or excerpts.

Follow us on Twitter at <http://twitter.com/blogCACM>

DOI:10.1145/3231166

<http://cacm.acm.org/blogs/blog-cacm>

Assessing Responsibility for Program Output

We lack an easy way to indicate that algorithms do not make decisions and are not biased; programmers do, and are.

Robin K. Hill
Articulation of Decision Responsibility
<http://bit.ly/2kDNgzY>
May 21, 2018

Remember the days when record-keeping trouble, such as an enormous and clearly erroneous bill for property taxes, was attributed to “computer error?” Our technological society fumbles the assignment of responsibility for program output. It can be seen easily in exaggerations like this, from a tech news digest: “Google’s Artificial Intelligence (AI) has learned how to navigate like a human being.” Oh, my. See the *Nature* article by the Google researchers² for the accurate, cautious, description and assessment. The quote given cites an article in *Fast Company*, which states that “AI has spontaneously learned how to navigate to different places.”⁴ Oh, dear.

But this is not the root of the problem. In the mass media, even on National Public Radio, I hear leads for stories about “machines that make

biased decisions.” Exaggeration has been overtaken by simple inaccuracy. We professionals in Tech often let this pass, apparently on the belief the public really understands machines and algorithms have no such capacity as is normally connoted by the term “decision”; we think the speakers are uttering our own trade shorthand. When we say “the COMPAS system decides that offender B is more likely to commit another crime than is offender D”¹ (paraphrase mine), it is short for “the factors selected, quantified, and prioritized in advance by the staff of the software company Northpointe assign a higher numeric risk to offender B than to offender D.” When the *Motley Fool* website⁶ says “computers have been responsible for a handful of ‘flash crashes’ in the stock market since 2010,” it means that “reliance on programs that instantaneously implement someone’s predetermined thresholds for stock sale and purchase has been responsible ... etc.”

The trouble is that there is no handy way to say these things. The paraphras-

es here expose the human judgments that control the algorithms, but the paraphrases are unwieldy. For decades of software engineering, we have adopted slang that attributes volition and affect to programs. Observations can be found on Eric S. Raymond’s page on anthropomorphization⁵. I doubt many hackers ascribe the intentional stance to programs; I suspect rather that programmers use these locutions for expedience, as the “convenient fictions that permit ‘business as usual’.”³ But the public misunderstanding is literal, and serious.

Algorithms are not biased, because a program does not make decisions. The program implements decisions made elsewhere. Programs are made up of assignments of value, evaluations of expressions, and branching to addresses for loading of instructions. There is no point of unpredictable choice, that is, a choice not determined by the code (even for “random” number generation), if we rule out quantum computation, which I am not qualified to consider. Certain scenarios may appear to

challenge this bald determinism. Let's scrutinize those briefly.

Deductive closure includes propositions not immediately obvious.

But even where the programmers are not sure what exactly will happen, because of obscure compound conditions, the algorithm does not “make a decision.” What happens is an implication of the assertions in force (written into the code if the programmer bothered to formulate assertions), that is, an implication of the deductive closure. The question whether programmers can be held responsible for the distant eventualities is significant, noting that what we view as algorithmic bias does not often seem deliberate. In any case, the deciding agent is certainly not the machine.

Timing of interactions may result in unanticipated outcomes, as in passive investment through computerized stock trading.

But unexpected states do not demonstrate demonic agency. Someone has decided in advance that it makes sense to sell a stock when it loses $n\%$ of its value. That's not what we would call a real-time decision on the spot, because it ignores (1) the real time and (2) the spot. We would correctly call that a decision made earlier and elsewhere by system designers, which played out into unforeseen results.

The pattern-matching of deep learning precludes the identification of symbolic variables and conditions.

With no semantics available, no agent prominent, and no execution through a conditional structure traceable, the computer looks like the proximate decider. But no. If there are training cases, some complex combination of numeric variables has developed from

given initial values which were adjusted over time to match a set of inputs with a set of outputs, where those matches were selected by the systems designers. In unsupervised learning, some sort of regularities are uncovered, regularities that were already there in the data. Although it may be tempting to say that no one is deciding anything, certainly no computer is making anything that could be called a decision. Someone has planned antecedently to seek those regularities.

Selection, recommender, and classification systems use the criteria implemented in their decision structure. We in the trade all know that whatever the algorithmic technique, the computer is not deciding. To explain to the public that computers are dumb may baffle and frustrate, rather than educate. The malapropisms that grant agency to algorithms confuse the determination of responsibility and liability, but also the public grasp of Tech overall. People may attempt to “persuade” the computer, or to try to fix, enhance, or “tame” the programs, rather than just rejecting their inappropriate deployment. At the extreme, people feel helpless and fearful when danger comes from beings like us—willful, arbitrary, capricious—except more powerful. Worse yet would be apathy: Society may ignore the difficulties and become resigned to the results, as if such programmed assessments were factive.

What would be the correct locution, the correct way to say it, passive toward machine and active toward programmer (or designer or developer or specification writer or whomever)? How should we note that “the deductive closure of home mortgage qualification criteria entails red-lining of certain neighborhoods”—other than to say those exact words, which are not compelling? How should we say that “The repeated adjustment of weighting criteria applied to a multi-dimensional function of anonymous variables, closely approximating an unknown function for which some correct outcomes have been identified by past users, associates

this individual record to your own discrete declared criteria for a date”—without saying “the dating app has chosen this match for you”?

We have no other way of expressing such outcomes easily. We lack the verbs for computing that denote reaching states that look like decisions, and taking actions that look like choices. We need a substitute for “decides” in “the algorithm decides that X,” something to fill in the blank in “the program _____ X.” Perhaps “the program fulfills X.” Perhaps “the program derives that X.” Well ... this seems lame. The trouble really is that we have to avoid any verb that implies active mental function. This is new. This is unique to computing, as far as I can tell. The Industrial Revolution brought us many machines that seemed to have human capacities, but they also had material descriptions. For mechanical devices, verbs are available that describe physical functionality without the implication of cognition: “The wheel wobbles.” “The fuel line clogged.” We may say, jokingly or naively, that “the car chooses not to start today,” but we are not forced into it by lack of vocabulary.

For this new technological requirement, the best locution I can come up with is, “the result of the programmed assumptions is that X.” I have not heard anyone seriously appeal to “computer error” as a final explanation for some time; that seems like progress in understanding Tech. If we can forgo that locution, maybe we can forgo “biased algorithms.”

Any other ideas?

References

1. Angwin, J., et al. Machine Bias. *ProPublica*. May 23, 2016, <http://bit.ly/2sGcEbH>.
2. Banino, A., et al. 2018. *Nature* 557; pages 429–433. Vector-based navigation using grid-like representations in artificial agents. doi:10.1038/s41586-018-0102-6.
3. Caporael, L.R. 1986. Anthropomorphism and Mechanomorphism: Two Faces of the Human Machine. *Computers in Human Behavior* 2; pages 215–234. [https://doi.org/10.1016/0747-5632\(86\)90004-X](https://doi.org/10.1016/0747-5632(86)90004-X).
4. Grothaus, M. Google's AI is learning to navigate like humans, *Fast Company*, May 15, 2018, <http://bit.ly/2JuyYzI>.
5. Raymond, E. Anthropomorphization. From *The Jargon File*, <http://bit.ly/2kPmF2Z>.
6. Williams, S. (TMFUltraLong). The Evolution of Stock Market Volatility, *The Motley Fool*, Apr 3, 2018, <http://bit.ly/2xKrROY>.

Robin K. Hill is adjunct professor in the Department of Philosophy, and in the Wyoming Institute for Humanities Research, of the University of Wyoming. She has been a member of ACM since 1978.