Fuzzy Ethics: Or How I Learned to Stop Worrying and Love the Bot

Michael James Heron
Robert Gordon University
Aberdeen
Scotland
m.j.heron1@rgu.ac.uk

Pauline Belford
Dundee and Angus College
Angus
Scotland
p.belford@dundeeandangus.ac.uk

ABSTRACT

The recent death of a Volkswagen worker at the hand of a factory robot has resulted in a number of editorials and opinion pieces discussing moral responsibility and robots. In this short response piece we outline some of the wider context of this discussion, with reference to the classic ethical study the Case of the Killer Robot. We argue that there is a growing need for the field of computer ethics to consider with some urgency what it means to be a responsible moral agent when tragic events occur, and to what extent it makes sense to 'blame the robot'.

Categories and Subject Descriptors

K.7.4 [**Professional Ethics**]: Codes of ethics; Codes of good practice; Ethical dilemmas.

General Terms

Security; Human Factors; Legal Aspects

Keywords

Ethics; Morality; Professional Issues; Killer Robot; Case of the Killer Robot; Volkswagen

1. INTRODUCTION

The question of who we should blame when a robot kills a human has recently become somewhat more pressing. The recent death of a Volkswagen employee at the hand of an industrial factory robot [1] has left ethicists and legislators unsure of where the moral and ethical responsibility for the death should lie – does it lie with the owners, the developers, the factory managers, or elsewhere? These are not easy questions – for many years, the authors of this paper have used the Case of the Killer Robot study [3] to explore issues of moral and ethical responsibility within classes on computer ethics and professionalism. The Killer Robot case study begins with the CX30 robot malfunctioning and killing its operator. It then progresses through the use of fictionalized newspaper articles. Each of these successively unpick and expand upon the facts as we know them to help illustrate the complex interrelated issues of responsibility in collaborative software development. As a teaching scenario, it is now showing its age, which resulted in the authors publishing their own ethical case study [7][8] as a spiritual successor. However, the issues that the Case of the Killer Robot raises are highly relevant to the recent unfortunate events in Baunatal, Germany.

Details about the incident are limited at the time of writing as an investigation is still ongoing. We know that the twenty-two year old victim was part of the team setting up a stationary robot at the factory. It activated, grabbed him, and crushed him against one of the metal plates that formed part of its rig. Volkswagen claim human error – the robot was functioning as expected, it was just that it shouldn't have been active while anyone was within its safety rig. However, until such a conclusion is delivered from a party not directly involved in the tragic events, we would like to take the opportunity to discuss the issue from the perspective of a malfunctioning piece of automated hardware. Within this paper we will use the term 'robot' and 'automation' largely interchangeably, representing an acknowledgement of the fact that 'robots' often do not take on the forms that we might expect from popular literature. The ethical issues are the same, in our view, whether we are talking about software or physical hardware.

2. Where Does Responsibility Lie?

Failures in industrial software engineering projects are multi-faceted and rarely can we point to a single individual in a large team as the sole originator of faulty programming or iffy hardware. Software is not just the product of the developers - it is also a product of wider societal norms, management paradigms, and cultural expectations. Software development is also hugely collaborative, and builds upon the work of others through layers of architectural abstraction – toolkits; frameworks; virtual machines; programming languages; and operating contexts. The executing code of a programmer is usually mediated through many software and hardware modules before it is eventually enacted upon by the underlying systems. When software is embedded within hardware, such as is often the case with robots, there may be many fewer layers. This doesn't greatly simplify the task of assigning responsibility for malfunctions. The context of software development is complex, and while fewer layers mean fewer mediations and abstractions, we can rarely point to a single code point and say 'That's the culprit'.

When a robot malfunctions and grabs an employee, which part of the software systems malfunctioned? One can realistically place the blame at almost any layer – adherents of the philosophy of **defensive programming** would argue that 'all data is tainted unless proven otherwise'. Every step of the system should be re-evaluating the instructions it was given to ensure that they make sense in context. It could be argued that it was the blame of the quality assurance department, as they should have caught errors before they ever made it into production code. It could be argued that it was a management issue, because management put in place the protocols through which the software could be marked off as complete. You could argue it was the fault of the factory owners for accepting delivery of a machine they had not tested for safety. You could argue a meaningful role for almost anyone – indeed, that is the real lesson that can be learned from the Case of the Killer Robot. However, such blame games are unhelpful and in many respects just obscure the core issue – that we don't have a meaningful framework within which we can assess collaborative responsibility in cutting edge software and hardware engineering. Often, we must simply conclude that nobody is to blame because everybody is to blame. That might be the truth, but it's a very unsatisfying truth. It offers no catharsis, and affords no closure.

What's more interesting perhaps in the Volkswagen incident is that we seem now to be willing to accept that the robot itself may have to shoulder some of the blame. When we teach the Case of the Killer Robot to our students, perhaps the most notable thing is that nobody ever considers the robot to be the real villain in the piece. I have heard students put a case for why the plagiarizing programmer or the bullying bosses or the browbeaten tester should bear the largest brunt of responsibility. Nobody has ever asked 'why is the robot getting a free pass?' We teach the case study with computing students so a certain degree of understanding of how hardware and software interacts likely has an impact on this moral judgement. For others though, where the underlying relationship between developers and software and hardware are obscured in essential unknowability, should the robot actually be a valid target for judgement?

3. Unknowable Machines

More and more, we're willing to accept that our computing and hardware devices should take some responsibility for their own wellbeing. We have software packages that can mend their own installations, and antivirus software that patrols our systems, often with minimal input from the operator. We have operating systems that keep up to date, invisibly re-engineering themselves as we use them.

Once upon a time when you pressed the 'eject' button on a video player, it obeyed instantly – video cassettes would be spat out in an instant, sometimes trailing ribbons of unwound tape behind them. Now, you're as likely to see a little hourglass wheeling as the device 'thinks' about whether or not to obey our instruction. We've gone from issuing commands to making suggestions, and we are sometimes over-ruled when the software decides that we need saved from ourselves. When I tell my Macbook to shut down, it usually tells me that it won't until I go and manually shut down all my running programs. The OS in such cases decides that it knows best. Perhaps it does, but in such circumstances we have to consider whether we as the users are actually in control. If we are not in control, then we need to consider how much responsibility we bear for our actions.

This moves the argument for ethical responsibility onto the developers, but this too is increasingly an area where it is the software that makes the decisions. Some software is now so complex that developers cannot say with any real confidence how it makes decisions. Advanced neural nets make so many connections, at such speed, and using such vast data-sets that no-one can be entirely sure how they arrive at their conclusions. Google's 'deep learning' machines are now so advanced that they sometimes outwit their own programmers [9]. The likely outcome of this is that such deep learning machines may need to be maintained by other, specialist, deep learning machines. This means fewer experts writing and developing the tools, and more layers of abstraction between their work and the eventual output of the systems. In other words, we are losing the ability at the bleeding edge of development to meaningfully understand why our software does what it does.

These trends may be alarming, but we must also consider the benefits that come from such automation. Google's self-driving cars, for example, have been involved in numerous collisions. The evidence though suggests that it has never been through an error on the part of the self-driving algorithms - it's always been 'other drivers' [5]. Whether this is true or not, we must accept the possibility that automation, when done well, simply makes fewer mistakes than humans in the same situations. Under the limited circumstances under which an AI platform may thrive, they think faster, think more broadly, and think more reliably. One paper [2] breaks down legal judgements to determine extraneous factors that might influence rulings – proximity to a lunch break, amongst other things, is a genuine influencing factor on the severity of a the sentence that a judge hands down. We overestimate, in many cases, our own rationality. Likewise we underestimate the degree to which factors over which we are not fully in control may influence our decisions.

On the other hand, barring a few vanishingly small incidents, for example the notorious Pentium FDIV bug [12], Computers do not make mistakes. At least, they do not make them within the parameters of the designed hardware that we provide them. It comes down again to the fallibility of software and hardware developers. Increasingly that too is becoming an unsatisfying answer that lacks closure and catharsis.

Ultimately, everything that a computer, or a robot, does ends back at the code a software developer writes: that would seem like a sensible termination point for where moral responsibility lies. We too though are slaves to our own genetic and neural programming[1] and yet we are an obvious unit of moral responsibility. We cannot simply argue that what we do is an inevitable consequence of our evolutionary firmware. That does not expunge us of the weight of immorality. We bear the ultimate responsibility for what we do – why not computers?

These issues are not simple to untangle. The growing importance of automation and robotics to modern society puts pressure on us to come up with at least some kind of framework within which we can properly evaluate the moral responsibility of software and hardware development. A recent letter, signed by Stephen Hawking, Elon Musk and Steve Wozniak amongst many others[2] argued for a ban on autonomous artificial intelligence in offensive warfare. It will be technically feasible in the next few years for military drones to be deployed without the moderating hand of a human at the kill-switch. The authors of this letter have argued in the strongest terms that we should never allow this to happen – that while AI can be used effectively in defensive systems, to allow its use in offensive roles is to spark off the next great arms race. This is a valid concern, and one which we share. However, we must also be mindful of the fact that drones are precise only when the information that is fed to their operators is similarly precise. With a human 'moral agent' at the kill-switch the death count that comes from drone warfare is still alarming. The human rights group *Reprieve* issued an analysis which suggested that from a targeting pool of 41 people, the US drone programme resulted in a death toll of 1,147 [13] – for every target, 28 bystanders are killed. The Hellfire missiles that rain down from Predator or Reaper drones are no respecter of precision, and so far we cannot say that having a human pull the trigger has led to

[1] And perhaps even unwilling to make any changes in what is an inherently deterministic universe, but let's not go down that particular rabbit hole.

[2] The text and list of signatories for this letter may be found here: http://futureoflife.org/AI/open_letter_autonomous_weapons

inspiring results. Drone operators work within formal systems of diffused responsibility. They by themselves cannot bear the moral burden of such deaths in the same way that we cannot single out individual programmers within a development team. It is hard though to see how much worse it could get with autonomous AI at the helm. Critics may argue that AI cannot feel sympathy, empathy or remorse – but neither can it feel anger, impatience or hate. The mistakes that automated AI might make are software errors, and those can be fixed – over time, a software system will tend towards (although probably never reach) zero defects. The question in such cases is 'what cost are we willing to accept for iterative improvements?' In making that decision, we should not undervalue the fact that improvements **can** be made.

In the early days of automated computer intelligence, we were at least partially saved from an escalating nuclear exchange by the cool head and moral compass of Colonel Stanislav Petrov [4], who interpreted incoming nuclear missile telemetry correctly as a malfunction in the Soviet early warning system. Had this been entirely automated, we might well not be in a position now to debate the ethical and moral responsibility of robots. However, we have come a long way since then and we need not be inherently fearful of the impact of automation. That is not to say that we shouldn't be wary. Leaving aside the issues of technical correctness we don't have the tools we need to meaningfully address the ethical dilemmas that arise from deaths that result from automation or faulty programming. It is often the case that technology outstrips our philosophy and legislation, and this is an area in which the gulf between 'what we can do' and 'how we understand what we do' is very wide.

4. Conclusion

The death of the Volkswagen worker was tragic, but it must be viewed in context – in terms of significant robot related deaths, we have this and Therac-25 [11] as the major headline cases. In the UK alone, there were an estimated 142 fatal workplace injuries in 2014/2015, and 136 in 2013/14 [6] - far more in one country, in one year, than we can realistically attribute within the workplace to failures of robotics. When considered in context, the number of robot-related deaths is actually very small – perhaps even comfortingly so. We do more damage to ourselves, as a species, than robots ever could.

However, we do need to start the conversation properly as to what moral role we should assign robots when things go wrong. At the moment, our frameworks for having that discussion are not well equipped to deal with the logistics of distributed authority in software development. We certainly don't have an effective ethical architecture for assigning blame to semi-autonomous units that we have no ability to even punish for transgressions. What can we realistically do to punish a robot that is deemed to have behaved outsides the bounds of societal norms? We can punish the human web around it, but we cannot truly punish an entity that has no conscious awareness of its own self.

Perhaps what we need is a kind of fuzzy ethics to go with the often fuzzy logic that underpins complex learning systems. One that is capable of handling a multi-dimensional relational web of roles, and assessing moral responsibility through collapsing certain pathways of that web to create judgement perspectives based on what it is we're looking to decide. We don't have that yet – we don't even have anything close to it. Until we do, arguments over blame and responsibility in these kinds of circumstances will always be shallow, failing to cut to the heart of the matter.

In other words, we have a very long way to go before blaming automated systems is anything more than an irrational outcome of the anthropomorphizing of human frustrations.

5. Acknowledgements

This paper was an invited response to the article 'Who is to blame when a robot kills a human: An ethical dilemma for the 21st Century'. This article was first published in Religion Dispatched by Hilary Ledwell [10].

6. REFERENCES

[1] Associated Press (2015). Robot kills worker at Volkswagen plan in Germany. [Available online from: http://www.theguardian.com/world/2015/jul/02/robot-kills-worker-at-volkswagen-plant-in-germany]

[2] Danziger, S., Levav, J., & Avnaim-Pesso, L. (2011). Extraneous factors in judicial decisions. *Proceedings of the National Academy of Sciences*, *108*(17), 6889-6892

[3] Epstein, R. G. (1996). *The case of the killer robot: stories about the professional, ethical, and societal dimensions of computing*. John Wiley & Sons Inc.

[4] Forden, G., Podvig, P., & Postol, T. A. (2000). Colonel Petrov's good judgment. IEEE spectrum, March, 37, 3.

[5] Gurney, J. K. (2013). Sue my car not me: Products liability and accidents involving autonomous vehicles. U. Ill. JL Tech. & Pol'y, 247.

[6] Health and Safety Execute (2015). Statistics on Fatal Injuries in the Workplace in Great Britain in 2015. [Available from http://www.hse.gov.uk/statistics/pdf/fatalinjuries.pdf]

[7] Heron, M. J., & Belford, P. (2014). Ethics in context: a scandal in academia. *ACM SIGCAS Computers and Society*, *44*(2), 20-51.

[8] Heron, M. J., & Belford, P. (2015). Power and perception in the scandal in academia. *ACM SIGCAS Computers and Society*, *45*(2), 11-19.

[9] Le, Q (2015). Large Scale Deep Learning. Retrieved from [http://www.slideshare.net/SessionsEvents/quoc-le-slides-m-lconf]

[10] Ledwell, H (2015). Who is to blame when a robot kills a human: An ethical dilemma for the 21st Century'. [Available online from http://www.salon.com/2015/07/20/who_is_it_ethical_to_blame_when_a_robot_kills_a_human_partner/]

[11] Leveson, N. G., & Turner, C. S. (1993). An investigation of the Therac-25 accidents. Computer, 26(7), 18-41.

[12] Price, D. (1995). Pentium FDIV flaw-lessons learned. Micro, IEEE, 15(2), 86-88

[13] Reprieve (2014). Us Drone Strikes Kill 28 Unknown People For Every Intended Target. [Available online at http://www.reprieve.org.uk/press/2014_11_25_us_drone_strikes_kill_28_each_target/]