



THE UNIVERSITY OF
MELBOURNE

Published on *Up Close* (<https://upclose.unimelb.edu.au>)

#363: The social life of algorithms: Shaping, and being shaped by, our world

VOICEOVER

This is Up Close, the research talk show from the University of Melbourne, Australia.

ANDI HORVATH

I'm Dr Andi Horvath. Thanks for joining us. Today we bring you Up Close to one of the very things that shapes our modern lives. No, not the technology as such, but what works in the background to drive it: the algorithm, the formalised set of rules governing how our technology is meant to behave.

As we'll hear, algorithms both enable us to use technology and to be used by it. Algorithms are designed by humans and just like the underpinnings of other technologies, say like drugs, we don't always know exactly how they work. They serve a function but they can have side-effects and unexpectedly interact with other things with curious or disastrous results.

Today, machine learning means that algorithms are interacting with, or developing other algorithms, without human input. So how is it that they can have a life of their own? To take us Up Close to the elusive world of algorithms is our guest, Paul Dourish, a Professor of Informatics in the Donald Bren School of Information and Computer Science at UC Irvine. Paul has written extensively on the intersection of computer science and social science and is in Melbourne as a visiting Miegunyah Fellow. Hello, and welcome to Up Close.

PAUL DOURISH

Morning, it's great to be here.

ANDI HORVATH

Paul, let's start with the term algorithm. We hear it regularly in the media and it's even in product marketing, but I suspect few of us really know what the word refers to. So let's get this definition out of the way: what is an algorithm?

PAUL DOURISH

Well, it is a pretty abstract concept so it's not surprising if people aren't terrible familiar with it. An algorithm is really just a way of going about doing something, a set of instructions or a series of steps you'll go through in order to produce some kind of computational result. So for instance, you know, when we were at school we all learned how to do long multiplication and the way we teach kids to do multiplication, well that's an algorithm. It's a series of steps that you can go through and you can guarantee that you're going to get a certain kind of result. So algorithms then get employed in computational systems, in computer systems to produce the functions that we want.

ANDI HORVATH

Where do we find algorithms? If I thought about algorithm-spotting say on the way to work, where do we actually encounter them?

PAUL DOURISH

Well, if you were to take the train, for instance, algorithms might be controlling the rate at which trains arrive and depart from stations to try to manage a stable flow of passengers through a transit system. If you were to Google something in the morning to look up something that you were going to do or perhaps to prepare for this interview, well an algorithm not only found the information for you on the internet, but it was also used to sort those search results and decide which one was the one to present to you at the top of the list and which one was perhaps going to come further down. So algorithms are things that lie behind the operation of computer systems; sometimes those are computer systems we are using directly and sometimes they are computer systems that are used to produce the effects that we all see in the world like for instance, the flow of traffic.

ANDI HORVATH

So Paul, we use algorithms every day in everything, whether it's work, rest, play, but are we also being used by algorithms?

PAUL DOURISH

Well, I guess there's a couple of ways we could think about that. One is that we all produce data; the things that we do produce data that get used by algorithms. If we want to think about an algorithm for instance that controls the traffic lights and to manage the flow of people through the streets of Melbourne, well, the flow of people through the streets of Melbourne is also the data upon which that algorithm is working. So we're being used by algorithms in the sense perhaps that we're all producing the data that the algorithm needs to get its job done.

But I think there's also a number of ways in which we might start to think that we get enrolled in the processes and effects of algorithms, so if corporations and government agencies and other sorts of people are making use of algorithms to produce effects for us, then our lives are certainly influenced by those algorithms and by the kinds of ways that they structure our interaction with the digital world.

ANDI HORVATH

So algorithms that are responsible for say datasets or computational use, the people who create them are quite important. Who actually creates these algorithms? Are they created by governments or commerce?

PAUL DOURISH

They can be produced in all sorts of different kinds of places and if you were in Silicon Valley and you were the sort of person who had a brand new algorithm, you might also be the sort of person who would have a brand new start-up. By and large, algorithms are produced by computer scientists, mathematicians and engineers.

Many algorithms are fundamentally mathematical at their heart and one of the ways in which computer scientists are interested in algorithms is to be able to do mathematical analysis on the kinds of things that computers might do and the sort of performance that they might have. But computer scientists are also generally in the business of figuring out ways to do things and that means basically producing algorithms.

ANDI HORVATH

One of the reasons we hear algorithms a lot these days is because they've caused problems, or at least confusion. Can you give us some tangible examples of where that's happened?

PAUL DOURISH

Sure. Well, I think we see a whole lot of those and they turn up in the paper from

time to time, and some are kind of like trivial and amusing and some have serious consequences. From the trivial side and the amusing side we see algorithms that engage in classification, which is an important area for algorithmic processing, and classifications that go wrong, places where an algorithm decides that because you bought one product you are interested in a particular class of things and it starts suggesting all these things to you.

I had a case with my television once where it had decided because my partner was recording Rocky and Bullwinkle, which is an old 1970s cartoon series [for just] America featuring a moose and a squirrel, that I must be interested in a hunting program so it started recording hunting shows for me. So although they're silly, they begin to show the way that algorithms have a role.

The more serious ones though are ones that begin to affect commerce and political life. A famous case in 2010 was what was called the flash crash, a situation in which the US stock market lost then suddenly regained a huge amount of value, about 10 per cent of the value of their system, all within half an hour, and nobody really knew why it happened. It turned out instead of human beings buying and trading shares, it was actually algorithms buying and trading shares. The two algorithms were sort of locked in a loop, one trying to offer them for sale and one trying to buy them up, and suddenly it spiralled out of control. So these algorithms, because they sort of play a role in so many different systems and appear in so many different places, can have these big impacts and in even those small trivial cases or ones that begin to alert us or tune us to where the algorithms might be.

ANDI HORVATH

Tell us about privacy issues; that must be something that algorithms don't necessarily take seriously.

PAUL DOURISH

Well, of course the algorithm works with whatever data it has to hand, and data systems may be more or less anonymised, they may be more or less private. One of the interesting problems perhaps is that the algorithm begins to reveal things that you didn't necessarily know that your data might reveal.

For example, I might be very careful about being tracked by my phone. You know, I choose to turn off those things that say for instance where my home is, but if an algorithm can detect that I tend to be always at the same place at 11 o'clock at night or my phone is always at the same place at 11 o'clock at night and that's where I start my commute to work in the morning, then those patterns begin to build up and there can be privacy concerns there. So algorithms begin to identify patterns in data and we don't necessarily know what those patterns are, nor are we provided necessarily with the opportunity to audit, control, review or erase data. So that's

where the privacy aspects begin to become significant.

ANDI HORVATH

Is there an upsurge about societal concerns about algorithms? Really, I'm asking you the question, why should we care about algorithms? Do we need to take these more seriously?

PAUL DOURISH

I think people are beginning to pay attention to the ways in which there can be potentially deleterious social effects. I don't want to sit here simply saying that algorithms are dangerous and we need to be careful, but on the other hand there is this fundamental question about knowing what it is the algorithm is doing and being conscious of its fairness.

On the trivial side, there is an issue that arose around the algorithm in digital cameras to detect faces, when you want to focus on the face. It turned out after a while that the algorithms in certain phones looked predominantly for white faces but were actually very bad at detecting black faces. Now, those kinds of bias aren't very visible to us, as the camera just doesn't work. Those are perhaps where as a society we need to start thinking about what is being done for us by algorithms, because lurking within those algorithms, unknown to people and certainly not by design, can be all sorts of unconscious biases and discriminations that don't necessarily reflect what we as a society want.

ANDI HORVATH

Are we being replaced by algorithms? Is this something that's threatening jobs as we know it?

PAUL DOURISH

Well, I certainly see plenty of cases where people are concerned about that and talk about it, and there's been some in the press in the last couple of years that talk for instance about algorithms taking over HR jobs in human resources, interviewing people for jobs or matching people for jobs. By and large though, lots of these algorithms are being used to supplement and augment what people are doing. I don't think we've seen really large-scale cases so far of people being replaced by algorithms, although it's certainly a threat that employers and others can hold over people.

ANDI HORVATH

Sure. Draw the connection for us between algorithms and this emerging concept of big data.

PAUL DOURISH

Well, you can't really talk about one without the other; they go together so seamlessly. Actually, one of the reasons that I've been talking about algorithms lately is precisely because there's so much talk about big data just now. The algorithms and the data go together. The data provides the raw material that the algorithm processes and the algorithm is generally what makes sense of that data.

We talk about big data not least in terms of this idea of being able to capture and collect, to get information from all sorts of sensors, from all sorts of things about the world, but it's the algorithm that then comes in and makes sense of that data, that identifies patterns and things that we think are useful or interesting or important. I might have a large collection of data that tells me what everybody in Victoria has purchased in the supermarket for the last month, but it's an algorithm that's going to be able to identify within that dataset well, here are dual income families in Geelong or the sort of person who's interested in some particular kind of product and amenable to a particular kind of marketing. So they always go together; you never have one without the other.

ANDI HORVATH

But surely there are problems in interpretation and things get lost in translation.

PAUL DOURISH

That's a really interesting part of the whole equation here. It's generally human beings have to do the interpretation; the algorithm can identify a cluster. It can say, look, these people are all like each other but it tends to be a human being who comes along and says now, what is it that makes those people like each other? Oh, it's because they are dual income families in Geelong. There's always a human in the loop here. Actually, the problem that we occasionally encounter, and it's like that problem of inappropriate classification that I mentioned earlier, the problem is that often we think we know what the clusters are that an algorithm has identified until an example comes along that shows oh, that wasn't what it was at all. So the indeterminacy of both the data processing part and the human interpretation is where a lot of the slippage can occur.

ANDI HORVATH

I'm Andi Horvath and you're listening to Up Close. In this episode, we're talking about the nature and consequences of algorithms with informatics expert Paul Dourish. Paul, given that algorithms are a formalised set of instructions, can't they simply be written in English or any other human language?

PAUL DOURISH

Well, algorithms certainly are often written in English. There's all sorts of ways in which we write them down. Sometimes they are mathematical equations that live on a whiteboard. They often take the form of what computer scientists call pseudo-code, which looks like computer code but isn't actually executable by a computer, and sometimes they are in plain English. I used the example earlier of the algorithm that we teach to children for how to do multiplication; well, that was explained to them in plain English. So they can take all sorts of different forms. Really, that's some of the difficulty about the notion of algorithm is this very abstract idea and it can be realised in many different kinds of ways.

ANDI HORVATH

So the difference between algorithms and codes and pseudo-codes are different forms of abstraction?

PAUL DOURISH

In a way, yes. Computer code is the stuff that we write that actually makes computers do things, and the algorithm is a rough description of what that code might be like. Real programs are written in specific programming languages. You might have heard of C++ or Java or Python, these are programming languages that people use to produce running computer systems. The pseudo-code is a way of expressing the algorithm that's independent of any particular programming language. So if I have a great algorithm, an idea for how to produce a result or sort a list or something, I can express it in the pseudo-code and then different programmers who are working in different programming languages can translate the algorithm into the language that they need to use to get their particular work done.

ANDI HORVATH

Right. Now, I've heard one of the central issues is that we can't really read the algorithm once it's gone into code. It's like we can't un-cook the cake or reverse engineer it. Why is that so hard?

PAUL DOURISH

Well, we certainly can in some cases; it's not a hard and fast rule. In fact, most computer science departments, like the one here at Melbourne, will teach people how to write code so that you can see what's going on. But there are a couple of complications that certainly can make it more difficult.

The first is that the structure of computer systems requires that you do more things than simply what the algorithm describes. An algorithm is an idealised version of what you might do, but in practice I might have to do all sorts of other things as well, like I'm managing the memory of the computer and I'm making sure the network hasn't died and all these things. My program has lots of other things in it that aren't just the algorithm but are more complicated.

Another complication is that sometimes people write code in such a way that it hides the algorithm for trade secret purposes. I don't want to have somebody else pick up on and get my proprietary algorithm or the secret source for my business or program, and so I write the software in a deliberately somewhat obscure way.

Then the other problem is that sometimes algorithms are distributed in the world, they don't all happen in one place. I think about the algorithms for instance that control how data flows across the internet and tries to make sure there isn't congestion and too much delay in different parts of the network. Well, those algorithms don't really happen in one place, they happen between different computers. Little bits of it are on one computer and little bits of it are on the other and they act together in coordination to produce the effect that we desire, so it can be often hard to spot the algorithm within the code.

ANDI HORVATH

Tell us more about these curious features of algorithms. They almost sound like a life form.

PAUL DOURISH

Well, I think what often makes algorithms seem to take on a life of their own, if you will, is that intersection with data that we were talking about earlier, because I said data and algorithms go together. There is often a case for instance where I can know what the algorithm does but if I don't know enough about the data over which the algorithm operates, all sorts of things can happen.

There's a case that I like to use as an example that came from some work that a friend of mine did a few years ago where he was looking at the trending topics on Twitter, and he was working particularly with people in the Occupy Wall Street movement who were sure that they were censored because their movement, the political discussion around Occupy Wall Street, never became a trending topic on Twitter. People were outraged, how can Justin Bieber's haircut be more important

than Occupy Wall Street? When they talked to the Twitter people, the Twitter people were adamant that they weren't censoring this, but nonetheless they couldn't really explain in detail why it was that Occupy Wall Street had not become a trending topic. You can explain the algorithm and what it does, you can explain the mathematics of it, you can explain the code, you can show how a decision is made, but that decision is made about a dataset that's changing rapidly, that's to do with everything that's being Tweeted online, everything that's being retweeted, where it's being retweeted, where it's being retweeted, how quickly it's being retweeted. What the algorithm does, even though it's a known, engineered artefact, is still itself somehow mysterious.

So the lives that algorithms take on in practice for us when we encounter them in the world or when they act upon us or when they pop up in our Facebook newsfeed or whatever, is often unknowable and mysterious and lively, precisely because of the way the algorithm is entwined with an ever roiling dataset that keeps moving.

ANDI HORVATH

I love the term machine learning, and it's really about computers interacting with computers, algorithms talking to other algorithms without the input of humans. That kind of spooks me. Where are we going?

PAUL DOURISH

Yeah. Well, I think the huge, burgeoning interest in machine learning has been spurred on by the big data movement. Machine learning is something that I was exposed to when I was an undergraduate student back more years ago than I care to remember; it's always been there. But improvements in statistical techniques and the burgeoning interest in big data and the new datasets mean that machine learning has taken on a much greater significance than it had before.

What machine learning algorithms typically do is they identify again patterns in datasets. They take large amounts of data and then they tell us what's going on in that. Inasmuch as we are generating more and more data and inasmuch as more and more of our activities move online and then become, if you like, ?datafiable?, things that can now be identified as data rather than just as things we did, there is more and more opportunity for algorithms, and particularly for machine learning algorithms, to identify patterns within that.

I think the question, as we said, is to what extent one knows what a machine learning algorithm is saying about one. Indeed, even, as I suggested with the Twitter case, even for people who work in this space, even for people who are developing the algorithms, it can be hard for them to know. It's that sort of issue of knowing, of being able to examine the algorithms, of making algorithms accountable to civic, political and regulatory processes, that's where some of the real challenges are that are

posed by machine learning algorithms.

ANDI HORVATH

We're exploring the social life of algorithms with computer and social scientist Paul Dourish right here on Up Close. And yes, we're coming to you no doubt thanks to several useful algorithms. I'm Andi Horvath. Let's keep moving with algorithms. You say that algorithms aren't just technical, that they're social objects. Can you tell us a bit more what that means?

PAUL DOURISH

Well, I think we can come at this from two sides. One side is the algorithms are social as well as technical because they're put to social uses. They're put to uses that have an impact on our world. For example, if I'm on Amazon and it recommends another set of products that I might like to look at, or it recommends some and not others, there's some questions in there about why those ones are just the right ones. Those are cases where social systems, systems of consumption and purchase and identification and so forth are being affected by algorithms. That's one way in which algorithms are social; they're put to social purposes.

But of course, the other way that algorithms are social is that they are produced by people and organisations and professions and disciplines and all sorts of other things that have a grounding in the social world. So algorithms didn't just happen to us, they didn't fall out of the sky, we have algorithms because we make algorithms. And we make algorithms within social settings, and they reflect our social ideas or our socially-constructed ideas about what's desirable, what's interesting, what's possible and what's appropriate. Those are all ways in which the algorithms are pre-social. They're not just social after the fact but they are social before the fact too.

ANDI HORVATH

Paul, you've mentioned how algorithms are kind of opaque, but yet you also mention that we need to make them accountable, submit them to some sort of scrutiny. So how do we go about that?

PAUL DOURISH

This is a real challenge that a number of people have been raising in the last couple of years and perhaps especially in light of the flash crash, that moment where algorithmic processing produced a massive loss of value on the US stock market. There are a number of calls for corporations to make visible aspects of their own algorithms and processing so that it can be deemed to be fair and above board. If

you just think for a moment about how much of our daily life in the commercial sector is indeed governed by those algorithms and what kind of impact a Google search result ordering algorithm has; there's lots of consequences there, so people have called for some of those to be more open.

People have also called for algorithms to be fixed. This is one of the other difficulties is that algorithms shift and change; corporations naturally change them around. There was some outrage when Facebook indulged in an experiment in order to see whether they could tweak the algorithms to give people happier or less happy results and see if that actually changed their own mood and what kinds of things they saw. People were outraged at the idea that Facebook would tweak an algorithm that they felt, even though it obviously belonged to Facebook, was actually an important part of their lives. So keeping algorithms fixed in some sense is one sort of argument that people have made, and opening things up to scrutiny.

But the problem with opening things up to scrutiny is well, first, who can actually evaluate these things? Not all of us can. And also of course that in the context of machine learning, the algorithm identifies patterns in data, but what's the dataset that we're operating over? In fact, we can't even really identify what those things are, we're only saying there's a statistical pattern and that some human being is going to come along and assign some kind of value to that. So some of the algorithms are inherently inscrutable. The algorithm processes data and we can say what it says about the data, but if we don't know what the data is and we don't know what examples it's been trained on and so forth, then we can't really say what the overall effect and impact is.

ANDI HORVATH

Will scrutiny of algorithms, whether we audit or control them, be affected by, say, intellectual property laws?

PAUL DOURISH

Well, this is a very murky area, and in particular it's a murky area internationally, where there are lots of different standards in different countries about what kind of things can be patented, controlled and licensed and so forth. Algorithms themselves are patentable objects. Many people choose to patent their algorithms, but of course patenting something requires disclosing it and so lots of other corporations decide to protect their algorithms as trade secrets, which are basically just things you don't tell anybody.

The question that we can ask about algorithms is actually also how they move around in the world and those intellectual property issues, licensing rights, patenting and so forth are actually ways that algorithms might be fixed in one place within a particular corporate boundary but also move around in the world. So no one has

really I think got a good handle on the relationship between algorithms and intellectual property.

They are clearly valuable intellectual property, they get licensed in a variety of ways, but this is again one of these places where the relationship between algorithm and code is a kind of complicated one. We have developed an understanding of how to manage those things for code; we have a less good understanding right now of how to manage those things for algorithms. I should probably say, since we're also talking about data, no idea at all about how to do this for data.

ANDI HORVATH

These algorithms, they've really got a phantom-like presence and yet they've got so much potential and possibility. They are practical tools that help with our lives. But what are the consequences of further depending upon the algorithms in our world?

PAUL DOURISH

I think it's inevitable and not really problematic. From my perspective, algorithms in and of themselves are not necessarily problematic objects. Again, if we say that even the things that we teach our children for how to do multiplication are algorithms, there's no particular problem about depending on that. I think again it's the entwining of algorithms and data, and one of the things that an algorithmic world demands is the production of data over which those algorithms can operate, and all the questions about ownership and about where that algorithmic processing happens matter.

For example, one manifestation of an algorithmic and data-driven world is one in which you own all your data and you do the algorithmic processing and then make available the results if you so choose. Another version of that algorithmic and [data-centred/data-central] world is one in which somebody else collects data about you and they do all the processing and then they tell you the results, and there's a variety of steps in between. So I don't think the issue is necessarily about algorithms and how much we depend on algorithms. Some people have claimed we're losing our own ability to remember things because now Google is remembering for us.

ANDI HORVATH

It's an outsourced memory.

PAUL DOURISH

Yes, that's right, or there's lots of things about people using their Satnav and driving into the river, right, because they're not anymore remembering how to actually drive

down the road or evaluate the things in front of them, but I'm a little sceptical about those. I do think the question about how we want to harness the power of algorithmic processing, how we want to make it available to people, and how it should inter-function with the data that might be being collected from or about people, those are the questions that we need to try to have a conversation about.

ANDI HORVATH

Paul, I have to ask you, just like we use our brain to understand our brain, can we use algorithms to understand and scrutinise algorithms?

PAUL DOURISH

[Laughs] Well, we can and actually, we do. One of the ways in which we do already is that when somebody develops a new machine learning algorithm we have to evaluate how well it does. We have to know is this algorithm really reliably identifying things. We sort of pit algorithms against each other to try to see whether the algorithm is doing the right work and evaluate the results of other kinds of algorithms. So that actually already happens.

Similarly, as I suggested on the internet, the algorithm for congestion control is really a series of different algorithms happening in different places that work cooperative or not in order to produce or not a smooth flow of data. Though we don't have to worry just yet I think about a sort of war between the algorithms or any kind of algorithmic singularity.

ANDI HORVATH

Paul, what do you mean by the singularity? Is this really a Skynet moment?

PAUL DOURISH

Well, the singularity is this concept that suggests that at some point in the development of intelligent systems, they may become so intelligent that they can design their own future versions and the humans become irrelevant to the process of development. It's a scary notion; it's one I'm a little sceptical about, and I think actually the brittleness of contemporary algorithms is a great example of why we're not going to get there within any short time.

I think the question though is still how do we want to understand the relationship between algorithms and the data over which they operate? A great example is IBM's Watson, which a couple of years ago won the Jeopardy TV show, and this was a real breakthrough for artificial intelligence. But on the other hand you've got to ask, what is it that Watson knows about? Well, a lot of what Watson knows it knows from

Wikipedia and I'm not very happy when my students cite Wikipedia and I'm not terribly sure that I need to be afraid of the machine intelligence singularity that also is making all its inferences on the basis of Wikipedia.

ANDI HORVATH

Paul, thanks for being our guest on Up Close and allowing us to glimpse into the world of the mysterious algorithm. I feel like I've been in the movie Tron.

PAUL DOURISH

[Laughs] Yes, well, we don't quite have the glowing light suits unfortunately.

ANDI HORVATH

We've been speaking about the social lives of algorithms with Paul Dourish, a professor of informatics in the Donald Bren School of Information Computer Science at UC Irvine. You'll find a full transcript and more info on this and all our episodes on the Up Close website. Up Close is a production of the University of Melbourne, Australia. This episode was recorded on 22 February 2016. Producer was Eric van Bommel and audio recording by Gavin Nebauer. Up Close was created by Eric van Bommel and Kelvin Param. I'm Dr Andi Horvath. Cheers.

VOICEOVER

You've been listening to Up Close. For more information visit upclose.unimelb.edu.au. You can also find us on Twitter and Facebook. Copyright 2016, the University of Melbourne.

© The University of Melbourne, 2016. All Rights Reserved.

Source URL: <https://upclose.unimelb.edu.au/episode/363-social-life-algorithms-shaping-and-being-shaped-our-world>