Computational beliefs

Article

Accepted Version

Grindrod, J. ORCID: https://orcid.org/0000-0001-8684-974X (2019) Computational beliefs. Inquiry: An Interdisciplinary Journal of Philosophy. ISSN 1502-3923 doi: 10.1080/0020174X.2019.1688178 Available at https://centaur.reading.ac.uk/87087/

To link to this article DOI: http://dx.doi.org/10.1080/0020174X.2019.1688178

Publisher: Taylor & Francis

Computational Beliefs

Dr Jumbly Grindrod
University of Reading
J.grindrod@reading.ac.uk

Abstract

In this paper, I outline and investigate the notion of *computational beliefs*: beliefs formed on the basis of a deliverance from a machine learning algorithm. Given the increased usage of such algorithms through smart devices, such beliefs are becoming increasingly common in everyday life. First, I argue that such beliefs can be successful (i.e. justified and true) by outlining particular examples that possess epistemic properties taken to be indicative of successful beliefs (i.e. being a reason for action, being reliable, being safe, and being sensitive). I then outline how computational beliefs are best understood as a form of what Sosa (2006) describes as *instrumental beliefs*. As such, it may be thought that computational beliefs hold little epistemological interest insofar as they are *reducible* to more basic epistemic processes that are already the focus of epistemological attention. However, in the final section, I argue that computational beliefs hold epistemological importance insofar as they have specific epistemically normative repercussions i.e. they give rise to *epistemic responsibility gaps*.

Keywords

Computational beliefs; learning algorithms; epistemic responsibility; instrumental beliefs; social epistemology

1. Introducing Computational Beliefs

Social epistemology recognises that a full account of the epistemic agent as a knower and believer must make some appeal to the social environment that the epistemic agent is in. But the social environment we now find ourselves in as epistemic agents is changing as we become more reliant on technology. It is no secret that we are becoming more dependent on computational devices in many aspects of life, using our computers and phones to write, purchase, learn, play, meet, and more. Data are now easily produced, stored, transported, copied, analysed, and generally utilised for a variety of purposes. This is not least because the technology required to analyse such data has seen a dramatic improvement. *Learning algorithms* are now available that, at the most general level, adjust the parameters by which they operate so that they perform at some (often very high) minimally-optimal level. These algorithms no longer have to operate via a predefined method explicitly formulated by data scientists, and as a result can provide ways of processing data that would otherwise have been out of human reach.

Today we are entering a situation whereby we are starting to rely on these algorithms as *epistemic sources* i.e. we form beliefs – justified beliefs – on the basis of the outcomes of such algorithms. The fact that this kind of belief has only recently become available should lead us to a *prima facie* position that it deserves philosophical attention in case it possesses epistemic features that are previously overlooked. Furthermore, it is reasonable to think that this kind of belief is on the rise not simply because the technology is available, but also because in some cases it is starting to *replace* other kinds of belief. Where previously we may have relied on testimony as a source of belief, there are now cases where we rely upon an algorithmic source. Philosophers ought to be sensitive to this kind of shift in our epistemic landscape and this is the second respect in which this new kind of belief via learning algorithm warrants inspection.

In recent years, there has been an increased philosophical focus the way we form beliefs in a digital environment. In particular, a number of authors have investigated the way information is ranked and displayed in search engines and news feeds. For example, Miller & Record (2013, 2017) have argued that the opacity of the algorithms used by search engines such as Google in order to provide search results can mean that the beliefs formed via those search results can sometimes be unjustified. Similarly, Simpson (2012) has argued that search engines effectively serve as 'surrogate experts', but that in doing so they fail to provide us with sufficient a level of objectivity due to the highly personalised results that the likes of Google now provide. Heersmink (2018) has argued from a virtue-epistemological perspective that there are a range of virtuous and vicious behaviours associated with such online inquiry. As we will see, the concerns of such authors are similar to my own. In particular, I will argue that the fact that we treat apps that employ learning algorithms as 'surrogate experts' on particular topics, combined with the opacity that such algorithms possess, leads to specific consequences regarding epistemic responsibility. But while the process of retrieving information via a search engine clearly will involve the kind of learning algorithms that are the topic of this paper, the beliefs formed via that process would not constitute a computational belief. Web pages tend to be human-authored documents, and so when one forms a belief based on a search engine result, there is an important sense in which this belief has a testimonial source, albeit mediated by a search engine. By contrast, this paper will be concerned with beliefs formed on the basis of a deliverance from a machine learning algorithm, and such beliefs will be referred to as *computational beliefs*.[1] This is

[1] I use the term 'deliverance' from (Sosa 2006), largely as a way of avoiding anthropomorphising these algorithms with talk of what they *decide, say, state,* etc. I use the term to refer to the proposition that an instrument presents as

clearly a very broad category, that would doubtless capture many beliefs formed in an academic setting, where machine learning algorithms may be employed during experimentation or data analysis. However, this paper is primarily concerned with the ordinary cases of computational beliefs, such as when we form beliefs based on the deliverances of our smart devices.

In this paper, I will investigate the extent to which computational beliefs can be viewed as a distinctive form of belief, suitable for epistemological theorising. In the debate concerning testimonial knowledge, a great deal of the discussion has concerned whether testimony as an epistemic source can be reduced to other, more basic, epistemic sources. In this paper, I will argue that computational beliefs are reducible in this sense. Nevertheless, computational beliefs are a distinct form of belief, worthy of study in their own right, due to the distinctive kinds of normative repercussions they hold. In the following section, I will outline more precisely what I have in mind by the term 'computational beliefs'. In section 3, I will attempt to motivate the idea that such beliefs can be justified and constitute knowledge. In section 4, I will consider the extent to which computational beliefs can be captured via more traditional epistemic sources e.g. perception, reason, and memory etc. There, I will argue that computational beliefs are best captured as what Sosa (2006) calls *instrumental belief*. In section 5, I will argue for reductionism about computational beliefs. Finally, in section 6, I will argue that computational beliefs are epistemologically interesting in their own right insofar as they give rise to *epistemic responsibility gaps*.

2. Cases of Computational Beliefs

The cases of interest are cases where an epistemic agent forms a belief based upon the deliverance of some app or program, and where that app or program uses a learning algorithm that analyses some large dataset in providing the output that it does. A learning algorithm is any algorithm that is able to optimise its own performance in ways that are not explicitly programmed. It will do so by 'learning' from a dataset. We can distinguish broadly between two different kinds of learning algorithm. A *supervised learning algorithm* is one that requires an annotated training set of data in order to undergo its learning phase. The training set is a sample of data that pairs the kind of data that the algorithm is designed to process with a desired set of outputs. For example, if a supervised learning algorithm is being designed to recognise scans of cancerous lungs, then the training set will consist of pairs of a lung scan plus a verdict on whether it is a cancerous scan or not. The algorithm will then go through a training phase whereby it will attempt to classify each input of the training set correctly and will continue to adjust its parameters until it succeeds in this task to some minimally optimal level. An *unsupervised learning algorithm,* on the other hand, does not require an annotated training set in order to proceed. Instead, it will recognise certain internal patterns present within the data. Clustering algorithms are a common example, whereby datapoints are organised into groups based on their statistical similarities.[2]

Learning algorithms are of particular interest because they are at the forefront of many recent advances, and because these algorithms themselves possess philosophically interesting properties, chief among them being their *opacity* (Miller and Record 2013, p. 128). That is, it is

being the case, as they were designed to do. For computational beliefs, the instrument in question will rely upon a learning algorithm.

[2] A third kind of learning algorithm is a *reinforcement learning algorithm*. A reinforcement learning algorithm is one that learns 'on-the-go' in its final operating environment. The idea is that evaluations are assigned to the outcomes that the algorithm provides. As a result, the algorithm can continuously adjust its own parameters of operation (sometimes in a trial-and-error fashion) so as to optimise its ability to achieve positive evaluations.

often not possible for any given human agent, nor any set of human agents, to properly understand the manner in which the algorithm has processed the data and reached the outcome that it has. There are (at least) two reasons for this. First, the overall manner in which these algorithms operate is often not explicitly encoded in a human-friendly language that a programmer would be able to read off. This is most clear in the case of neural networks. Neural networks consist of layers of neurons that will simply send a signal to the neurons it is connected to depending upon the kind of input that it has received. If the weights associated with the neural connections are adjusted in an appropriate manner, a neural network is able to process data in an incredibly sophisticated way and perform tasks of surprising complexity. As just one example, Owens et al. (2015) produced a deep learning system that is able to take as input soundless videos of a drumstick hitting various objects (e.g. a gutter, a banister, a bush etc.) and insert an appropriate matching sound based purely on the visual data within the video. But importantly, beyond the weights assigned to each neuron and the initial framework of the network, there is no record to be had of *how* the network processes data. Instead, the best way for a programmer to understand how a neural network processes data is simply by exploring the types of outputs the network provides given certain types of input. Secondly, the manner in which the algorithm operates – the way in which it provides an output based on a given input – is often too complex for a human to comprehend. As stated, learning algorithms learn by adjusting their operational parameters until they have reached some minimally-optimal level of performance, and the number of parameters can be potentially huge. In the case of neural networks, the connection weights that hold between neurons can rise into the millions. Human agents are not be able to keep track of how each is adjusted, and so it seems the operation of such algorithms is simply beyond our ken.

We are already coming to rely epistemically upon such algorithms. I will present three examples. If I want to know about traffic on a particular road, it used to be that I would either have to go see the road for myself or rely on some testimonial chain that leads back to someone who has been on the road themselves. I would have listened out for a traffic report, where a reporter will inform about the traffic on the road on the basis of further testimony, or on some kind of footage (e.g. CCTV), or perhaps from their own birds-eye view from a helicopter. Now, however, I tend to find out about traffic via Google Maps. Google Maps analyses the traffic on a given road on the basis of a huge set of constantly updated data points. In particular, it is able to estimate the average speed of their travelling customers on the basis of locational data of all android phones (with location services enabled), and all other smart devices with Google Maps running.[3] From this dataset, it is able to provide an accurate estimate of the traffic in a given area, and it will notify a user via their smartphone if they are about to travel on a road with a high amount of traffic. In this way, we rely on the algorithms that are operative as part of Google Maps as an epistemic source with regard to traffic, and we form beliefs on the basis of what it tells us and act accordingly – possibly by changing our route so as to avoid a particularly busy road. It seems plausible also that we are *justified* in doing so, although I return to this issue in the next section.

A second example concerns cancer diagnosis. In recent years, a great deal of research has been devoted to using various types of learning algorithms to identify the presence of cancerous cells

[3] In the same way, Google Maps is able to identify when there is a running event on in a particular town i.e. when there are a huge number of highly-concentrated smart devices moving at a jogging pace through a town.

either via a scan or a tissue sample (Kuruvilla and Gunavathi 2014, Zhou et al. 2002).[4] The exciting possibility of this research is that there is no principled reason to think that such an approach could not exceed expert human diagnostic abilities. Whether we are close to that situation or not, it seems easy to imagine a possibility where this type of highly accurate system is successfully produced and is then employed in a medical setting. In that case, when the diagnostic system informs a patient that their scan or biopsy has indicates the presence of cancerous cells, the patient would be using this system as an epistemic source and would then form a belief and adjust their behaviour accordingly.

Finally, a third example concerns machine translation, the most famous example being Google Translate. Suppose that while travelling to a conference in Germany, an exasperated train conductor calls me a 'dummkopf'. Not knowing what this word means, I can then type the word into Google Translate, which informs me that the English translation is 'idiot'. Here Google Translate serves as an epistemic source regarding certain linguistic facts i.e. facts about translations between two languages. Again, I can then justifiably form the belief that the train conductor called me an idiot and adjust my behaviour accordingly (i.e. look indignant and internalise negative feelings). But Google Translate does not operate via any kind of previously-recorded translation dictionary. Instead, it relies on a vast corpus of linguistic data, as well as a vast corpus of translations between languages.[5] Essentially, translations are reached via analysis of the properties of linguistic corpora rather than testimony from a translator.

With these examples, we find that algorithms already serve as an epistemic source in ordinary life and will do so in areas of increasing significance, such as in medical settings. I take it to be highly plausible that, provided there are no relevant defeaters or other situational deficiencies, such cases are cases of justified belief and knowledge. There may be some, however, who are more suspicious of the quality of computational beliefs. Certainly, I do not want to claim that all cases where we find these algorithms playing the putative role of an epistemic source are cases where they do so successfully. It doesn't seem plausible, for instance, that when e-Harmony states that I am well-matched with a given individual that I can then justifiably believe that I am well-matched romantically with that individual. Instead, I only want to claim that there are cases where we do rely on an algorithm as an epistemic source. In order to further assuage the concern that such beliefs are improperly formed, in the next section I will briefly examine the epistemic properties of computational beliefs in an effort to show that they do possess the same properties as more run-of-the-mill beliefs.

3. Epistemic properties of computational belief

The contemporary epistemological literature provides us with a range of interesting concepts that I intend to use here in a kind of diagnostic role. In the post-Gettier years, concepts such as sensitivity, safety, reliability, etc. have been proposed as part of an analysis of knowledge or justified belief. But while many consider the various proposals associated with these concepts to be deficient in some way, these concepts still have a use in epistemological theorising insofar as they do capture features common across a good number of the successful cases of belief. With that in mind, if we find that the given set of beliefs we are concerned with fall under the

[4] Neural networks are particularly suitable for such a task as they are well-suited to processing visual data. In particular, *convolutional neural networks* – which possess an architecture specifically designed to process visual data – have produced a great deal of impressive results in recent years.

[5] Note that a corpus of translations *does not amount* to a translation dictionary. A translation dictionary will provide a list of words in one language accompanied by their meaning in another language. A body of text coupled with a translation of it obviously will not take this form.

application of such concepts, this gives us some reason to think that the beliefs we are concerned with are successful.[6] Given this, I want to focus in this section on whether it is possible for computational beliefs to possess the following properties: being a reason for action, reliability, safety, and sensitivity.

3.1. Being a reason for action

Some have argued that there is an interesting relation between the beliefs we hold and the kinds of actions we are rationally-permitted to perform; that is, there is a normative relation between our justified beliefs and our actions. Stanley & Hawthorne (2008, p. 577) flesh this out in terms of their *Action-Knowledge Principle*:[7]

> Action-Knowledge Principle: Treat proposition p as a reason for acting only if you know that p.

We can imagine this principle being adjusted in a number of ways, perhaps by replacing knowing that p with justifiably-believing that p, and there is plenty of interesting discussion to be had regarding how to correctly capture the wide variety of cases where we reason about how to act. Here I'll assume the justified belief version of the principle as a kind of heuristic by assuming that if we have a case where an agent is right to treat p as a reason for acting, then this is good reason to think that they justifiably believe that p.

In the three examples of computational belief considered above, we can ask: is there positive reason to act upon the beliefs in question? As I suggested, I believe in all three cases there can be positive reason to act. For example, suppose that I usually travel home via the King's road, which is the quickest route provided that it doesn't have one of its occasional traffic jams. While getting to the car, Google Maps alerts me that there is indeed a traffic jam on the King's road. In that case, if I believe what Google Maps has stated, *it seems that I would be justified in acting upon that belief*. I will either take an alternative route home or contact anyone whose expecting me to be at home on time. Either way, my reason for acting is my belief that there is a traffic jam on the King's road, and the source of this belief is Google Maps. So we find that computational beliefs can rightly play the role of reasons for action, and this is good reason to think that they are justified beliefs.

3.2. Reliability

Reliabilist theories of justification at least state that if S's belief that p is formed on the basis of reliably truth-conducive belief-forming processes, then S's belief that p is justified (Goldman 1979). Whether reliabilism as a theory of justification is still considered plausible, the reliability of a belief is nevertheless considered an important property when considering the success of a given belief, and it is commonly thought that for a great deal of cases, the success or failure of a given belief will depend on its reliability. For the purposes of this paper, I am happy to view reliability as a necessary condition of justified belief.

Computational beliefs can clearly be reliable. Algorithms are designed and tested to reach some minimal level of accuracy and so provide some level of guarantee over their own performance. Of course, they are never 100% accurate, and so any belief-forming process based on their

[6] We see this kind of approach in, for instance, (Lackey 2008).

[7] Stanley & Hawthorne actually strengthen their claim to one of necessity and sufficiency in the following principle: 'Where one's choice is p-dependent, it is appropriate to treat the proposition that p as a reason for acting iff you know that p' (p. 578).

accuracy will not be completely reliable, but it's doubtful that there are many if any belief-forming processes that would meet a perfect standard (even basic deductive reasoning goes wrong sometimes). Furthermore, as stated earlier, it seems plausible that for a great number of tasks, such algorithms will exceed human performance, and so as far as reliability is concerned, computational beliefs look very well-off. We may have quibbles with the particular examples I have used: perhaps your experience of Google Translate or Google Maps has rarely been a happy one. It is difficult to place an exact success rate on either app regarding the particular tasks I have in mind, and it is likewise difficult to place a precise figure on what constitutes the threshold of reliability. But if the real-life examples seem unsatisfactory, I am happy to appeal to future cases where a higher success rate has been achieved and more stringent standards of reliability are met.[8]

3.3. Safety

S's belief that p is safe iff there are no nearby possibilities where S believes p and p is false (put in terms of subjunctives: if S were to believe p, p would not be false) (Sosa 1999). In the case of computational beliefs, the idea would be that there are no close-by or realistic possibilities whereby the app provides the deliverance that p, S subsequently believes p on that basis, and nevertheless p does not hold. Initially, it may seem that this condition doesn't hold for some of the cases I am considering, such as the Google Maps case. However, it is important not to confuse cases where a system like Google Maps goes wrong as proof of lack of safety. Take the earlier example, where I believe that there is a traffic jam on the King's Road. That belief was formed on the basis of a deliverance of Google Maps. Google Maps produced that deliverance on the basis of the available locational data and the analysis produced by the algorithm. But once we have the case fixed like this, and we ask whether there are nearby possibilities where Google Maps said that there was a traffic jam when there was none, this seems much more doubtful. The application will obviously proceed algorithmically, and there are no nearby possible worlds where the algorithm proceeded in a different manner. For the algorithm to have behaved differently we would need to go back to its design and training phases. In which case, in order for Google Maps to have provided an incorrect deliverance that there was a traffic jam when there was none, there would have to be a kind of anomalous situation in their locational data e.g. the distribution and movement of smart devices on the King's road would be indicative of a traffic jam even if there was none. But this is a possibility far-removed from the actual world, involving many cars not being on the road and many smart devices in particular locations, and we ought not include this in the realm of nearby possibilities. For this reason, I think it is plausible to claim that the Google Maps case is a case of safe belief. I should emphasise, however, that all I intend to show here is that it is possible for computational beliefs to be safe, that there is nothing about computational beliefs that prevents from being safe.

3.4. Sensitivity

[8] A referee raises the point that it is difficult to see how reliability could be an independent criterion from being a reason for action, because it plausibly seems to be the case that the former is at least necessary for the latter. That may well be the case, but in this section I do not want to assume any such conceptual relations between the epistemic notions that I am using to diagnose the success of computational beliefs. This does put me in danger of double-counting the reasons in favour of thinking that computational beliefs are successful if it turns out that, say, a belief being reliable is the same thing as a belief being a reason for action. But I think that is a price worth paying in order to investigate the success of computational beliefs in a way that is consistent with a wide range of epistemological views.

The issue of sensitivity with regard to computational beliefs is not straightforward. S's belief that p is sensitive iff in the nearest possible world where ¬p, S does not believe that p. In the Google Maps case, this would mean that in the nearest possible world where there was no traffic jam, Google Maps would not provide a deliverance that there is one; in the Google Translate case, this would mean that in the closest possible world where 'dummkopf' does not translate as 'idiot', Google Translate would not provide the deliverance that it does; and in the cancer detection case, this would mean that in the closest possible world where an actually cancerous patient does not have cancer, the relevant system would not provide a deliverance that it does. One useful way to think about this is whether the system (app, program, etc.) would have provided a *false positive* in the closest possible world where the proposition is false. Machine learning systems tend to perform to sub-optimally, meaning that there will always be some chance of a false positives and false negatives. Nevertheless, I suggest that the three cases of computational belief differ with regard to their sensitivity.

To take the Google Translate case first, it seems plausible that this belief is sensitive. Plausibly, the closest possible world in which it is not the case that 'dummkopf' translates into 'idiot' – i.e. where the linguistic facts about German or English are different – is one in which Google Translate would operate in the same way and would pick up on this difference. So this looks like a sensitive computational belief.

Whether the Google Maps case is sensitive is, frankly, difficult to discern. It may be that one of the closest possible worlds where there is no traffic jam is one that would give rise to a *false positive* deliverance from Google Maps (perhaps because the nearest possible world where there is no traffic jam is one in which there is something *very close to a traffic jam* which the system would struggle with), but far more would have to be known about the system's operation in order to make a call either way. For the purposes of this paper, I suggest it simply isn't clear whether the belief is sensitive.

The case of a cancer diagnosis is an interestingly different case with regards to sensitivity. Matthias (2004, p. 177), in discussion of Zhou et al.'s (2002) neural network ensemble for identifying cancerous biopsies, says 'The system has been constructed so that false negative diagnoses are highly improbable (proclaiming the patient to be healthy when there are, in reality, cancer cells present), but there is accordingly much less precaution about false positiva.' This is because the repercussions of a false negative diagnosis – a cancer unknowingly threatening the patient's health – are far greater than the repercussions of a false positive diagnosis. As a result, the possibility of a false positive in the closest possibility world where one does not in fact have cancer may be significant, and so the sensitivity of such a belief is questionable. But this particular feature of this cancer diagnostic system raises an important point. Just as one could tilt a system towards producing fewer false negative results at the cost of producing more false positive results, one could also tilt a system towards producing fewer false positive results at the cost producing more false negative results – this would be to make the system more conservative in its deliverances rather than more liberal. It would then become more plausible that computational beliefs formed on the basis of such a system would be sensitive. The point of this is to show that there is nothing inherent in the nature of computational beliefs that prevents them from being sensitive. I take the Google Translate to show this, but we can also motivate this claim by considering a machine learning system that is by design very unlikely to produce false positives.

It is also worth keeping in mind that if it turns out that a particular set of computational beliefs are not sensitive, this does not entail that they are unsuccessful, as it has been argued elsewhere there are cases of insensitive knowledge. Vogel (2007, p. 82) presents the following case, among others:

> *Hole-In-One Case*: Sixty golfers are entered in the Wealth and Privilege Invitational Tournament. The course has a short but difficult hole, known as the "Heartbreaker". Before the round begins, you think to yourself that, surely, not all sixty players will get a hole-in-one on the "Heartbreaker".

Here, you clearly do know that not all sixty players will get a hole in one, but in the closest possible world where ¬p is the case, you still believe that p, and so your belief is insensitive. Vogel generalises by arguing that sensitivity simply fails to capture cases of inductive knowledge, of which this is an instance. To this extent, it would not be surprising if computational beliefs are insensitive, as there is in some sense a process of induction that takes place, albeit by the algorithm rather than the user.

To summarise this section, we have seen that computational beliefs can clearly be reasons for action, are highly reliable, they can be safe, and they can be sensitive. The point of this is to show that computational beliefs share many of the same properties as those beliefs typically taken to be successful.

4. Epistemic sources

Thus far I have spoken of learning algorithms that analyse large datasets as an epistemic source of our computational beliefs, insofar as it is the deliverances of such algorithms (or the applications that contain them) that we base our computational beliefs on. But a complete theory of epistemic sources may seek to properly delineate between only a limited number of sources and seek to account for all beliefs with only this limited number. As a starting point that I alluded to earlier, we may seek to distinguish between perception, inference, and memory as epistemic sources, as it is not plausible that beliefs generated from one can be accounted for via appeal to one or both of the other two. This issue has come into sharp focus more recently, particularly with regards to testimonial belief, and whether testimony ought to be viewed as a distinct kind of epistemic source, or whether the process of forming testimonial beliefs can be reduced to the employment of other epistemic sources. In this section, I will pose that question with regards to computational beliefs.

When it comes to computational beliefs, the question is whether such beliefs can be captured using more basic epistemic sources, such as perception, inference, and memory. It doesn't straightforwardly fit into any such category. Memory can be ruled out immediately, as the process of obtaining a computational belief is not akin to remembering a certain proposition to be the case. Nor does computational belief look akin to perceptual beliefs. Perceptual beliefs work in the following way: if I have a perceptual experience of a bottle in front of me, then absent defeaters, *that perceptual experience justifies me in believing that there is a bottle in front of me*. Granted this does oversimplify the matter regarding perceptual beliefs somewhat, as there is a great deal of controversy over the precise content of a justifiably-formed perceptual belief. But it does not seem to capture computational beliefs correctly to say that if I have a perceptual experience of an app (e.g. Google Translate) providing some deliverance, then in virtue of this perceptual experience I am justified in believing that deliverance. Instead, the perceptual experience justifies me at best in believing merely that there is an app before me providing some deliverance. So computational beliefs aren't going to be fully-captured as a form of perceptual belief.

What about capturing computational beliefs as a form of inference? Certainly it does seem plausible that a kind of inference is made in the production of the computational belief, but it is not plausible that the epistemic agent is making all of them. For instance, the Google Maps case is one in which a kind of inference is made on the basis of vast amounts of locational data to a verdict about the traffic on a given road, which is then subsequently believed by the agent. But the agent did not draw those inferences, the heavy inferential work was done by the algorithm. This is reflected by the fact that the agent is not blameworthy if that inferential procedure goes wrong, nor is he worthy of credit when it succeeds (if were worthy of such credit, he would be worthy of doing something quite astounding for a human). So computational beliefs at first glance are not straightforwardly captured as inferential beliefs (although more will be said on this).

Finally, what about testimonial beliefs? There is certainly an initial sense in which computational beliefs appear akin to testimonial beliefs. The app has used available data and processed it in a particular way such that it has reached a verdict relatively autonomously, and it will then inform us of this proposition. In this way, we're treating the app *as if* they are epistemic agents that we can rely upon via testimony. I do think this is also somewhat reflected in the way in which we talk about such apps, by applying terms that usually apply to epistemic agents, such as what the app *thinks* is the case, what it *says* is the case, or how it might be *tricked* or *fooled*. But this can only be the product of anthropomorphisation. The app is not an epistemic agent, it does not have beliefs in any ordinary sense of the term. Nor is it really able to communicate, if we understand communication as something which requires *intentions* of a particular kind.[9] For this reason, computational beliefs simply cannot be properly characterised as the product of a testimonial exchange.

A category that does capture computational beliefs is what Sosa (2006) calls *instrumental belief* i.e. belief formation via the use of an instrument. Instruments – calculators, thermometers, compasses – can provide deliverances, and we are able to form beliefs on the basis of those deliverances. Not all instruments perform this role, of course, and to this extent we can distinguish between epistemic instruments – calculators, telescopes, abacuses – and practical instruments – hammers, kettles, and bicycles. We can view computational beliefs as the result of employing an app as an epistemic instrument, in the same way that we employ an epistemic instrument when we use a thermometer to form a belief about the temperature. Clearly when we engage with such instruments, it is because we seek to find out something about the world and we have some awareness that the instrument can provide it. Sosa is clear that epistemic reliability is a necessary condition for the success of an instrumental belief. He states:

> A deliverance of a proposition by an instrument is epistemically reliable only if that proposition belongs to a field, and that instrument is so constituted and situated, that not easily would it then deliver any falsehood in that field (p. 117)

As stated in the previous section, these algorithms certainly possess reliability within their given field, and so it is clear that this condition is met in the case of computational beliefs. Sosa argues further that in using such instruments, we make manifest an assumption we hold regarding their

[9] This is particularly relevant if one holds the view, as Lackey (2006) does, that testimony is a case of forming a belief based upon a communicative act rather than the particular beliefs held by the testifier.

reliability. But he falls short of claiming that instrumental beliefs come about inferentially. Instead, he is merely claiming that instrumental beliefs require that the instrument be epistemically reliable, and that the agent holds an implicit assumption that this is so.[10] Nevertheless, I think it is plausible that successful computational beliefs are the product of a kind of inference that appeals to beliefs about the instrument in question. In the next section, I will say more about the inferential makeup of computational beliefs. In doing so, I will consider whether computational beliefs can be reduced to a form of inferential belief.

5. Reductionism about Computational Belief

As stated earlier, a great deal of the literature on testimony has focused on whether testimonial beliefs are, as a kind, reducible to more basic kinds of belief. Abstracting away from testimony in particular, the question of reductionism can be phrased in the following way: is a belief that is generated on the basis of a particular epistemic source provided a kind of justification distinctive of that epistemic source, or do further positive reasons have to be present in order to justify that belief. If one claims that a kind of belief is afforded some justification purely on the basis of its source, then one advocates anti-reductionism about that kind of belief. If one claims that there is no such justification, then one advocates anti-reductionism.

Reductionism about testimony has proven to be a controversial topic. For the purposes of this essay, understanding a few key considerations on either side of the debate will prove useful for thinking about reductionism in the case of computational beliefs. Reductionists have argued, on the one hand, that the idea that we have a default justification to believe what we are told would simply make justified testimonial belief too easy, given the clear possibility of error or misdirection on behalf of the testifier. For example, Fricker states:

> …does not mere logic, plus our commonsense knowledge of what kind of act an assertion is, and what other people are like, entail that we should not just believe whatever we are told, without critically assessing the speaker for trustworthiness? We know too much about human nature to want to trust anyone, let alone everyone, uncritically. (Fricker 1995, p. 400)

On this basis, they argue that some inferential procedure is required in order to justify a belief based on testimony. Such an inference will appeal to the trustworthiness or reliability either of the specific testifier (local reductionism) or of testimony in general (global reductionism).

Anti-reductionists, on the other hand, argue that it is not plausible that agents regularly do have the kind of positive reasons that reductionists would require. In the case of global reductionism, if there is even a true claim to be known about the general reliability of testimony, it seems doubtful that we could ever come to know such a claim given the severely limited sample of total number of testimonial exchanges we are exposed to (Coady 1994). And even in the case of local reductionism, the claim that we must constantly possess prior beliefs about our testifiers in order to hold justified testimonial beliefs is in danger of rendering a great number of our beliefs as unjustified.

I won't seek to make any positive claim about the nature of testimony here. Instead, I want to apply the question to computational beliefs: is it the case that a computational belief can be

[10] In the terms of section 5, Sosa does seem to endorse reductionism about instrumental beliefs. He contrasts instrumental beliefs to perceptual beliefs in the following way: '…we are default-justified in accepting the deliverances of our senses, but we need a rational basis for accepting the deliverances of our instruments' (p. 123).

justified (absent defeaters) merely in virtue of its status as a computational belief? I think the answer should be negative. First, notice that the supposed deficiencies of reductionism in the case of testimony do not apply here. It *is* plausibly the case that a given user could hold certain beliefs about the reliability of computational sources, and this could be at a more local or more global level. For instance, I might hold the belief that anything that any huge tech firm releases is going to be reliable, or I may hold a belief only about the products of Google in particular, or I may only hold a belief about the reliability of a given app.

Notice also that while it seems like the positive reason I require to trust an app will have to account for the reliability of that app, this positive reason can itself be formed in a number of ways. First, it might be the case that I form it on an inductive basis: I may use Google Maps a few times with no particular view as to its reliability, and consistently find it to produce the right answer, such that I eventually infer that the deliverances from Google Maps are reliable. Alternatively, it may be that I acquire a belief about the reliability of a given application via testimony. If Google advertises that their app is 98% accurate, then (provided I trust what they say) they are providing a testimonial basis to believe that the app is 98% accurate. Or if my trusted friend tells me that the app is extremely reliable, then, again, I form a belief about its reliability via testimony. Finally, one can form a belief about the reliability of a given app on the basis that it was designed to be reliable and it would not have been released if it were not so. Even this seems to me to be an adequate basis from which to reach the belief that the app is reliable in its deliverances, although such an inference would be on shakier grounds where one is less sure of the provenance of the app. But the important point here is that whereas we have reason to doubt whether we could plausibly be in a position to know the reliability of a given testifier or of testimony as whole, apps are comparatively more stable in their behaviour (their behaviour is algorithmic and is also not affected by intentions to mislead), and so we can plausibly form beliefs about their reliability through a variety of different means.[11]

A second reason in favour of reductionism for computational beliefs is that it better captures our intuitions in particular cases. Suppose that an app turns up on Bertrand's phone, and it claims that it will tell Bertrand not only whether someone has read one of his journal articles, but whether that person enjoyed the article as well. However, the app provides no information about how it works, nor does it claim to be accurate to any level, nor is it obviously made by a big tech firm that regularly produces reliable apps. It simply says things like 'Someone in California read your paper but was not particularly impressed!' The question is: in this situation, does Bertrand have a default justification in believing what the app says? I claim he does not, and to this extent anti-reductionism is implausible.

Finally, it could be argued that the positive reasons in favour of anti-reductionism about testimony are absent in the case of computational beliefs. For example, Burge (1993) has argued

[11] The above is not necessarily supposed to serve as a psychological description of how ordinary people come to form such beliefs. The question of reductionism is a question of the justification of our beliefs, not the causal story of how they were formed. It may well be the case that ordinary practice takes a few inferential shortcuts where possible, and this can at least be tolerated provided the justification for a given belief is properly based within a wider set of beliefs. It is for this reason that Schiffer states: 'Whether knowing p is based on knowing q, isn't about the actual movement of thought, the considerations one actually ponders; it's about the structure of beliefs that sustains one's conclusion.' (Schiffer 2003, p. 303, fn. 2). In that respect, reductionism about computational beliefs does not amount to the view that we infer in this way each time we use such an app. As such apps become increasingly part of our epistemic lives, it seems plausible that our reliance on them will skip this explicit inferential procedure.

that we have a default *a priori* prima facie entitlement to belief testimony because of a prior assumption that when propositions are presented as true, they derive from a rational source. Burge's argument is fascinating in its own right, and I don't hope to provide an evaluation of it here. Instead, I merely want to note that the argument does not straightforwardly transfer to the case of computational beliefs. Whereas in the case of testimony, truth-telling is arguably a core behaviour that one has to display to some extent in order to properly engage as a rational member of the community, the apps from which we receive such deliverances are under no such constraint simply because they are not rational members of the community. With this in mind, it doesn't seem like the *a priori* prima facie entitlement that Burge defends could apply to computational beliefs.

If all this is right, it seems plausible that computational beliefs do require further positive reason in order to be justified i.e. reductionism about computational beliefs is true. For many, the motivation behind debates concerning reductionism is the implicit assumption that if a kind of belief is reducible, then it does not warrant epistemological study. After all, if epistemologists can provide an account of the non-reductive kinds of belief, they will in indirectly provide some account of all reductive kinds of beliefs also. Notably, however, Goldberg (2006) has argued to the contrary in the case of testimony. He has argued that even if reductionism about testimony is true, testimonial beliefs still merit epistemological attention because they give rise to distinctive kinds of epistemic harms. In the next section, I will argue in the same way for computational beliefs. Although they do appear to be reducible to a kind of inferential, instrumental belief, they warrant their own epistemological attention because they have important epistemically normative consequences. In particular, they give rise to epistemic responsibility gaps.

6. Computational beliefs and epistemic responsibility

To summarise thus far, we have found the following: computational beliefs possess a range of epistemic properties usually associated with successful beliefs; they can be viewed as a form of instrumental belief; and that one requires further positive reason in order to trust computational beliefs (i.e. they are reducible to a particular kind of inferential process). In this section, I will outline a normative ramification of computational beliefs, that are particularly important to keep in mind when considering the fact computational beliefs are on the rise. I will argue that the rise in computational beliefs where previously beliefs would have had a testimonial source gives rise to a kind of responsibility gap, analogous to the moral responsibility gap argued for by Matthias (2004).

Matthias argues that in cases of learning algorithms making decisions that would otherwise be made by a human agent, a *responsibility gap* arises. His argument for this claim is fairly simple. In order to be held morally responsible for a decision or action, an agent must at least have control over that action. In the case of actions via instruments, an agent is morally responsible for the operation of that instrument only if they have control over the operation of the instrument. When a programmer encodes the operation of a computer, they have control over the operation of that computer, and so for that time at least, she is morally responsible for any morally significant outcomes associated with the computer. When we buy a computer, the computer company provides us with control by informing us of how it operates (via a user manual). At that point, we then become responsible for the operation of the computer and for any significant outcomes associated with it.

The problem with learning algorithms is that, due to their opacity, it is often the case that *no-one* is directly in control of the decisions they make. For any given output of a learning algorithm, it is not plausible to pin the responsibility of that particular output on a given individual. To illustrate this, consider the following parallel cases. In Darlington, officer Euan is given the job of deciding whether suspects should be released from custody, and in one case, he decides a suspect to be low risk and suitable for bail, when in fact the person is quite dangerous and subsequently reoffends. In nearby Hartlepool, where there is a much more enterprising police force, they have recruited a Harm Assessment Risk Tool (HART) in order to decide whether a given suspect should be released on bail.[12] The HART is a learning algorithm that takes into account years of previous police records and estimates whether a given suspect would be likely to reoffend if released, based on the information available about that suspect. Suppose that HART decides someone to be low risk and suitable for bail, when in fact the person is quite dangerous and subsequently reoffends. Now in both cases there are background responsibilities held by others related to the role. Darlington's senior officer who placed Euan in his role is responsible for having done so. Equally, Hartlepool's senior officer who employed HART for the role is responsible for having done so. But there is an important difference between the two cases. Officer Euan has moral responsibility over the particular decisions he makes. Having released someone that subsequently reoffended, Euan must justify his decision to prove that he has not done something wrong, otherwise he is due some level of blame. But in Hartlepool, there is no such agent that takes on the blame when a similarly harmful outcome is reached. There is no agent that takes the equivalent role of Euan, nor is it plausible that the senior officer can take the responsibility simply because they did not have control over that decision. The responsibility has simply gone, and this is the moral responsibility gap that Matthias is concerned with.

A similar responsibility gap arises in the epistemic realm. Consider two more cases based on Google Maps. In one case, I look to Google Maps to see whether there is traffic on the King's Road, and Google Maps tells me there is no jam when in fact there is, and so I am very late home. In the second case, instead of asking Google Maps, I ask a colleague who has only recently come into work whether there was a traffic jam along the King's Road. Despite the fact that there was and there still is, my colleague says there is no traffic jam, and so I am home late. In the second case, my colleague holds *epistemic responsibility as a testifier*, and the fact that he failed as a testifier on this occasion – whether it be through maliciousness or error – means that he is open to a particular form of blame. But in the Google Maps case, there is no-one that takes the equivalent responsibility for the outcome. Just as in the ethical scenarios, there are background responsibilities common to both cases. I am responsible for choosing to appeal to either Google Maps or to my colleague, given the reliability that both exhibit. But in addition to this background responsibility, my colleague has responsibility over his testimony for which there is no equivalent in the Google Maps case. This form of epistemic responsibility is captured well in the following quote from Goldberg (2006), who considers the difference between instrumental and testimonial belief:

> A rational being engages in the project of shaping its beliefs to fit the evidence it has. Because this project is to some degree under the being's own rational control, this shaping process can be done in better and worse ways (epistemologically speaking), in ways that are epistemically sanctioned, and in ways that are not. Consequently, the notion of epistemic responsibility finds a home here. The result is that, in relying on a rational being's testimony, one is relying on that being to have lived up to her relevant epistemic responsibilities. A

[12] Durham police force employ such a tool on an advisory basis: http://www.bbc.co.uk/news/technology-39857645

> merely reliable instrument, by contrast, operates according to the laws of nature. Because there is no rational control to speak of, the notion of epistemic responsibility has no home here. (Goldberg 2006, p. 136)

This highlights the differences in terms of epistemic responsibility between testimonial beliefs and instrumental beliefs – of which computational beliefs fall under. Yet this doesn't quite capture the distinctive normative properties of computational beliefs in particular.

When we form instrumental beliefs, we are relying on an inferential process that partly appeals to the operation of an instrument. When we do this, we may or may not be aware of how the instrument works. If we are aware of how the instrument works – for example, I know how the mercury comes to rise and fall in my thermometer – then the epistemic responsibility lies entirely with the believer. But we can also form instrumental beliefs using instruments the workings of which we do not understand – you might do this when you search for your GPS location, for example. Importantly in such cases, as mitigation for the fact that my inference relies on an instrument the workings of which I do not understand, I have something in the social epistemic realm that I can fall back on. That is, I am part of an epistemic community that contains members who do understand the workings of my instrument, and this is reflected by the fact that I could appeal to that epistemic community if my instrumental inferences seem to go awry. Indeed, there is an important sense in which, in using instruments that I do not understand, I am reliant upon my epistemic community. But computational beliefs depend upon autonomous learning algorithms, the opacity of which prevents any member or any set of members from understanding how exactly they operate. Because of this, when we form computational beliefs, we cannot properly rely on our epistemic community as mitigation for the fact that we do not understand on what basis our belief was formed. So not only do computational beliefs give rise to a kind of responsibility gap, but there is also the lack of an appeal to the epistemic community as kind of fall-back.

One further point speaks in favour of computational beliefs warranting epistemological interest. As stated earlier, computational beliefs are *on the rise*. In particular, there seems to be a general trend of computational beliefs replacing testimonial beliefs. Whereas before I would ask others for directions, now I ask a smart device; whereas before I would consult a translation dictionary, now I ask a machine translation service; whereas before we ask doctors for medical opinions on scans, now we ask image processing software. Considering that testimonial beliefs give rise to a distinctive structure of epistemic responsibility relations that is lacking in the case of computational beliefs, this rise in computational beliefs constitutes a shift in the epistemically normative relations that hold between agents, of which social epistemology has done so much to shed light on. We ought to investigate what this new picture of social epistemic relations is going to look like.

7. Conclusion

In this paper, I have introduced and argued for the legitimacy of computational beliefs. Computational beliefs are able to possess many of the epistemic features commonly associated with successful beliefs. They are best understood as a form of instrumental belief. Finally, although they are reducible to more basic forms of belief in the sense that there is not a distinctive form of justification associated with computational beliefs, they nevertheless warrant epistemological attention in their own right due to their normative implications.[13]

References

Burge, Tyler. 1993. "Content Preservation." *The Philosophical Review* 102 (4):457-488. doi: 10.2307/2185680.

Coady, C.A.J. . 1994. *Testimony: A Philosophical Study*. Oxford: Oxford University Press.

Fricker, Elizabeth. 1995. "Critical Notice." *Mind* 104 (414):393-411.

Goldberg, Sandford. 2006. "Reductionism and the Disitnctiveness of Testimonial Knowledge." In *The Epistemology of Testimony*, edited by Jennifer Lackey and Ernest Sosa. Oxford: Oxford University Press.

Goldman, Alvin. 1979. "What is Justified Belief?" In *Justification and Knowledge*, edited by G.S. Pappas, 1-25. Dordrecht: Reidel.

Hawthorne, John, and Jason Stanley. 2008. "Knowledge and Action." *The Journal of Philosophy* 105 (10):571-590.

Heersmink, Richard. 2018. "A Virtue Epistemology of the Internet: Search Engines, Intellectual Virtues, and Education." *Social Epistemology* 32 (1):1-12.

Kuruvilla, Jinsa, and K. Gunavathi. 2014. "Lung cancer classification using neural networks for CT images." *Computer Methods and Programs in Biomedicine* 113 (1):202-209. doi: https://doi.org/10.1016/j.cmpb.2013.10.011.

Lackey, Jennifer. 2006. "Learning from words." *Philosophy and Phenomenological Research* 73 (1):77-101.

Lackey, Jennifer. 2008. *Learning from Words: Testimony as a Source of Evidence*. Oxford: Oxford University Press.

Matthias, Andreas. 2004. "The Responsibility Gap: Ascribing Responsibility for the Actions of Learning Automata." *Ethics and Information Technology* 6:175-183.

Miller, Boaz, and Isaac Record. 2013. "Justified Belief in a Digital Age: On the Epistemic Implications of Secret Internet Technologies." *Episteme* 10 (2):117-134.

Miller, Boaz, and Isaac Record. 2017. "Responsible epistemic technologies: a social-epistemological analysis of autocompleted web search." *New Media & Society* 19 (12):1945-1963.

Owens, Andrew, Phillip Isola, Josh McDermott, Antonio Torralba, Edward H. Adelson, and William T. Freeman. 2015. "Visually Indicated Sounds." *eprint arXiv* 1512.08512.

Schiffer, Stephen. 2003. *The Things We Mean*. Oxford: Oxford University Press.

Simpson, Thomas W. 2012. "Evaluating Google as an Epistemic Tool." *Metaphilosophy* 43 (4):426-445.

Sosa, Ernest. 1999. "How Must Knowledge Be Modally Related to What Is Known?" *Philosophical Topics* 26 (1/2):373-384.

Sosa, Ernest. 2006. "Knowledge: Instrumental and Testimonial." In *The Epistemology of Testimony*, edited by Jennifer Lackey and Ernest Sosa. Oxford: Oxford University Press.

Vogel, Jonathan. 2007. "Subjunctivitis." *Philosophical Studies* 134 (1):73-88. doi: 10.1007/s11098-006-9013-8.

Zhou, Zhi-Hua, Yuan Jiang, Yu-Bin Yang, and Shi-Fu Chen. 2002. "Lung cancer cell identification based on artificial neural network ensembles." *Artificial Intelligence in Medicine* 24 (1):25-36. doi: https://doi.org/10.1016/S0933-3657(01)00094-X.

[13] This paper has been greatly improved by an audience of postgraduate students at the University of Reading, as well as two anonymous referees.