6-17-2021

Algorithms and Their Work: A Performativity Perspective

Mikael Gustavsson
University of Gothenburg, mikael.gustavsson@ait.gu.se

Jan Ljungberg
University of Gothenburg

ALGORITHMS AND THEIR WORK: A PERFORMATIVITY PERSPECTIVE

Research paper

Gustavsson, Mikael, University of Gothenburg, Gothenburg, Sweden, mikael.gustavsson@ait.gu.se

Ljungberg, Jan, University of Gothenburg, Gothenburg, Sweden, jan.ljungberg@ait.gu.se

Abstract

Algorithms are becoming increasingly prominent. More and more aspects of our everyday lives are being mediated, produced and directed by digital artefacts and connected systems which in turn are powered by algorithms. In this paper, we engage with algorithms and their performative aspects during development, implementation, and performances in the world. Based on theorizing, we develop an algorithmic perspective of performativity which centres on how algorithms evolve from initially being shaped to becoming those who shapes. Our proposed research opportunities address pressing conditions where the presented framework can prove beneficial as a conceptual device.

Keywords: Algorithms, Performativity, Machine Learning, Artificial Intelligence, Deep Learning.

1 Introduction

> *"The hope is that, in not too many years, human brains and computing machines will be coupled together very tightly, and that the resulting partnership will think as no human brain has ever thought and process data in a way not approached by the information-handling machines we know today."*
>
> 'Man-Computer Symbiosis' – J.C.R. Licklider (1960)

Increasingly our everyday lives are mediated and directed by digital artefacts and interconnected systems which in turn are powered by algorithms (Baskerville et al., 2020; Ågerfalk, 2020). Thus, in just a few years, the 'algorithm' has made a journey from having mainly occurred as an obscure, technical term in mathematical and computer science discourses, to now also being perceived as a cultural object embraced by social scientists, media scholars and journalists alike. Some authors claim we are entering an era of widespread algorithmic governance, where algorithms will have an increasing role in the exercise of power in society (Beer, 2017; Diakopoulos, 2014; Kitchin, 2017). While others point at and discuss the shaping nature of algorithms, where they automate, augment, sort, govern, and in different ways control our daily practices (Gillespie, 2012; Matzner, 2016; Trielli and Diakopoulos, 2019). Yet others inquire into algorithms' said obscurity, and inscrutability and investigate how these characteristics make it problematic to understand exactly what is at stake (de Laat, 2018; Introna, 2016; Ziewitz, 2016). Recent research within Information Systems (Baskerville et al. 2020; Ågerfalk, 2020; Lyytinen et al., 2020) recognize this diffusion of algorithms rendering a new breed of sociotechnical systems "where machines that learn join human learning and create original systemic capabilities" (Lyytinen et al., 2020, p. 1). Thus, these systems appear as hybrids, amalgams, where its members complementary and enhancing capacities contribute to the overall functioning of the system, and thus performs at a higher level than would they have done separately. Hence, these systems are acknowledged to contain members of different cognitive architectures, those of humans and computers (Lawrence, 2017). The operationalization of such systems takes these dissimilarities into account and leverage on them. At the core of this development, we find machine learning algorithms and more specifically deep neural nets. One example of such a system is Uber, where drivers are managed by machine learning algorithms: algorithms which handles everything from the actual matching of riders and drivers to the calculation of the price, the execution of the monetary transaction, with the subsequent assessment of the driver and rider through their respective app.

In this conceptual paper we complement research on how machines and humans may coexist in evermore complex systems (see Lyytinen et al., 2020; Lee, 2020). With a primary focus on machine learning algorithms, our aim is to explore how algorithms increasingly mediate our lives and thereby appear as vivid actors in a digital society. We specifically turn to the development and implementation of algorithms – thus, how they come into existence and later operate. Hence, we see a specific need to contemplate and discuss what the latest advances in technology have unleashed in terms of qualities of compute, and what these advances empowers. The recent breakthroughs in large-scale operations (e.g., cloud platforms), the accelerated powers of computing, and the possibility to collect and store vast amounts of data, enables capabilities of different kinds (Smith and Browne, 2019). Hence, since an underlying architecture does not dictate how different types of 'intelligent' behaviour emerges, but merely specifies what is possible to operationalize or not, there is a need in understanding how these architectural conditions enables different characteristics of compute: One where algorithms is produced by humans operating in the realm of logic, and one where algorithms are derived by computers that learns, operating in the realm of probability (Cantwell Smith, 2019). This shift must be recognized and made comprehensible in order to be able to evaluate and assess algorithms performances in the world, under which circumstances and conditions they excel and when they might fail. While intuitions have been built with respect to algorithms developed in the former paradigm (e.g. algorithmic and computational thinking), same intuitions applied with regards to machine learning algorithms can be illusory and hence deceptive. Thus, there is a pressing need, we argue, to be able to reason and discuss on an abstract, systems level how algorithmic assemblages operates and performs. In our attempt to realize this, we find the concept of performativity (e.g. Barad, 2003; Butler, 1993; Pennycook, 2004) as an important prerequisite for being able to shed light on and theorize how algorithms – during development – are shaped but when put to operate in the world, becomes the ones that shapes. Accordingly, it is necessary to pay attention to practices involved in developing and designing algorithms – along with an appreciation of the mechanics by which they operate – to be able to say something about issues such as autonomy, decision-making, and prediction. Hence, to be able to "understand what assemblages of people and machines should assigned what kind of tasks, we need to understand [...] what kind of work require what kind of capacity" (Cantwell Smith, 2019, p. xiii). We therefore propose a framework which acknowledges algorithms not as objective artefacts, but as creations which develops and operate through time. Given these prerequisites, the question that guide this paper is: *How can we understand algorithms performances in the world*?

In what follows, we first position the paper within related research on algorithms. We then present the reader to a conceptual framework of algorithmic performativity. The subsequent section discusses the framework in relation to three aspects of algorithms: agency, reliability, and complexity. The last section concludes the paper.

2 Theoretical Background

Traditionally, an algorithm is described as a mathematical, computational method: "[A] series of steps undertaken in order to solve a particular problem or accomplish a defined outcome" (Diakopoulos, 2014, p. 3). Hence, the algorithm appears as an abstraction, free of the material constraints which is embodied in its implementation. Accordingly, and due to the chosen method of implementation, an algorithm can be materialized in different ways (Dourish, 2016). Thus, an algorithm is a model of what a machine can do, when said model is translated into executable computer code. Furthermore, the nature of this execution can differ, dependent on the type of computer architecture, e.g., storage, memory, processor etcetera (Dourish, 2016; Lee, 2017). But as the algorithm has gained increased value in various discursive systems – e.g., security (Amoore and Raley, 2017); law (Chesney and Citron, 2018); social movements (Milan, 2015); ethics (Sandvig et al. 2016); politics (Wooley and Howard, 2016) – it has at the same time lost much of its general explanatory power as being a logical, step-by-step formula. Hence, the 'algorithm' has become a notion with many faces and thus "emerges in a complex interplay of social practices, material properties, discourses, mathematical abstractions, and code" (Matzner, 2019, p. 4). Moreover, the notion of algorithm – its semantics and connotations – tends to shift as we move along its lifespan (Seaver, 2017; 2019). When describing and discussing algorithms in the early development phases, one tends to talk about them as abstract mathematical, computational models (Rieder, 2017) while in the later stages they are rather perceived as complex, sociotechnical systems of which the actual algorithm only constitutes a smaller part (Dourish, 2016; Gillespie, 2012). Therefore, depending on context, the notion of algorithm may assume an artefact or a set of artefacts from the early developed creations in mathematics and computer code, to the later implemented and materialized instances within infrastructures and wider sociotechnical systems making their imprints in the world. Thus, the algorithm as a concept has become both elusive and extensive in discourse (Mittelstadt et al., 2016). In this paper, we make use of this somewhat faded conceptualization when referring to the algorithm sometimes as an individual instance (e.g., machine learning algorithm), and sometimes a placeholder for a wider, computational system (e.g., 'the Google algorithm', 'the Facebook algorithm'). Hence, we do not delineate algorithms in relation to scale but instead to functionality. Thus, we are interested in how different computational theories, approaches and methods give rise to different types of algorithms; algorithms of various qualities and kinds, operating in accordance with different epistemic and ontological conditions. And it is these conditions, we argue, that must be exposed, understood, and discussed in order to be able to make informed and sound decisions regarding algorithms – i.e., where to use them, how much autonomy we can allow them, and finally to what extent we can actually trust them. Consequently, in this paper, algorithms are delineated into two, broader categories: those of logic and learning.

2.1 From Logic to Learning

In the *logic* paradigm, an algorithm is a step-by-step function which controls and prescribes what and what not to do under given circumstances (Goffey, 2008). An algorithm is designed and based on previous gathered knowledge, in a sense: 'what we know we can program'. Thus, "every step of the procedure is explicitly specified by its human designers and written down in a general-purpose programming language such as Python or C++" (Kearns and Roth, 2019, p. 6). Hence, the algorithm arises as an orchestrator that parses and acts upon digital representations (data structures) of the 'world' manifested in code. The ontological worldview of this paradigm is symbolic, discrete and conceptually well-defined (Cantwell Smith, 2019). Hidden in this notion of 'algorithm' is the conception of an artefact being developed and crafted by man and therefore comprehensible and understandable (Dourish, 2016): Although it is a computer which performs actions through execution of code, humans have prescribed the states of action during design. Thus, when confronted with problems due to the actions of algorithms, we sense a programmer lurking in the background. A programmer who clearly could not envision the specific situation per se, or did a miscalculation, or made a misconception; hence, we attribute the issue at hand – or the 'bug' – to the shortcoming of human developers and is counting on them to solve the situation at hand through deliverance of a patch or version update of the software. An update which requires the developers to refactor, restructure, and/or redesign existing code (Barr, 2018). In this sense the algorithm ambiguously appears as a deterministic yet uncertain entity with a consistent behaviour.

Today, the notion of algorithm has mainly come to refer to artificial intelligence and machine learning, which in this paper is termed the *learning* paradigm. In the learning paradigm, the notion of algorithm is perceived in a quite different way. As an example, algorithms could be shaped to fit a domain-specific purpose through a so-called supervised learning process[1] (Ford, 2018) calibrated by massive amounts of labelled but unstructured data (Burrell, 2016). Instead of being deterministic these types of algorithms are probabilistic (Domingos, 2012). During a learning process the algorithms are trained to infer a specific outcome from a specific income – is it a dog or not? This means that these algorithms are not designed and conceptualized by humans in a comprehensible step-by-step manner as in the logic paradigm; instead, these types of algorithms are shaped and moulded through being exposed to a massive flow of training data. E.g., if a learning algorithm is to be used in image recognition scenario, the final computer code will not operate in accordance with how humans would delineate images; instead, the algorithm will 'understand' the image as a series of pixels that, when they occur in the right

[1] Thus, we acknowledge that there exist various types of learning mechanisms, e.g. supervised-, semi-supervised, unsupervised-, and reinforcement learning (see section 4.1).

order, equals the sought-after image. The human concept of dog - 'sharp teeth', 'barking', 'have fur' - is non heard of in the realm of learning. Here, the algorithmic logic operates in accordance with pattern recognition of pixels and thereby a certain probability that it is a 'dog'. In comparison with the logic paradigm, an important ontological difference is here to be noted. An ontological difference which is key to understand how these types of algorithms ‘make sense’ of the world: “The classical assumption of a discrete, object-based ‘formal’ ontology is not a prerequisite for machine learning […] On the contrary, the success of ML systems, particularly on perception tasks, suggest a different picture: that the world is a plenum of unbelievable richness” (Cantwell Smith, 2019, p. 66). Hence, we as humans can be deceived in believing that these types of algorithms recognize a ‘dog’, when in fact they only confirm a representational pattern of pixels (cf. Smith, 2020). Accordingly, a learning algorithm predict and categorize based on probability, but are not able to reason and judge based on knowledge (Cantwell Smith, 2019; Dourish, 2016). Thus, these approaches are largely empirical to their nature, and thus not guided by theory (cf. 'competence without comprehension'; Dennett, 2017).

2.2 Algorithms and their implementations

Algorithms reach or impact do not solely rely on their mathematical and logical characteristics. What an algorithm can achieve is always relative to the computer systems which embodies them. As Dourish (2017, p. 213) observes: “The same algorithm, implemented on different computers or supported by different technical infrastructures, has quite different capacities. Mathematically, what it can achieve is the same, but practically, a new architecture or implementation of an algorithm can bring new possibilities into view and new achievements into the realm of the possible.”

One example of the importance of how the effects of algorithms have developed with regards to how they have been implemented is the success of deep learning (Clarke, 2019). Deep learning mimics the architecture of the brain. It is based on the so-called philosophy of connectionism where – in analogy with the neurons in the brain – a single feature of a machine learning model is not considered intelligent, while a large population of said features, acting together, can exhibit intelligent behaviour (Goodfellow et al., 2016). The model that forms the basis of the algorithm, designed to illustrate fundamental properties of an intelligent system, was created in the late 1950s when Rosenblatt (1958) put forward the theory of the perceptron where he explained: “In an environment of random stimuli, a system consisting of randomly connected units [...] can learn to associate specific responses to specific stimuli” (p. 405). Although Rosenblatt envisioned the possibilities of the perceptron, the model failed to deliver due to the problem of scaling. The model was too shallow to be able to produce interesting results. It would take until the 2010s before neural networks would seriously return to the scene (Ford, 2018). First in speech recognition (Mohamed et al., 2009) and then in computer vision (Krizhevsky et al., 2012). As the computer power significantly increased over the years, the model could now scale to a level where the inherent benefits of the architecture revealed themselves. Hence, the functionality and reach of an algorithm is not only hidden in its logical formula, or in the code that implements it, but also in the hardware of its specific instantiation. Our perception, experience and even the actual outcome of an algorithm can vary as its manifestation evolves due to changes in the underlying technological infrastructure (Dourish, 2016; Tan and Le, 2019). Accordingly, “[a]lgorithms cannot be divorced from the conditions under which they are developed and deployed” (Kitchin, 2017, p. 10).

2.3 Algorithms and data

Most of the algorithms in the learning paradigm of today falls under the sub-paradigm of supervised learning. Supervised learning entails providing a machine thoroughly labelled training input data to feed a learning algorithm (Ford, 2018). Hence, this paradigm presupposes two things: (1) Massive amounts of training data, and (2) that the data is correctly annotated so that the algorithm is be able to differentiate between objects in the training set (Goodfellow et. al 2016). These two things, and how corporations and firms get hold on them, has bearings on humans and the social fabric at large. For example, in the world of social media, when people are posting, liking, commenting they are in fact feeding the platforms labelled data on behaviour and preferences. From this data, platforms and data brokers can build profiles of users. Thus, digital traces are used as pieces to build an online persona which in-turn can be used to target, for example, advertisement (Martzner, 2016). But it is not primarily content that is key, but metadata. Data about data is ever more important in trying to envisage online personas – their relationships to others, what they like/dislike, where they are located, messages sent, webpages visited etcetera. Thus, a reciprocal relationship between users and platforms emerges: Through actions, transactions, and overall participations users feed digital platforms and their algorithms with data. Data which in-turn is algorithmically employed to further analyse, categorize and predict (Mitchell, 2019). In this perspective, users of platforms engage in algorithmic work by on the one hand producing training data, but also more or less helping in annotating and labelling it (Ekbia and Nardi, 2017).

2.4 Towards a performativity perspective

Originally, performativity refers to actions performed through language, i.e. utterances that produce what they name. Hence, language is not only used to describe and represent the world, but also to do things in the world (Austin, 1962; Searle, 1969). Performativity has since been introduced in language-, discourse-, gender- and entrepreneurship research (Barad, 2003; Butler, 1993; Garud et al. 2018; Pennycook, 2004), and more recently in relation to computing (Mackenzie, 2005; Matzner, 2016). Performatives was applied in the IS-field in the late eighties, and nineties, when social and communicative aspects of IT became more stressed. In this process speech act theory came to play an important role (Auramäki et al.,1988; Ljungberg and Holm, 1996; Winograd and Flores, 1986), and is now taken up again in the digital first discourse (Baskerville et al., 2020). Thus, the notion has evolved into a broader view of discursive performativity and applied to other discursively mediated practices.

Butler describes performativity as the "reiterative power of discourse [which] produce the phenomena that it regulates and constraints" (Butler, 1993, p. xii). Hence, performativity "can be understood as the way in which we perform acts of identity as an ongoing series of social and cultural performances rather than as the expression of a prior identity" (Pennycook, 2004, p. 8). The key thing to understand is that 'the performative' not only constitutes identity, but also constitutes what it is purported to be, and thus appears as a circular, self-producing activity. Over time, these repeated acts bring forward a material 'sediment' - a taken-for-granted knowledge - which gives the appearance of an underlying, and objective 'reality'. Performativity then, is not only linked to the formation and representation of a subject but also, and more, to the production and becoming of the same (Barad, 2003; Pennycook, 2004). "Thus we need to shift from a logic of *causality* (which assumes pre-existing beings as the source/origins of action) to *performativity* (which treats becoming as a radical ontological openness)" (Introna, 2013, p. 336; italics in original). Hence, performativity can be understood as a way to explain and understand how the world is being made and reconfigured through material-discursive practices (Barad, 2003). Accordingly, technical performance and digital materiality is always existing within a discursive system (Bazerman, 1998; 1999; Dourish, 2017; Lee, 2017; Mackenzie, 2005). This extended view of performatives, a kind of discursive performativity, is especially suited for applying on the diffusion of machine learning algorithms. Algorithms which lurk in the background and determines the flow of information – dynamically filtering content, deciding what to show and what to suppress – acting as invisible yet powerful gatekeepers and watchdogs (Beer, 2017). Algorithms which shape different domains of everyday life through their performances: i.e. finding friends, or lovers, matching taxi trips, arrange hotel bookings, cater food deliveries, monitoring suspects etcetera.

3 Algorithmic Performativity

In this paper, we propose a performativity perspective[2] on algorithms where "[p]erformativity implies that digital technologies operate with some level of autonomy" (Seidel et al., 2020, p.127). Thus, a perspective which rests on the assumption of software (e.g. computer code) being a technology of simulation; hence, a malleable technology which bears the capacity to be shaped, and the ability to subsequently shape, in various ways (Galloway, 2006). In order to link the concept of algorithm to the notion of performativity more strongly, we suggest an amalgam of three aspects as a vehicle for thought:

[2] Although we focus on learning algorithms, we find it suitable to contrast with algorithms created in the logical paradigm. Thus, in our conceptual framework, we compare how performativity plays out differently depending on design process adopted. Hence, the very point of the paper is to demonstrate which mechanisms give learning algorithms their performative properties, and this is most easily done by juxtaposing them with how traditional algorithms have been developed and shaped.

(1) algorithms exists as material instances of compiled, executable computer code which resides on hard drives or other type of physical mediums; this perspective highlights the important notion that “code exists first and foremost as commands issued to a machine” (Galloway, 2006, p. 326); (2) algorithms execute and operates within a 'linguistic', symbolic realm, where they continuously “renders objects, events and relations into communicable signs” (Mackenzie and Vurdubakis, 2011, p. 4); (3) algorithms perform and thereby influence the very same reality in which they are embedded (Beer, 2017; Kitchin, 2017; Milan, 2015). The amalgam puts forward a performativity perspective where algorithms appear as ‘machines’ operating through ‘language’ and ‘symbols’, making imprints in the world.

We suggest algorithmic performativity unfolding during two phases (see Table 1). First, algorithms get shaped and tailored during design and development; this phase unfolds differently depending on the paradigm, hence our decision to handle them as separate cases (*symbolic* and *learning*, see Table 1). Secondly, the algorithm is implemented and put into action and thereby becomes a performative instance of code, running on specific pieces of hardware, and thus influences and make imprints in the domain it is set to operate within. By adopting a performative perspective, the algorithm is perceived not as a static, but as a malleable yet formative creation: “The use of phrases ‘the Google algorithm’ and ‘the Facebook algorithm’ should not fool us into thinking that our objects are, deterministic black boxes […]” (Seaver, 2019, p. 419).

Phase	Domain	Description	Performative aspects
One – Logic	Development	Designed by humans; algorithms modelled based on knowledge; algorithms created as workflows.	Performativity pivots around the designers and developer’s contribution in strengthening or weakening stereotypes, cultures and discourses through code.
One - *Learning*	Development	Algorithms are shaped (‘designed’) by data during training; built on probabilistic foundations.	Open to learning about the world through patterns in data; vulnerable to insufficient, distorted, or false data will shape them into skewed models; data is hereby becoming performative
Two	Deployed; ‘Put in use’	Algorithms embody certain worldviews (by design) and seeks to organize the 'world' accordingly; when put into work, the shaped algorithm is becoming the one that shapes.	Algorithms performs in and influence the very same reality in which they are embedded; performativity can play out differently depending on context, types of data and algorithmic design.

Table 1: Conceptual Framework

3.1 Phase One: Shaping algorithms

In the *logic* paradigm the algorithms are designed and developed by humans. How an algorithm will be modelled is based on the knowledge of the specific problem at hand and any pre-given data structure that are about to be approached and acted upon. In other words, the data which is to be operated upon is well-known to the designer in the sense of predefined categories, semantic comprehension and structure. Algorithms are created as workflows and runs, step-by-step, calculating decisions based on input data (e.g. If…Then…Else) (Kitchin, 2017) and sometimes executing specific commands at given points in time. If a logical rule is missing, or the program misbehaves, it is likely that humans has not been able to foresee a specific situation. This hiccup will either result in a misleading result, a runtime error, or a crash in the program; either way the error occurred will be treated as a 'bug' and probably be attributed “the human factor”. Hence, in this paradigm, designers and programmers envision models based on problems in the wild, to develop algorithms which handles and solves said problem. Here, the developer/designer works and “operates within the context of a wider system (or discourse) that significantly shapes the designer's contribution” (Kallinikos, 2002, p. 289). The intelligence of the algorithm is thereafter translated into computer code. A translation which can prove problematic and in need of “a great deal of expertise, judgement, choice and constraints” (Kitchin, 2017, p. 10) In this paradigm, performativity therefore pivots around the designers and developers and their contribution in strengthening or weakening stereotypes, cultures and discourses through their choices and practices iteratively inscribed in code during development (Lee, 2017; Mackenzie, 2005). The discursive systems (Bazerman, 1998) that the designer resides and operates within will be reflected in their approaches to problem-solving; on how they define situations as problems; on how they reason about what can be considered a good outcome or not; on the way they present or hide data; on how they incentivize use etcetera (Barr, 2018). Over reiterations of tests and evaluations, the algorithm is moulded to primarily fit the requirements which describes how it is intended to work; but also, and more subtle, to operate in accordance with the 'algorithm' as the developer (residing in a specific discursive system) imagines, and continuously reimagines it.

In the *learning* paradigm, the wanted outcome of the algorithm is in a way dictated by humans, while the inner workings of the algorithm are shaped by data during training (LeCun et al., 2015; Lee, 2018). Here, the primary focus in reaching success is to gain access to huge amounts of data and massive computing power. Data for training, and computing power for the needed depth of the algorithms. Through their ability to learn patterns – and further being able to generalize these learnings onto future data feeds – these algorithms are shaped into pattern recognition machines (Smith, 2020; Domingos, 2012; Goodfellow et al., 2016): “This data-driven process is how we get algorithms for more human-like tasks, such as face recognition, language translation, and lots of other prediction problems” (Kearns and Roth, 2019, p, 6). Three recent technological advancements have made the diffusion of the learning paradigm possible. First, computing power has progressed to a level where the massive numbers of calculation needed for these types of models to scale, is met. Second, the big cloud platforms of today make large amounts of these powers accessible to developers and firms without the need for them to make capital investments in massive amount of hardware otherwise needed. Third, the explosion of digital data has made possible to build vastly larger datasets, in order to train these machine learning systems (Smith and Browne, 2019). Thus, in this paradigm, data is both an outcome as well as a 'designer' of the algorithm as such. During training, the algorithm compares a given output with a wanted one. If the result is not correct, the system changes itself (learns) by adjusting its internal parameters (called weights) to better suit the wanted result. This process is supervised by an algorithm (backpropagation) (LeCun et al., 2015). Over time, this training process results in a model that is ready to take on and classify new, unlabelled data which it has never encountered before (Burrell, 2016). Hence, data is becoming performative since the training sessions needed for tuning the algorithms, and make them learn, could be seen as “repeated acts within a highly rigid regulatory frame that congeal over time to produce the appearance of substance” (Pennycook, 2004, p.16). Herewith the algorithms, to some extent, is exposed to the same type of performativity that we as humans are in discourses of our everyday life; they will process, and thus be shaped by, the data that is given to them (cf. Clarke, 2019): “The abductive logics of many of these families of algorithms contrast with deductive reasoning [present in the logic paradigm] so that they are closer to experimental processes of learning and verifying through the available data” (Amoore and Raley, 2017, p. 6). Consequently, these types of algorithms are open to learning about the world through patterns in data (Lawrence, 2017; Smith, 2020). Accordingly, they are also vulnerable to the fact that insufficient, distorted, or false data will shape them into skewed models of the reality they are set to manifest (Clarke, 2019). Thus, the process of generating a machine learning model depends on two things: An objective function (e.g., what to optimize for, a goal), and a massive data set to train on. The objective function becomes the thing that the algorithm, during training, is trying to excel at. And the end product of this shaping process is manifested in a model adapted to do its job in a specific domain. A domain which is intrinsically defined in relation to the data on which the algorithm has been trained. Although the model might appear as effective and potent, it can nevertheless – when put into work – be perceived as flawed and problematic for example in relation to fairness. Hence, the algorithm can, with regards to its objective function, have been trained to optimize in ways which later turns out to compromise certain culturally accepted norms and assumptions: “In the era of data and machine learning, society will have to accept, and make decisions about, trade-offs between how fair models are and how accurate they are” (Kearns and Roth, 2019, p. 72).

3.2 Phase Two: Algorithms which shapes

Thus, a fundamental theoretical implication of adopting a performative approach is that algorithms and their representations are not just descriptions of something that exists on the 'outside' of reality but are constituent parts of it. Put another way, algorithms do not simply represent, they perform and thereby influence the very same reality in which they are embedded (Beer, 2017; Kitchin, 2017; Milan, 2015). They act as curators on social media platforms; they recognize faces and act on other types of biometric inputs; they buy and sell stocks; they fly planes and (maybe) soon drive cars. Hereby, algorithms are not just 'recipes' who prescribes and set boundaries of what can and cannot be done. Through their performances they also contribute to shape society in accordance with how they operate. Performativity is thereby related to the algorithms and their workings in the world. How they affect people and discourses through decisions, recommendations and categorizations. Thus, algorithms do not only process and provide information, but they also construct information by sorting and classifying in accordance with specific worldviews (Mackenzie, 2005). Put differently, algorithms embody and describes how specific constellations understands and thereby seeks to organize their 'world': "[T]he myriad of 'clicks' that regulate our daily lives, are all inspired by algorithmic models. The logic of numeric functions enters the practical world, often unseen, and firmly takes root in everyday life and our consciousness" (Totaro and Ninno, 2014, p. 30).

Thus, algorithmic performativity in this phase depends on the design of the algorithms and how they are implemented. When put into work, the shaped algorithm is now becoming the one that shapes. To visualize, we give two hypothetical examples of how data, processed by implemented algorithms, materializes as performances in the world. Imagine two different types of datasets: one non-man made (weather) and one man-made (twitter). Consider the weather dataset as input to an algorithm which predicts weather based on previous weather. The output of that algorithm will not change the actual weather per se, but the result will colour how meteorologists and media talk about and report the coming weather, and thereby how people prepare themselves for the same. On the other hand, consider twitter data as input: Here the twitter algorithm(s) massage the flows of data on the platform, and presents what is trending depending on the algorithms predefined ways of categorizing and presenting data. These trends, in turn, can steer what people talk about and start to interest themselves with, and by that strengthening phenomenon in discourse. These examples highlight how performativity can play out differently depending on context, types of data and algorithmic design. Hence, the first example will inform us about potential conditions coming ahead, while the other has the potential to form us - our thoughts and worldviews - by presenting certain types of information, while hiding others.

Consequently, due to the capacity to learn, act and react in narrow domains, algorithms (and primarily deep learning algorithms) have the possibility to execute in decision making by extracting patterns through vast amounts of data (Domingos, 2012; Goodfellow et al., 2016). E.g., as gatekeepers and watchdogs of social media platforms, algorithms curate and steer the flow of information – highlight some parts, while hiding others (Gillespie, 2012). "The so-called 'user behaviour' changes as new practices emerge, as different platforms become more or less popular, and perhaps above all, as predictive models act as part of platforms in the world" (Mackenzie, 2015, p. 442). In this, algorithms appear as cocreators of the 'world'. Not only do they orchestrate much of the information flow happening on social media platforms and others alike, but they also do invite for some types of creation and use, while precluding others (Kallinikos et al., 2013). Through automated chains of algorithms – e.g., Google search; Facebook newsfeed; Uber app – we are thus formed to attain certain practices.

4 Discussion

In public discourse "[a]lgorithms per se are supposed to be strictly rational concerns, marrying the certainties of mathematics with the objectivity of technology" (Seaver, 2019, p. 412). Seaver himself disputes and argues against this statement in his essay "Knowing Algorithms", and our previous discussion on algorithmic performativity sympathizes extremely well with this questioning on the perceived nature of algorithms. As the conceptual framework explained, algorithms are not objective creations. They do carry traces – norms, opinions, values – of their creators, let it be developers or the data that shapes them. Furthermore, the behaviour and efficiency of an algorithm may differ depending on the implementation along with the quality of data and force and scale of underlying infrastructures and related systems. Adding to the fact that algorithms are becoming increasingly sophisticated, potent and entangled they now – through 'intelligent' IoT artefacts – also makes performances in physical reality. Thus, if we perceive algorithms – and their representations in code – as temporal manifestations and not fixed creations we acknowledge them as vibrant and vivid actants doing their work in the world. By recognize them as co-creators and mediators of the 'world', we can appreciate and understand their wider implications: How they during development become performatively shaped, but when implemented and put into work performatively shapes.

Algorithms are beginning to affect us in the physical domain as well as the digital, and in ways we hardly can comprehend. Undoubtedly, machine learning algorithms contribute to novel avenues of innovation both in business as in society at large. However, there is a grinding concern that this force of compute could be misdirected (unintentionally) or misused (intentionally) and thus strike back in unpredictable ways (e.g. Mitchell, 2019; Clarke, 2019; Cantwell Smith, 2019; Zittrain, 2019). Hence, issues surfacing with regards to these families of algorithms are as much political and societal, as they are technical. Thus, discussions on topics such as autonomy, reliability and vulnerability are becoming increasingly important (see Schneier, 2018; Mitchell, 2019; Smith and Browne, 2019). Discussions where policy makers in society, organizations, and perhaps above all the large digital platform companies, needs to find a way forward. A way which does not hinder and stifles further innovation but protects citizens and the society at large against threats and attacks (Brundage et al., 2018). E.g., can we, on a global level, agree upon algorithmic ethics and apply regulations when it comes to AI algorithms (Smith and Browne, 2019; Sandvig et al., 2016)? If so, in which of the phases will regulations be needed/required? Is it during the first phase to impede specific designs and/or models? Or is it in the second phase, regulating how connections between implemented algorithms and wider sociotechnical systems can be made?

In what follows we engage in discussions on three aspects of algorithms we find increasingly pressing and where the framework of algorithmic performativity can be of help: Agency, reliability, and complexity.

4.1 Agency

'Agency' carries a wide variety of connotations: “It may encompass actions and the freedom to choose those actions; intentionality, will and power; causality, consequences, and outcomes (which may be intended or unintended); and decision making” (Rose and Truex, 2000, p. 372). In relation to algorithms, the notion of 'agency' has become somewhat ambiguous and problematic. We argue that the inherent capabilities of digital technology to change, shape and transform has contributed to the situation (e.g. Zittrain, 2008; Yoo et al., 2010; Henfridsson et al., 2018). Given that digital technology is in constant flux (Kallinikos et al., 2013) not only makes the landscape of the technological hard to predict, but more treacherously, continuous technological shifts can contribute to discursive rigidity through a “presumed ‘constancy’ of technology over time” (Faraj and Azad, 2012, p. 244). Hence, dominant discourses tend to hold established concepts hostage. We imagine that the notion of ‘agency’ in relation to algorithms somewhat suffers from such a phenomenon, as ‘artificial intelligence’ has evolved from being understood primarily as expert logic system of sorts, to now being equated with learning algorithms. This shift in conceptualization is partly a pure technological shift, moving from logical, deductively produced algorithms to inductive, data-driven, learning based approaches. But also, and maybe more subtle, a semantic shift on what AI really is, can do, and could come to be; a shift which divides the research community where now strong, yet shared opinions exist on the matter (cf. Ford, 2018; Brockman, 2020).

However, given recent breakthroughs, we stand at a brink where algorithms can – through reinforcement learning (Ford, 2018) – learn, tabula-rasa, without any previous domain specific knowledge or data, and achieve superhuman capacity in specific domains just within hours (Silver et al., 2018). Also, algorithms are starting to excel in unsupervised learning (Ford, 2018) and perform “a surprising amount of task without the need of explicit supervision” (Radford et al., 2019, p. 10). Such breakthroughs speak of digital artifacts, driven by algorithms, independently drawing inferences, taking actions, making decisions and thus operate with a much greater degree of autonomy than ever before (cf. Andersen et al., 2016; Ågerfalk, 2020). Thus, as the discourse on artificial intelligence progresses, 'agency' in relation to 'algorithm', tends to become a disputed concept as it tries to encompass machine learning (deep learning) algorithms, their perceived intelligent capacities, and further operations in the world. In this sense, it becomes a deceptive conceptualization since it suggests digital technology acting as a form of sentient being. Supposedly, this chimera stems from a perception that certain types of digital technology seem to handle situations and problems on their own, in an increasingly adaptable fashion. Hence, although digital technology is developed and designed for specific purposes, it like no other technology remains malleable and adaptable even after putting in use (Henfridsson et al., 2018). Consequently, digital technology can be designed and developed in accordance with an initial idea, but be further integrated, combined, and modified in various other ways (Zittrain, 2008; Yoo et al., 2012). Hereby, the open-ended nature of digital artefacts, with a subsequent obscure trajectory of future use, makes them somewhat dubious to understand (Kallinikos et al., 2013; Ekbia, 2009). The characteristic of unpredictability in *becoming* can make them appear as actants who operate autonomously, and even more so with regards to artefacts operating in the learning paradigm. Hence, in narrow domains, these artefacts – driven by machine learning algorithms – gain a kind of independent, adaptable behaviour. But behaviour says little about inner workings: "The algorithms we develop don't have a sentient nature, if we were to characterise them according to the dual process model of cognition, they are data-driven, input-output. They see then do" (Lawrence, 2017, p. 8). Hence, to judge only by behaviour is to treat the phenomenon at hand as a black box. Here, we align with Rose and Truex (2000) and their notion of 'perceived autonomy'; that 'autonomy' is a function which depends on how an observer approach a phenomenon as an object of study. On the contrary, machine learning algorithms and their performance in the world can emerge as a limiting factor on human agency, and the perceived autonomy of people in society (Lawrence, 2017). Through sorting and categorizing algorithms can, for example, herd people into various digital spaces based on personal as well as aggregated information. Hence, as people get increasingly "known" by algorithmic systems, they become ever more shepherded by them: "As a result of emergent artefact autonomy, humanity is in the process of delegating not to humans, but to human inventions. This gives rise to uncertainties whose nature is distinctly different from prior and well-trodden paths of human and organisational practice" (Clarke, 2019, p. 427).

Hence, the notion of 'agency' is value-laden and may open for controversies and discussions regarding for example whether (and then when) algorithms will reach human capacity in generalization or not (see Bostrom, 2014; Brockman, 2020; Ford, 2018). Although these discussions are important in preparing for a possible future technological breakthrough, we must at the same time focus on areas where today's technology make impressions and affects organisations and societies in tangible ways. Thus, since machine learning algorithms already do perform and hence must be understood as actors in the world, we need ways to build intuitions on how they operate and make their performances comprehensible without the risk of being caught in the crossfire of philosophical discussions. Here, the notion of performativity can prove to be of good help in trying to theorize and explain autonomous algorithmic behaviour: Is it during design and development? Or does it simply emerge through the algorithm's performances in the world as a member of increasingly complex sociotechnical systems? Or maybe as an amalgam of the two?

4.2 Reliability

Can we trust in machine learning algorithms and their performances? Or maybe a bit more appropriate: What are these networks really learning? If we are about to rely on these artefacts it is of utmost important that we can build intuitions on how they operate. The very fact that learning algorithms have become so powerful and potent with many almost unimaginable results, can contribute to us anthropomorphising their functions. It is often said that these algorithms work in similar ways as the brain, but perhaps this analogy is sometimes more problematic than we suspect and thus creates unrealistic expectations? It may lead us to believe that machine learning algorithms operate according to a human ontology of how we categorize the 'world' and subsequently build increasingly abstract concepts. Hence, we may attribute to them qualities such as judgment and common sense (Cantwell Smith, 2019). But the analogy with the brain is to be interpreted as a sign of similarity between architectures rather than functions. That is, the neural network in its structure is inspired by how the brain is structured. How then these different architectures contribute to functions such as learning have proven to differ markedly (Lawrence, 2017; Zittrain, 2019). As previously pointed out in the framework of how learning algorithms are shaped, this 'intelligence' seems to operate in accordance too different premises than human intelligence (at least for now) (Lawrence, 2017). That is, learning algorithms find patterns in amounts of data, but what these patterns really have to do with the inferred result (i.e., humanly defined objects and concepts) is hard to reveal. And in this lack, a question arises: “[A]re we fooling ourselves when we think that these networks have actually learned the concepts we are trying to teach them?” (Mitchell, 2019, p. 135). Thus, it may be in this gap of knowledge that that the battle for our trust and reliance in these systems are fought. Hence, the crucial question of how these artifacts should be employed and used may not primarily be found in the technical functionality per se but rather in our deficient understanding of how they actually work.

Jonathan Zittrain puts forward the notion of an ‘intellectual debt’ as a way of describing this gap. In analogy with the more rooted term 'technical debt', he sees that we build up an intellectual debt when we do not really know how these systems operate: “This approach to discovery – answers first, explanations later – accrues what I call intellectual debt” (Zittrain, 2019, para. 3). This debt can be exemplified in the so-called “long tail” phenomenon (Mitchell, 2019). A phenomenon which highlights the extensive scope of potential unexpected situations a learning system could face. Hence, if an AI system is trained to act within a well-defined domain (e.g., Chess), an algorithm can virtually learn to categorize and/or predict most potential scenarios with high probability. But if a system instead is to be trained to operate in a more unpredictable and open domain (e.g., autonomous car), data can be acquired which represents most conceivable scenarios, but there are nevertheless situations (edge cases) which are difficult to predict, and therefore to train for – hence, a long tail of unexpected occurrences. Consequently, machine learning algorithms are constrained to the context of which they ‘know’, thus “they will fail when placed in unfamiliar circumstance” (Lawrence, 2017, p. 8). This ‘intellectual debt’ can also be understood in the light of the so-often discussed notion of ‘bias’. Given that these algorithms learn from what they observe during training, two perspective seem to be important in relation to data: (1) Collected data that are insufficient in relation to categories that will be dealt with may render a biased system. But also, data that inherently bear marks of inequalities in society will be reflected during training. (2) A more subtle phenomenon concerns the intuition on how these systems operates: As previously discussed, we may be led to believe that machines learn what we want them to learn in terms of objects or concepts. But if there are irrelevant (for humans) patterns in the data there is a risk that these patterns are correlated with what we want the machine to learn, and that the trained models therefore categorize in a completely different ways than we think they do (Smith, 2020; Mitchell, 2019).

Thus, the first perspective concerns machine learning algorithms and their performances in society in relation to norms, expectations, socially constructed frameworks and to what extent individuals managed by these systems are treated in equal manners or not (cf. Kearns and Roth, 2019). Hence, this perspective foreground how algorithms performatively contributes rendering a social 'reality' where ethical questions of privacy, fairness and interpretability are in focus. The latter perspective concerns machine learning algorithms on a more philosophical level and concerns what these systems actually do perceive, and thus their potential brittleness in relation to what they actually 'understand': I.e., can we be sure that the patterns discovered relates to objects and concepts in a human ontology? (cf. Cantwell Smith, 2019) Hence, this perspective foreground how algorithms performatively contributes rendering a social 'reality' built on knowledge gained through data-driven, pattern-seeking, inductive empirical processes unguided by theory (Clarke, 2019; Zittrain, 2019).

4.3 Complexity

Algorithms does not perform in isolation. Rather, they are part of wider webs consisting of other algorithms, operating in relation to various infrastructures such as databases, storage servers, and information systems (Ågerfalk, 2020). These resources have traditionally been capabilities and systems residing within the digital boarders of a firm (i.e. on-premises data centre). But given the emergence of cloud platforms (e.g. AWS, Microsoft Azure, Google Cloud), an increasing number of organizations outsource their infrastructures, as well as relying on specific services delivered from these platforms (e.g. machine learning capabilities). These new types of entanglements between organizations and platforms are of certain interest when discussing aspects such as accountability, transparency, and explainability (Smith and Browne, 2019). If we return to the earlier discussed paradigms (logic, learning) and phases (shaped, shapes) outlined in the framework above an interesting discussion on a system and it parts emerge. Hence, a developer could make use of a specific, pre-trained, AI-capability (say, face-recognition) delivered from one of the above-mentioned cloud platforms. The developer then design, develops and construct an algorithm for the specific task at hand, and outsources (through API calls) the task of recognising faces to the platform. Hereby, the algorithm comes to rely on a pre-trained machine learning model deployed by said cloud platform. The notion of using API: s delivered by others is not a new phenomenon; the possibility of using public API:s and hereby creating mashups of all sorts has been household since the inception of Web 2.0 (O'Reilly, 2017). Rather, the interesting part with this scenario is that these types of machine learning algorithms comes with characteristics and possibilities of great powers.

Thus, these machine learning APIs are to be perceived as gateways into realms of compute to which an ordinary developer often lacks access. Hence, the interface between developer and platform act as a bridge between two computing domains – one controlled by the developer, and the other controlled by the platform firm. The rising power of cloud platforms then becomes obvious, as they present capacities of using machine learning and artificial intelligence capabilities to a wide audience (Clarke, 2019). Consequently, the cloud platforms become increasingly entangled into solutions and systems where they are the supplier of the 'intelligent' parts of said systems. Hence, cloud platform services become powerful layers of abstractions which obscures the underlying fabric of machine learning (i.e., training, modelling, deploying). Abstractions that allow developers to use these powers of compute without having any previous knowledge of machine learning. Then, when it comes to questions like interpretability and transparency the developer cannot really explain the pre-learned model – which is hidden behind the curtain of the cloud service – but must trust the platform. While the developer makes use of a pre-trained machine learning model, and utilizes its power, they cannot really account for how the specific model was trained – e.g., how much data, or of which type, that were used during training.

Consequently, opacity not only emerges because a machine learning model cannot be explained but also, and perhaps more, due to systems becoming ever more interconnected and thus increasingly complex (cf. Schneier, 2018). Increased complexity in relation to how IT infrastructures evolves is not a new phenomenon (e.g. Sommerville et al., 2012). But while system complexity previously emanated from increased relationships between systems, and where we have been able to rest in an intuition that the individual components are understandable, we now encounter situations where parts of the infrastructure in themselves are incomprehensible. Hence, the overall opacity increases as a function of the complexity of the system itself – its individual components and the relationships between them: "Taken in isolation, oracular answers can generate consistently helpful results. But these systems won't stay in isolation. As AI systems gather and ingest world's data, they'll produce data of their own - much of which will be taken up by still other AI systems" (Zittrain, 2019, para. 11).

5 Conclusion

This paper set out to theorize on algorithms and their performances in the world, with a primary focus on algorithms operating within (what we term) the learning paradigm. Thus, at the centre of attention is the dual nature of algorithms as they are shaped during design but becomes the ones who shapes when put into work. Through a conceptual framework of algorithmic performativity – divided into two phases: (1) 'Shaping algorithms', and (2) 'Algorithms which shapes' – we account for the dynamics by which algorithms are performatively produced and reproduced as they continuously become involved in performances in the 'world'. By applying the framework on aspects of algorithms, we contribute through discussions on how notions such as 'agency', 'reliability' and 'complexity' can be understood and further problematized using a performativity lens. The conceptual framework thus spurs further theorizing by highlighting the temporal, relational and re-shaping aspects of algorithms.

6 References

Andersen, J. V., Lindberg, A., Lindgren, R., and Selander, L. (2016). "Algorithmic agency in information systems: Research opportunities for data analytics of digital traces." In *49th Hawaii International Conference on System Sciences (HICSS)*.

Amoore, L., and Raley, R. (2017). "Securing with algorithms: Knowledge, decision, sovereignty." *Security Dialogue* 48(1), 3–10.

Austin, J. L. (1962) *How to Do Things with Words*. Clarendon Press.

Auramäki, E., Lehtinen, E., and Lyytinen, K. (1988). "A speech-act-based office modelling approach." ACM Transactions on Information Systems 6(2), 126–152.

Barad, K. (2003). "Posthumanist Performativity: Toward an Understanding of How Matter Comes to Matter." *Signs: Journal of woman in culture and society* 28(3), 801–831.

Barr, A. (2018) *The Problem with Software: Why Smart Engineers Write Bad Code*. Cambridge, MA: The MIT Press.

Baskerville, R. L., Myers, M. D., and Yoo, Y. (2020). "Digital First: The Ontological Reversal and New Challenges for Information Systems Research." *MIS Quarterly* 44(2), 509–523.

Bazerman, C. (1998). "The Production of Technology and the Production of Human Meaning." *Journal of Business and Technical Communication* 12(3), 381–387.

Bazerman, C. (1999) *The Languages of Edison's Light*. Cambridge, MA: The MIT Press.

Beer, D. (2017). "The social power of algorithms." *Information, Communication & Society* 20(1), 1–13.

Brockman, J. (2020). *Possible Minds: Twenty-Five Ways of Looking at AI*. New York: Penguin Books.

Brundage, M., Avin, S., Clark, J., Toner, H., Eckersley, P., Garfinkel, B., ... and Amodei, D. (2018). The malicious use of artificial intelligence: Forecasting, prevention, and mitigation. *arXiv:1802.07228*.

Burrell, J. (2016) "How the machine "thinks": Understanding opacity in machine learning algorithms." *Big Data & Society* 3(1), 1–12.

Butler, J. (1993) *Bodies that Matter: On the Discursive Limits of Sex*, New York: Routledge.

Bostrom, N. (2014). *Superintelligence: Paths, Dangers, Strategies*. Oxford: Oxford University Press.

Cantwell Smith, B. (2019). *The promise of artificial intelligence: reckoning and judgement*. Cambridge, MA: The MIT Press

Chesney, R. and Citron, D. K. (2018)." Deep Fakes: A Looming Challenge for Privacy, Democracy, and National Security." *107 California Law Review* (2019, Forthcoming); U of Texas Law, Public Law Research Paper No. 692; U of Maryland Legal Studies Research Paper No. 2018–12.

Clarke, R. (2019). "Why the world wants controls over Artificial Intelligence." *Computer Law & Security Review* 35(4), 423–433.

D'adderio, L. (2010) Artifacts at the centre of routines: performing the material turn in routines theory. *Journal of Institutional Economics*, 7(2), pp. 197–230.

de Laat, P. B. (2018). "Algorithmic decision-making based on machine learning from Big Data: Can transparency restore accountability?" *Philosophy & Technology* 31(4), 525–541.

Dennett, D. (2017). *From Bacteria to Bach and Back – The Evolution of Minds*. New York: W. W. Norton & Company.

Diakopoulos, N. (2014) *Algorithmic accountability reporting: On the investigations of black boxes*.

Domingos, P. (2012). "A Few Useful Things to Know about Machine Learning." *Communications of the ACM* 55(10), 78–87.

Dourish, P. (2016). "Algorithms and their others: Algorithmic culture in context." *Big Data & Society* 3(2), 1–11.

Dourish, P. (2017). *The Stuff of Bits: An Essay on the Materialities of Information*. Cambridge, MA: MIT Press.

Ekbia, H. R. (2009). "Digital Artifacts as Quasi-Objects: Qualification, Mediation, and Materiality." *Journal of the American Society for Information Science and Technology* 60(12), 2554–2566.

Ekbia, H. R., and Nardi, B. A. (2017). *Heteromation, and Other Stories of Computing and Capitalism*. Cambridge, MA: MIT Press.

Faraj, S., and Azad, B. (2012). "The materiality of Technology: An Affordance Perspective." In: *Materiality and organizing: Social interaction in a technological world*. Ed. by Leonardi, P. M., Nardi, B. A., and Kallinikos J. Oxford: Oxford University Press.

Ford, M. (2018). "Introduction." In: *Architects of Intelligence*. Ed. by Martin Ford. Birmingham: Packt Publishing Ltd.

Galloway, A. R. (2006). "Language Wants To Be Overlooked: On Software and Ideology." *Journal of Visual Culture* 5(3), 315–331.

Garud, R., Gehman, J. and Guiliani, A. P. (2018). “Why not take a performative approach to entrepreneurship?” *Journal of Business Venturing Insights* 9, 60–64.
Gillespie, T. (2012). “Can an algorithm be wrong?” *Limn* 1(2), 17–20.
Goffey, A. (2008). “Algorithm.” In: *Software Studies: A Lexicon, Cambridge*. Ed. by M. Fuller. Cambridge, MA: The MIT Press
Goodfellow, I., Bengio, Y. and Courville, A. (2016). *Deep Learning*. Cambridge, MA: The MIT Press
Henfridsson, O., Nandhakumar, J., Scarbrough, H., and Panourgias, N. S. (2018). “Recombination in the Open-Ended Value Landscape of Digital Innovation.” *Information and Organization* 28(2), 89–100.
Introna, L. D. (2013). “Epilogue: Performativity and the becoming of sociomaterial assemblages.” In: *Materiality and Space: Organization, Artefacts and Practices*. Ed. by F-X. de Vaujany, and N. Mitev. London: Palgrave Macmillan.
Introna, L. D. (2016). “Algorithms, governance, and governmentality: On governing academic writing.” *Science, Technology, & Human Values* 41(1), 17–49.
Kallinikos, J. (2002). “Reopening the Black Box of Technology Artifacts and Human Agency.” In: *International Conference on Information Systems*.
Kallinikos, J., Aaltonen, A., and Martonen, A. (2013). “The Ambivalent Ontology of Digital Artefacts.” *MIS Quarterly* 37(2), 357–370.
Kearns, M., and Roth, A. (2019). *The Ethical Algorithm: The Science of Socially Aware Algorithm Design*. New York: Oxford University Press.
Kitchin, R. (2017). “Thinking critically about and researching algorithms.” *Information, Communication & Society* 20(1), 14–29.
Krizhevsky, A., Sutskever, I., and Hinton, G. (2012). “ImageNet Classification with Deep Convolutional Neural Networks.” In: *Advances in neural information processing systems*. pp. 1–9.
Lawrence, N. D. (2017) “Living Together: Mind and Machine Intelligence.” *arXiv preprint arXiv:1705.07996v1*.
LeCun, Y., Bengio, Y. and Hinton G. (2015). “Review: Deep Learning.” *Nature* 521(7553), 436–444.
Lee, E. A. (2017). *Plato and the Nerd: The Creative Partnership of Human and technology*. Cambridge, MA: The MIT Press
Lee, E. A. (2020). *The Coevolution: The Entwined Futures of Humans and Machines*. Cambridge, MA: The MIT Press.
Lee, K-F. (2018) *AI Superpowers: China, Silicon Valley, and the New World Order*. Boston: Houghton Mifflin Harcourt
Licklider, J.C.R. (1960). “Man-Computer Symbiosis.” *IRE Transactions on Human Factors in Electronics* (1), 4–11.
Ljungberg, J. and Holm, P. (1996). “Speech Acts on Trial.” *Scandinavian Journal of Information Systems* 8(1), 29–52.
Lyytinen, K., Nickerson, J. V., and King, J. L. (2020). ”Metahuman systems = humans + machines that learn.” *Journal of Information Technology*, 1–19.
Mackenzie, A (2005) “The performativity of code: Software and cultures of circulation.” *Theory, Culture & Society* 22(1), 71–92.
Mackenzie, A. (2015) “The production of prediction: What does machine learning want?” *European Journal of Cultural Studies* 18(4–5), 429–445.
Mackenzie, A., and Vurdubakis, T. (2011). “Codes and Coding in Crisis: Signification, Performativity and Excess.” *Theory, Culture & Society* 28(6), 3–23.
Matzner, T. (2016). “Beyond data as representation: The performativity of Big Data in surveillance.” *Surveillance & Society* 14(2), 197–210.
Matzner, T. (2019). “The Human is Dead–Long Live the Algorithm! Human-algorithmic ensembles and liberal subjectivity.” *Theory, Culture & Society* 36(2), 123–144.
Milan, S. (2015). “When algorithms shape collective action: Social media and the dynamics of cloud protesting.” *Social Media + Society* 1(2), 1–10.
Mitchell, M. (2019). *Artificial Intelligence: A Guide for Thinking Humans*. Pelican Books
Mittelstadt, B. D., Allo, P., Taddeo, M., Wachter, S., and Floridi, L. (2016). “The ethics of algorithms: Mapping the debate.” *Big Data & Society*, 1–21.

Mohamed, A. R., Dahl, G., and Hinton, G. (2009) "Deep belief network for phone recognition." In: *Nips workshop on deep learning for speech recognition and related applications* 1(9), 1–9.
O'Reilly, T. (2017). *What's the future and why it is up to us*. London: Random House Business Books.
Pennycook, A. (2004) "Performativity and Languages Studies." *Critical Inquiry in Language Studies: An International Journal* 1(1), 1–19.
Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., and Sutskever, I. (2019). "Language models are unsupervised multitask learners." *Technical report, OpenAI Blog*.
Rieder, B (2017). "Scrutinizing an algorithmic technique: the Bayes classifier as interested reading of reality." *Information, Communication & Society* 20(1), 100–117.
Rosenblatt, F. (1958). "The perceptron: a probabilistic model for information storage and organization in the brain." *Psychological review* 65(6), 386–408.
Rose, J., and Truex, D. (2000). "Machine agency as perceived autonomy: an action perspective." In: *Organizational and Social Perspectives on Information Technology*. Ed. by R. Baskerville, J. Stage, and J. I. DeGross. Boston, MA: Springer.
Sandvig, C., Hamilton, K., Karahalios, K., and Langbort, C. (2016). "Automation, algorithms, and politics | When the algorithm itself is a racist: Diagnosing ethical harm in the basic components of software." *International Journal of Communication* 10, 4972–4990.
Schneier, B. (2018). *Click here to kill everybody: Security and survival in a hyper-connected world*. New York: W. W. Norton & Company.
Searle, J. (1969). *Speech Acts: An Essay in the Philosophy of Language*. Cambridge: Cambridge University Press.
Seaver, N. (2017). "Algorithms as culture: Some tactics for the ethnography of algorithmic systems." *Big Data & Society* 4(2), 1–12.
Seaver, N. (2019). "Knowing Algorithms." In: *digitalSTS: A Field Guide for Science & Technology Studies*. Ed. by J. Vertesi and D. Ribes. Princeton, New Jersey: Princeton University Press
Seidel, S., Berente, N., Lindberg, A., Lyytinen, K., Martinez, B., and Nickerson, J. V. (2020). "Artificial Intelligence and Video Game Creation: A Framework For the New Logic of Autonomous Design." *Journal of Digital Social Research* 2(3), 126–157.
Silver, D., Hubert, T., Schrittwieser, J., Antonoglou, I., Lai, M., Guez, A., Lanctot, M., Sifre, L., Kumaran, D., Graepel, T., Lillicrap, T., Simonyan, K., and Hassabis, D. (2018). "A general reinforcement learning algorithm that masters chess, shogi, and Go through self-play." *Science* 362(6419), 1140–1144.
Smith, G. (2020). "Data mining fool's gold." *Journal of Information Technology* 35(3), 182–194.
Smith, B., and Brown, C. A. (2019). *Tools and Weapons: The promise and the peril of the digital age*. New York: Penguin Press.
Sommerville, I., Cliff, D., Calinescu, R., Keen, J., Kelly, T., Kwiatkowska, M., Mcdermid, J., and Paige, R. (2012). "Large-scale complex IT systems." *Communications of the ACM*, 55(7), 71–77.
Tan, M., and Le, Q. V. (2019). "EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks." *arXiv preprint arXiv:1905.11946*.
Totaro, P., and Ninno, D. (2014). "The concept of algorithm as an interpretative key of modern rationality." *Theory, Culture & Society* 31(4), 29–49.
Trielli, D., and Diakopoulos, N. (2019). "Search as News Curator: The Role of Google in Shaping Attention to News Information." In: *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (CHI 2019).
van Dijck, J. (2014) Datafication, Dataism and Dataveillance: Big Data between scientific paradigm and ideology. *Surveillance & Society* 12(2), pp. 197–208.
Winograd, T., and Flores, R. (1986). Understanding Computers and Cognition: A new foundation for design. Ablex Publishing Corporation.
Wooley, S. C., and Howard, P. N. (2016). "Automation, algorithms, and politics | Political Communication, Computational Propaganda, and Autonomous Agents – Introduction." *International Journal of Communication*, 10, 4882–4889.
Yoo, Y., Henfridsson, O., and Lyytinen, K. (2010). "Research commentary – The new organizing logic of digital innovation: An agenda for information systems research." *Information Systems Research* 21(4), 724–735.

Yoo, Y., Boland Jr, R. J., Lyytinen, K., and Majchrzak, A. (2012). "Organizing for innovation in the digitized world." *Organization Science* 23(5), 1398–1408.

Ziewitz, M. (2016). "Governing algorithms: Myth, mess, and methods." *Science, Technology, & Human Values* 41(1), 3–16.

Zittrain, J. (2008). *The Future of the Internet – and how to stop it*. New Haven: Yale University Press

Zittrain, J. (2019). "The Hidden Costs of Automated Thinking." *The New Yorker*, 23 July.

Ågerfalk, P. (2020). "Artificial intelligence as digital agency." *European Journal of Information Systems* 29(1), 1–8.