RECURSIONS

A Mechanology

Engines

of Algorithmic

of Order

Techniques

BERNHARD RIEDER

Engines of Order

The book series RECURSIONS: THEORIES OF MEDIA, MATERIALITY, AND CULTURAL TECHNIQUES provides a platform for cuttingedge research in the field of media culture studies with a particular focus on the cultural impact of media technology and the materialities of communication. The series aims to be an internationally significant and exciting opening into emerging ideas in media theory ranging from media materialism and hardware-oriented studies to ecology, the post-human, the study of cultural techniques, and recent contributions to media archaeology. The series revolves around key themes:
– The material underpinning of media theory
– New advances in media archaeology and media philosophy
– Studies in cultural techniques

These themes resonate with some of the most interesting debates in international media studies, where non-representational thought, the technicity of knowledge formations and new materialities expressed through biological and technological developments are changing the vocabularies of cultural theory. The series is also interested in the mediatic conditions of such theoretical ideas and developing them as media theory.

Editorial Board
– Jussi Parikka (University of Southampton)
– Anna Tuschling (Ruhr-Universität Bochum)
– Geoffrey Winthrop-Young (University of British Columbia)

Engines of Order

A Mechanology of Algorithmic Techniques

Bernhard Rieder

Cover illustration: The full text of this book, represented as a *feature vector*. © Bernhard Rieder

Cover design: Suzan Beijer
Lay-out: Crius Group, Hulshout

AUP
Amsterdam
University
Press

Table of Contents

Acknowledgements

Introduction

Abstract
The introduction chapter positions algorithmic information ordering as a central practice and technology in contemporary digital infrastructures, a set of techniques that serve as 'levers on reality' (Goody). While algorithms used in concrete systems may often be hard to scrutinize, they draw on widely available software modules and well-documented principles that make them amendable to humanistic analysis. The chapter introduces Gilbert Simondon's mechanology and provides an overview of the structure and argument of the book.

Keywords: algorithmic information ordering, information search and retrieval, mechanology, software-making

Over the last decades, and in particular since the widespread adoption of the Internet, encounters with algorithmic procedures for 'information retrieval' – the activity of getting some piece of information out of a collection or repository of some kind – have become everyday experiences for most people in large parts of the world. We search for all kinds of things on the open web, but also for products, prices, and customer reviews in the specialized databases of online retailers, for friends, family, and strangers in social networking services or dating sites, and for the next thing to read, watch, play, listen to, or experience in quickly growing repositories for media contents. There are at least three remarkable aspects to this spread of information seeking. First, computer-supported searching has sprawled beyond the libraries, archives, and specialized documentation systems it was largely confined to before the arrival of the web. Searching, that is, the act of putting a query into a form field, has become such a fundamental and ubiquitous gesture that a missing search box on a website becomes an almost disturbing experience. Second, what retrieval operates on – information – has come to stand for almost anything, from scraps of knowledge to things, people, ideas, or experiences. Digitization, datafication, and the capture of

always more activities in software are, in the words of Netscape founder and venture capitalist Marc Andreessen (2011), 'eating the world'. Search has become a dominant means to access and order the masses of digital and datafied bits and pieces that clutter the environments we inhabit. Third, the deliberate and motivated act of formulating a query to find something is only one of the many forms in which information retrieval nowadays manifests itself. Automated personalization, localization, recommendation, filtering, classification, evaluation, aggregation, synthetization, or ad hoc generation of information are similarly pervasive practices that do not require explicit user input to select, sequence, arrange, or modulate some set of digital items. And retrieval techniques are no longer limited to producing result lists: they generate scores, suggest items, discard or promote messages, set prices, arrange objects and people in relation to each other, assemble texts, forbid or grant access, fabricate interfaces and visualizations, and even steer objects in the physical world. In short, various activities or gestures this book addresses under the broad notion of 'information ordering' have become both pervasive and subtle in terms of how they operate in the thickening layers of digital mediation.

The proliferation of these algorithmic practices has been accompanied by considerable efforts in the humanities and social sciences to investigate techniques and applications in terms of power and social significance. Early analyses of search engines already highlighted their political dimension, claiming that 'there is no such thing as algorithms without their own weight' (Winkler, 1999, p. 36). This meant that one could examine 'the wide-ranging factors that dictate systematic prominence for some sites, dictating systematic invisibility for others' (Introna and Nissenbaum, 2000, p. 171) from a point of view concerned with social impact and public interest. Beyond search, authors have called attention to 'moments of algorithmic judgement' (Graham, 2005, p. 576) that abound when 'code-based technologized environments continuously and invisibly classify, standardize, and demarcate rights, privileges, inclusions, exclusions, and mobilities' (Graham, 2005, p. 563). Terms like 'automated management' (Kitchin and Dodge, 2011), 'algorithmic ideology' (Mager, 2012), 'algorithmic governmentality' (Berns and Rouvroy, 2013), and, more recently, 'algorithmic accountability' (Diakopoulos, 2015) all subscribe to 'the central premise that algorithms have the capacity to shape social and cultural formations and impact directly on individual lives' (Beer, 2009, p. 994). This broad recognition of the 'relevance of algorithms' is not, however, a symptom of a sudden curiosity for the fundamentals of computational theory. It stems from a more specific interest in the particular instances where algorithms serve as 'a means to know what there is to know

and how to know it, to participate in social and political discourse, and to familiarize ourselves with the publics in which we participate' (Gillespie, 2014, p. 167). Most of the techniques that sit at the center of these questions and concerns directly relate to the field of information ordering.

Search engines remain the most instructive illustration for the issues at hand since the tensions between their remarkable practical utility, their technical prowess, and their political relevance are so clearly visible. We intuitively understand that ranking web pages – and thus the services, contents, and viewpoints they stand for – is delicate business. But, as Grimmelmann (2009) argues, search engines face the 'dilemma' that they *must* rank in order to be useful. This imperative collides with the uncomfortable observation that there is arguably no technical procedure that can lay serious claim to producing assessments concerning ambiguous and contested cultural matters in ways that could be broadly accepted as 'objective'. In fact, whenever data are processed algorithmically, the transformation from input to output implies a perspective or evaluation that, through the coordination between data and what they stand for, is projected back into spheres of human life. Techniques for information retrieval become engines of order that actively intervene in the spaces they seek to represent (cf. Hacking, 1983).

The need to better understand the specificities of these processes becomes even clearer if we broaden the scope beyond everyday online experiences to activities where algorithms evaluate and inform decisions that can have dramatic effects, for example, in hiring, credit assessment, or criminal justice (cf. O'Neil, 2016; Christin, 2017; Eubanks, 2018). These emblematic and troubling applications point to a myriad of instances in business and government where procedures from the broad field of information ordering are used to inspire, choose, or impose a specific course of action.

The technical procedures involved are loaded, often implicitly, with specific ideas and attitudes concerning the domains they intervene in. Search engines evaluate the 'relevance' of information, news aggregators generate front pages according to various measures of 'newsworthiness', dating sites calculate 'compatibility coefficients' between members and order them accordingly, social networking sites filter friends' status updates based on quantified ideas of 'interest' or 'closeness', and microblogging services give prominence to 'trending' topics. In each of these cases, there is a framing of the application domain that implies various kinds of conceptual and normative commitments. This can involve a general allegiance to the broad epistemological 'style' (Hacking, 1985) of computation as a means of knowing; but it can also take more specific forms, for example, when

psychological research on partnership satisfaction flows into the design of a matching algorithm or when the optimization objectives for a machine learning system are being selected on the basis of business considerations.

At the same time, technical procedures are more than just a means to efficiently enact values and ideas that are themselves nontechnical. Jack Goody (1977) argued that list-making, from the start an essential part of writing, 'gives the mind a special kind of lever on "reality"' (p. 109) by supporting mnemonics and, more importantly, by facilitating different operations of ordering and reordering pieces of text and, by extension, the things these pieces refer to. As Goody knew all too well, the advent of list-making meant not just a quantitative extension in cognitive capacity. More fundamentally, it stimulated the production and recording of knowledge, spurred modes of classificatory and hierarchical thinking, and supported more complex forms of social organization. As Peters (2015) argues, '[i]n list writing, serial order loosens its hold' (p. 290), with wide-ranging consequences. The information ordering techniques that have become so pervasive today share the transversal character and broad applicability of list-making and may prove to have equally fundamental repercussions for how we construct and relate to the world around us.

Like list-making, algorithmic ordering comes with a genuine operational substance that rarely boils down to a simple transposition of a manual method into computational form. A web search engine, for example, orders documents through iterative processing of vast amounts of distributed signals and the specific way it produces an aggregate appreciation of these signals defines an epistemic substance and character that has little to do with the knowledge practices that have defined libraries, encyclopedias, or archives over the last millennium. As Edsger Dijkstra, one of the central figures in the history of software, remarked about computers over 40 years ago:

> [T]he amount of information they can store and the amount of processing that they can perform, in a reasonably short time, are both large beyond imagination. And as a result, what the computer can do for us has outgrown its basic triviality by several orders of magnitude. (Dijkstra, 1974, p. 608)

Computers' capacity to run billions of data points through billions of iterations of small calculative steps means that they 'think' (Burrell, 2016) in ways that are not only opaque, but potentially *strange* and hard to fit into established categories. Techniques like machine learning, network algorithms, or relational database management systems are not just powerful

means to produce and apply knowledge, to enact value preferences, or to control practice; they participate in the very definition of what knowledge, value, and practice mean and *can* mean, both through the conceptual resources they propose to think with and the actual interpretations and orderings they generate when applied in practice. We should consider the possibility that they challenge cultural modes and social institutions in more fundamental ways than the necessary discussions of algorithmic opacity or bias can lead us to believe.

The methods and procedures involved in actual practices are often hidden from our sight by technical and legal means, latched not even in black boxes but somewhere in the 'black foam' (Rieder, 2005) of systems whose contours are hard to delineate. But, paradoxically, they have also become highly accessible, in the sense that concrete implementations draw heavily on open reservoirs of technicity and knowledge that find their expression in scholarly publications, software libraries, and communities of practice gathering on websites like Stack Overflow. These reservoirs are neither hidden nor closed off and we are free to examine a steadily growing archive of techniques that enable computers to accomplish tasks that seem increasingly 'cultural' or 'intelligent' in nature. This book is an expedition into this archive and more specifically into the areas that deal with information ordering.

The actual makeup of Google's search ranking may indeed be 'unknowable' for a number of practical, commercial, and legal reasons, but, as shown in Chapter 7, the content, history, and substance of its most famous algorithm, PageRank, stands wide open. We may never get access to the concrete specifications of the machine learning methods behind the personalized filtering Facebook applies to its users' News Feed, but we can ask, as in Chapter 6, where machine learning comes from, what concepts and ideas it builds on, and how it operates in general terms. The second part of this book is thus dedicated to a series of investigations into specific 'algorithmic techniques', that is, into the defined-yet-malleable units of technicity and knowledge developers draw on when designing the function and behavior of computers acting in and on the world. Offering many different ways to order and organize information, they serve as levers on the 'reality' of a world eaten by software.

While this book draws heavily on work situated in the 'cultural techniques' tradition, an approach coming out of German media scholarship, there is at least one important difference. Unlike Young's (2017) inspirational take on the list, which follows a particular cultural form through various societal settings, I examine a set of techniques as they traverse what is maybe not a single cultural domain but nonetheless a somewhat demarcated practice:

software-making. The broader theoretical perspective guiding these probes will be discussed at length in part one, but the particular focus on technical creation calls for some background and clarification.

Toward Mechanology

This book is largely motivated by the remarkable spread of algorithmic information ordering but also translates a feeling of hesitation or uneasiness toward the way software is often presented and discussed in media studies and associated fields, or, more specifically, toward the emphasis on *code* as software's quintessential technical quality or substance. To be clear, understanding how written instructions produce machine behavior is fundamental to understanding software, but it is also a comparatively small step into the massive world of technicity software constitutes. Code is neither trivial nor transparent, but for any experienced developer it is a familiar means to access a domain of *function* that is vastly more complex than the term is able to address. Building a program or system is to craft a composite technical object, 'a being that functions' in the words of French philosopher Gilbert Simondon, who plays a central role in what follows. This may entail, today more than ever, the assemblage of many preexisting chunks of software. Code serves as the means to draw on an archive, to 'build-with', and to create in ways that are deeply relational and embedded. As I will argue over the following chapters, the world of software-making is structured around 'techniques', expressions of knowledge and technicity that enable developers to make computers *do* things that are more involved or complex than their 'basic triviality' suggests. This book does not presume any practical technical knowledge or experience, but it addresses algorithmic information ordering from the perspective of technical creation.

My own background plays an important role in this setup. While I have little formal training in any technical discipline, I have been developing software on a regular basis for a long time. I started to program when I was a still in high school, worked as a web developer during my university studies, and taught programming to students ranging from beginners to computer scientists at master's level for about a decade. I continue not only to code but to make software, nowadays mostly in the domain of digital methods for Internet research (Rogers, 2013). The part of the software landscape under scrutiny in this book, algorithmic information ordering, is not only socially relevant but also closely connected to the technical practice I have been pursuing over the last 20 years. As a web developer, I worked

extensively with relational database management systems (Chapter 4) and I encountered advanced information retrieval techniques (Chapter 5) during my PhD in information and communication science at Paris 8 University when I was investigating the possibilities for 'society-oriented design' (Rieder, 2006). This work led to a system, procspace (Rieder, 2008), which used a variety of algorithmic methods to generate navigational pathways between documents to support a logic of connection, enrichment, and overview that breaks with the serial forms of order dominating search. The encounter with information retrieval, an established technical field that comes with a large body of well-documented methods, came as a shock: as an autodidact programmer I felt very comfortable when it came to writing code, but I was not fully aware how much I was missing. The techniques I discovered gave me a new sense of possibility and opened the door to forms of technical expression that have stimulated my imagination ever since. Although often more heavily mathematized than what I was used to, these techniques were relatively simple to implement and, like clay, could be modeled in countless ways. The entanglement between information ordering and the politically, culturally, and economically significant matters it is increasingly involved in became my principal research interest. This eventually led to work in digital methods, where I focused on studying online platforms that rely on algorithmic techniques in fundamental ways and, paradoxically, to a situation where I would apply similar techniques as analytical instruments to make sense of large sets of empirical data. The chapters about machine learning (Chapter 6) and network algorithms (Chapter 7) draw on this work.

The reason I mention these details is not to claim technical authority but to introduce and situate a perspective that has been fundamentally shaped by these experiences. This perspective is still uncommon in media studies and in the broader discussions of software or, to use the buzzwords of the day, of 'algorithms' or 'artificial intelligence'. Following Johanna Drucker's (2013) suggestion to give '[m]ore attention to acts of producing and less emphasis on product' (n.p.), my conceptual vantage point is software-making, a series of practices that increasingly revolve around the use of packaged function as a means to extend programmers' capabilities. It takes hardly more than an hour to install and set up PyTorch or TensorFlow, powerful open-source libraries for machine learning, and to have a first classifier trained. While some people will want to peek under the hood of these artifacts to make adaptations or simply out of intellectual curiosity, developers often draw on technicity and knowledge that they understand only in broad terms or not at all. What programming languages, software libraries, and similar

artifacts do is to enable software-makers to step further faster, not merely regarding resource efficiency but in terms of what can be considered possible in the first place. Such packages widen the spaces of expressivity, broaden the scope of ambitions, but also structure, align, and standardize. Spelled out, stabilized, and 'frozen', algorithmic techniques spread through technical imaginaries and artifacts, and further into application logics and business models. They are *means of production*, not simply outpourings of computational principles or scientific ideas.

Algorithmic techniques are ways of making computers do things, of creating function, and their history is characterized to a greater extent by accumulation and sedimentation than by paradigm shifts or radical breaks. Certainly, methods and approaches are regularly superseded or fall out of fashion, but it is clear that the archives that inform and constitute software-making have grown vastly over time. While this book entertains a somewhat complicated relationship with the field of media archeology, another prominent approach coming out of German media theory, it indeed follows a selection of techniques into their historical trajectories to excavate some of the fundamental ideas that resonate through our technical present. But throughout these historical probes, I strive to keep an eye on the possibilities for variation, combination, and divergence that invariably emerge when a technique becomes part of a concrete technical object. The developer, in contrast to the computer scientist, philosopher of science, or science historian, neither looks at the reservoir of techniques from below, as an emanation of foundational mathematical principles, nor from above, as outpourings of scientific progress. The developer is right in-between, surrounded by technicity coming in all shapes and forms, and thus '*among* the machines that operate with him' (Simondon, 2017, p. 18).

To interrogate technology both in terms of its fundamental nature and from the perspective of technical practice is the task Simondon laid out for 'mechanology', a discipline or mode of thinking that would serve as a 'psychology' or 'sociology' of machines (Simondon, 2017, p. 160), capturing their 'interior life' and 'sociability' in terms that do not reduce them to an exterior finality or effect. As a general science of technology, mechanology would approach technical function as human gesture, examine technical creation as mediation between human beings and nature, and interrogate the *values* implied in mechanical operation itself. This book, suffice to say, is an attempt to develop a mechanological perspective on software and to apply it to the engines of order that increasingly adjudicate (digital) life.

Organization and Overview

The book is divided into two parts. The first part is dedicated to the theoretical and methodological foundations that inform and support the examination of four clusters of algorithmic techniques for information ordering in the second part.

The first chapter discusses central terms like 'information' and 'order', and it proposes the concept of 'engine' to point toward the infrastructural embeddings that have allowed techniques initially conceived for document retrieval to become pervasive mediators in online environments. While this book constitutes a humanistic exploration of technical substances rather than their practical application, the chapter pays tribute to the fact that the techniques under scrutiny have become prevalent in a specific situation, in *this* world and not another.

The second chapter then formulates a conceptual perspective on software, starting from an attempt to situate the project in relation to existing takes on the subject. But it is mainly dedicated to the presentation and appropriation of Simondon's philosophy of technology, which reserves a central place to technical creation and evolution. Here, we find an understanding of technicity as a domain of life that constitutes its own substance and regularity, whilst remaining a fundamental form of human gesture. Simondon's inductive view, which frames technology as multitude of technical objects rather than idealized techne, grounds the conceptual and analytical apparatus I then bring to the analysis of algorithmic techniques.

Chapter 3 builds on central ideas from Simondon's work, such as the distinction between invention and concretization and the delineation of technical elements, individuals, and ensembles, to conceptualize algorithmic techniques as the central carriers of technicity and technical knowledge in the domain of software. In dialogue with the cultural techniques tradition, it addresses them as methods or heuristics for creating operation and behavior in computing and discusses how they are invented and stabilized. Algorithmic techniques, in this perspective, are at the same time material blocks of technicity, units of knowledge, vocabularies for expression in the medium of function, and constitutive elements of developers' technical imaginaries.

The second part of the book then launches a series of probes into the history of algorithmic information ordering. These probes do not follow a single lineage or logic and cover different periods of time, but they come together in staking out an 'excavation ground' (Parikka, 2012, p. 7) that marks the 1960s and 1970s as the period where the fundamentals of contemporary

information ordering were laid out. While Simondon's understanding of technology as human gesture and my emphasis on adaptation and variation lead away from certain core tenets of media archeology, I seek 'to investigate not only histories of technological processes but also the current "archaeology" of what happens inside the machine' (Parikka, 2012, p. 86). The goal is to excavate select roots of an increasingly technological present. The four clusters of algorithmic techniques examined share the characteristic that they are highly relevant to contemporary information ordering while remaining fundamentally understudied, both in their historical and conceptual dimension. Looking at the inception and evolution of algorithmic techniques allows us to examine them in a state of relative 'liquidity', where they have not yet been fully stabilized or 'frozen' into the canon, remaining precarious propositions that have to be explained and justified in terms that are absent from contemporary publications in the computing disciplines.

Chapter 4 serves as a topic-focused introduction that situates contemporary information ordering in a historical lineage that is largely absent from dominant narrations. Although the story starts off from standard takes on knowledge organization and classification in libraries and encyclopedias, it zeros in on the field of information retrieval, which develops in fundamental opposition to even the most visionary of library techniques, not merely in terms of technology and method, but regarding the idea of order itself. Coordinate indexing, the first and defining technique in this lineage, is explicitly designed to eliminate the influence of librarians and other 'knowledge mediators' by shifting expressive power from the classification system to the query and, by extension, to the information seeker. Order is no longer understood as a stable map to the universe of knowledge but increasingly as the outcome of a dynamic and purpose-driven process of ordering. Although equally foundational for the statistical tradition in information retrieval, the chapter closes by discussing coordinate indexing as a precursor of the relational model for database management, which underpins large swaths of contemporary information handling, from enterprise software to web platforms.

Chapter 5 investigates the early attempts in information retrieval to tackle the full text of document collections. Underpinning a large number of contemporary applications, from search to sentiment analysis, the concepts and techniques pioneered by Hans Peter Luhn, Gerard Salton, Karen Spärck Jones, and others involve not only particular framings of language, meaning, and knowledge, they also introduce some of the fundamental mathematical formalisms and methods running through information ordering, preparing the extension to digital objects other than text documents. The chapter specifically seeks to capture the considerable technical expressivity that

comes out of the sprawling landscape of research and experimentation that characterizes the early decades of information retrieval. It also documents the emergence of a conceptual construct and 'intermediate' data structure that is fundamental to most algorithmic information ordering at work today: the feature vector.

Chapter 6 examines one of many areas where feature vectors play a central role. Machine learning is currently one of the most active domains in computer science and the wide availability of datasets and increasingly robust techniques have led to a proliferation of practical applications. The chapter uses the Bayes classifier as an entry point into the field, showing how a simple statistical technique introduced in the early 1960s is surprisingly instructive for understanding how machine learning operates more broadly. The goal is to shed light on the core principles at work and to explain how they are tweaked, adapted, and developed further into different directions. This chapter also develops the idea that contemporary information ordering represents an epistemological practice that can be described and analyzed as 'interested reading of reality', a particular kind of inductive empiricism.

Chapter 7 ventures into the field of network algorithms to discuss yet another way to think about information ordering. While Google's PageRank algorithm has received considerable attention from critical commentators, the vast intellectual landscape it draws on and contributes to is less well known. Graph algorithms are used in many different settings, not least in the social sciences, yet the technical and epistemological commitments made by graph theoretical formulations of 'real life' phenomena are hardly a subject of discussion beyond specialist circles. The chapter shows how algorithmic ordering techniques exploit and integrate knowledge from areas other than information retrieval and demonstrates how the 'politics' of an algorithm can depend on small variations that lead to radically different outcomes. The context of web search means that the various techniques covered in the second part of the book can be brought together into a shared application space, allowing for a more concrete return to earlier discussions of variation and combination in software.

The conclusion, finally, synthesizes algorithmic information ordering into a denser typology of ordering gestures, paying particular attention to the modes of disassembly and reassembly that inform the underlying techniques. The attempt to distill an operational epistemology from the cacophony of techniques begs the question whether we are witnessing the emergence of a new *épistémè* (Foucault, 2005), a far-reaching set of regularities that characterize how we understand and operationalize the very notion of order at a given time and place. Independently from how we answer this

question, it is clearly impossible to avoid the more immediately pressing need to understand how the capacity to arrange individuals, populations, and everything in-between in highly dynamic and goal-oriented ways relates to contemporary forms of capitalism. To face this challenge, I come back to Simondon's mechanology and its broader cousin, technical culture, as a means to promote a 'widening' of technical imagination and appropriation. While certainly not enough to solve the many concrete issues surrounding advanced algorithmic techniques, an understanding of technicity as human gesture – albeit of a specific kind – can sharpen our view for the many instances where technology has become complicit in domination, for the reconfigurations of power relations that occur when new levers begin to operate in and on society, and for the increasing interdependence between technical critique and social critique.

Bibliography

Andreessen, M. (2011). Why Software Is Eating the World. *Wall Street Journal*, 20 August. Retrieved from http://online.wsj.com.

Beer, D. (2009). Power through the Algorithm? Participatory Web Cultures and the Technological Unconscious. *New Media & Society 11*(6), 985-1002.

Berns, T., and Rouvroy, A. (2013). Gouvernementalité algorithmique et perspectives d'émancipation. Le Disparate comme condition d'émancipation par la relation? *Réseaux 177*, 163-196.

Burrell, J. (2016). How the Machine 'Thinks': Understanding Opacity in Machine Learning Algorithms. *Big Data & Society 3*(1), 1-12.

Christin, A. (2017). Algorithms in Practice: Comparing Web Journalism and Criminal Justice. *Big Data & Society 4*(2), 1-14.

Diakopoulos, N. (2015). Algorithmic Accountability. *Digital Journalism 3*(3), 398-415.

Dijkstra, E. W. (1974). Programming as a Discipline of Mathematical Nature. *American Mathematical Monthly 81*(6), 608-612.

Drucker, J. (2013). Performative Materiality and Theoretical Approaches to Interface. *Digital Humanities Quarterly 7*(1), n.p.

Eubanks, V. (2018). *Automating Inequality: How High-Tech Tools Profile, Police, and Punish the Poor*. New York: St. Martin's Press.

Foucault, M. (2005). *The Order of Things: An Archaeology of the Human Sciences*. London: Routledge.

Gillespie, T. (2014). The Relevance of Algorithms. In T. Gillespie, P. J. Boczkowski, and K. A. Foot (eds.), *Media Technologies: Essays on Communication, Materiality, and Society* (pp. 167-195). Cambridge, MA: MIT Press.

Goody, J. (1977). *The Domestication of the Savage Mind*. Cambridge: Cambridge University Press.

Graham, S. D. N. (2005). Software-Sorted Geographies. *Progress in Human Geography 29*(5), 562-580.

Grimmelmann, J. (2009). The Google Dilemma. *New York Law School Law Review 53*, 939-950.

Hacking, I. (1983). *Representing and Intervening*. Cambridge: Cambridge University Press.

Hacking, I. (1985). Styles of Scientific Reasoning. In J. Rajchman and C. West (eds.), *Post-Analytic Philosophy* (pp. 145-163). New York: Columbia University Press.

Introna, L. D., and Nissenbaum, H. (2000). Shaping the Web: Why the Politics of Search Engines Matters. *The Information Society 16*(3), 169-185.

Kitchin, R., and Dodge, M. (2011). *Code/Space: Software and Everyday Life*. Cambridge, MA: MIT Press.

Mager, A. (2012). Algorithmic Ideology. *Information, Communication & Society 15*(5), 769-787.

O'Neil, C. (2016). *Weapons of Math Destruction: How Big Data Increases Inequality and Threatens Democracy*. New York: Crown.

Parikka, J. (2012). *What Is Media Archaeology?* Cambridge: Polity Press.

Peters, J. D. (2015). *The Marvelous Clouds: Toward a Philosophy of Elemental Media*. Chicago: University of Chicago Press.

Rieder, B. (2005). Networked Control: Search Engines and the Symmetry of Confidence. *International Review of Information Ethics 3*, 26-32.

Rieder, B. (2006). *Métatechnologies et délégation: pour un design orienté-société dans l'ère du Web 2.0*. PhD dissertation, Université Paris 8. Retrieved from https://tel.archives-ouvertes.fr/tel-00179980/.

Rieder, B. (2008). An Experimental Collaboration Tool for the Humanities: The Procspace System. In P. Hassanaly, T. Herrmann, G. Kunau, and M. Zacklad (eds.), *Supplement to the Proceedings of the 7th International Conference on the Design of Cooperative Systems, Carry-le-Rouet, France, May 9-12* (pp. 123-130). IOS Press: Amsterdam.

Rogers, R. (2013). *Digital Methods*. Cambridge, MA: MIT Press.

Simondon, G. (2017). *On the Mode of Existence of Technical Objects* (C. Malaspina and J. Rogove, trans.). Minneapolis: Univocal Publishing.

Winkler, H. (1999). Search Engines: Metamedia on the Internet? In J. Bosma (ed.), *Readme! Filtered by Nettime: ASCII Culture and the Revenge of Knowledge* (pp. 29-37). New York: Autonomedia.

Young, L. C. (2017). *List Cultures: Knowledge and Poetics from Mesopotamia to BuzzFeed*. Amsterdam: Amsterdam University Press.