London School of Economics and Political Science

NEGOTIATING WITH SOCIAL ALGORITHMS IN THE DESIGN OF SERVICE PERSONALIZATION

Cody Dodd

A thesis submitted to the Department of Management

London School of Economics and Political Science

for the degree of Doctor of Philosophy

London, United Kingdom. October 2021

Declaration

I certify that the following thesis presented for examination for the PhD degree of the London School of Economics and Political Science is solely my work, other than where clearly indicated.

 I declare that my thesis consists of 65,101 words (excluding references and appendices).

Statement of conjoined work

I confirm that Chapter 5 was co-authored with Dr. Antonio Cordella. Both authors were first authors, equal in contribution.

I also confirm that Chapter 6 was co-authored with Dr. Maha Shaikh and Dr. Antonio Cordella. Dr. Maha was the first author, having played a leading role in shaping the analysis. However, as second author I played a formative and primary role in the research methodology and data interpretation. Data was made accessible not only through ethnographic investigations that were conducted by myself, but the interpretation of the algorithms were only possible due to my technical expertise.

Acknowledgement

Dr. Antonio Cordella

The most important experience for me as a researcher in this program has been the tutelage and mentorship by Dr. Antonio Cordella. From the practical lessons like writing techniques to the rigorous theoretical training, you have helped me expand my research boundaries.

Dana, Rohit, Hannah

I made many life-long friends in this program. Through twists and turns, three research peers were important allies in helping me understand this dynamic and exciting field.

Family + friends

I could not have completed this journey without the support from my family and my close friends.

Conference papers

This thesis began with a series of conference papers exploring the relationship between personalization and AI.

1. Dodd, C. Negotiating Personalization in Dynamic User Interfaces for the Public Sector: Domain Specificity and Algorithmic Explainability. 53rd Hawaii International Conference on System Sciences. January 2019. *Withdrew from publication due to schedule conflict with a mandatory department meeting.*
2. Dodd, C. It Takes Two to Tango: Bringing Together Users and Artificial Intelligence to Create Public Value. 20th Annual International Conference on Digital Government Research. Dubai, UAE. June 2019.

Papers included

This thesis includes three papers which provide empirical support for the broader arguments made herein. It comprises versions of the following original works that are in review.

1. Cordella, A., & Dodd, C. AI and Public Value Creation: How Algorithms Shape Organization Action. [Under review at a leading Information Systems journal]
2. Shaikh, M., Cordella, A. & Dodd, C. Learning Algorithms in Organizations: A Process View of Algorithmic Explainability to Algorithmic Opacity. [Under review at a leading Information Systems journal]
3. Dodd. C. Negotiating complexity across the development of algorithmic and non-algorithmic personalization. [Under review at a leading Information Systems journal]

The conference submissions and papers were all researched and written within enrolment in the PhD in Information Systems and Innovation at the Department of Management, London School of Economics and Political Science, 2016-2021.

NEGOTIATING WITH SOCIAL ALGORITHMS IN THE DESIGN OF SERVICE PERSONALIZATION

Abstract

Supported by three standalone yet complimentary essays, this thesis investigates the development of service personalization that has been mediated by technologies characterized as having elements of Artificial Intelligence (AI), including prediction, natural language processing, and machine learning. The aim of this work is to expand our understanding of the role emerging technologies play in affording personalization, and personalization's relationship with systems increasingly capable of mediating experiences directly with users.

Data was collected from participant observation of an AI development company over two and a half years and comprised of a detailed mapping of the technologies as well as development documents, chats, meetings, and interviews with developers and key users.

We found that the implementation of deeper forms of personalization over time led to the adoption of emerging technologies like AI. In the context of a government agency, these algorithms changed the way employees are screened and selected. We also found that requests for personalization led to increasingly opaque systems where interpretation about how algorithms work emerges in place of an explanation of how they work. Building upon these findings, a framework was developed to investigate 34 discrete cases of personalization across dimensions of ease of design and ease of understanding. We found that the pursuit of deeper personalization leads to the adoption of tools that make increasingly social decisions. That is, we utilize social technologies despite their complexity because they make faster and deeper decisions about individuals from social data than can be done without them. To accomplish this, various strategies are employed to help increase user tolerance for a lack of understanding of their inner workings and to ensure they operate within bounds acceptable to users and the designers of the systems. As these systems gain increasing autonomy, issues of bias amplification, privacy, and an increasingly inexplainable logic behind decision-making remain persistent and this has implications for theory and practice.

Contents

Figures

Tables

Abbreviations

AI	Artificial intelligence
API	Application programming interface
FAST	Fairness, accountability, sustainability and transparency
COMPAS	Correctional Offender Management Profiling for Alternative Sanction
CSS	Cascading style sheets
GLP	Global Leadership Programme
HR	Human resources
HTML	Hypertext markup language
ICTs	Information communication technologies
IT	Information technology
JSON	JavaScript object notation
LDA	Latent Dirichlet allocation
ML	Machine learning
NLP	Natural language processing
NLTK	Natural Language Toolkit
PMO	Prime Ministers Office
RFP	Request for proposal
SQL	Structured query language
The Company	The AI company that was the source of data for Chapters 5, 6, 7
TD-IDF	Term frequency–inverse document frequency
UAE	United Arab Emirates
UI	User interface
UK	United Kingdom
UX	User experience

Chapter 1: Introduction

At a macro glance this thesis explores humans and machines from the perspective of personalization. More precisely, this research elaborates how efforts at creating personalized service experiences influence the adoption of new technologies like artificial intelligence and machine learning. The adoption of these tools to facilitate more advanced personalized experiences in turn introduces changes to our organizational processes by handing over some decision-making autonomy to machines. The process of building these new systems is introducing new risks, both from the technologies and from the designers of these technologies. As machines take on increasingly consequential roles in our social networks, such as in the delivery of service experiences, they are increasingly important actors that are worthy of careful understanding.

1.1 Machines that learn

Throughout history humans have long been fascinated with the idea of human-like machines. In the Talmud, an early Jewish text, *golems* are anthropomorphic husks that resemble humans, made of mud. They are close to human but not fully. In the early stories, they could not speak, and were *unlearned*. The name golem itself references being a raw form rather than full form, or fully human. The ancient Greeks tell stories about having living bronze statues and mechanical handmaidens that carried out daily tasks (Gera, 2003). Egyptian statues were believed to be animated to support rituals (Maspero, 2009). 7th century Buddhist scholar Daoxuan describe human-shaped automata that recite sacred texts (Littlejohn and Dippmann, 2011). Across many of these stories, learning or agency separates humans from machines. They fulfilled predefined roles that supported human tasks. Increasingly however, machines are being given tasks that they can execute themselves and are being asked to make judgements built from social experiences and social learning. That is, golems are learning. The rise of learning machines challenge common childhood taxonomies that delineate stones (non-living) from plants (living), animals (conscious) and humans (rational). An attention to human-like characteristics such as decision-making autonomy, combined with *seemingly rational behaviour* can create the impression of a rational machine actor (Goffman, 1958; Marakas et al., 2000). This has contributed to a metaphorical personification that has helped computers be seen as social actors with social roles, as opposed to purely neutral tools (Marakas et al., 2000).

The rise of machines with learning and autonomy has also been a source of concern. Not only are automatons increasingly treated as actors instead of neutral tools, but their non-neutral influence is often characterized in terms of risk. Stanley Kubrick's 2001 *Space Odyssey* (1968) had a line that has entered the collective consciousness as a prescient warning about giving control over to machines. "I'm sorry, Dave. I'm afraid I can't do that". 1989's *Kōkaku Kidōtai: Ghost in the Shell* involved an advanced artificial intelligence hijacking human cybernetic systems. In the 1999's *The Matrix* an artificial agent calls humanity a "disease" to the planet as it justifies the continual enslavement of humanity. The fear of giving control to these systems is well placed within contemporary culture. The backdrop to each of these examples is a growing control over humanity by non-living but nonetheless 'deciding' agents. They make decisions, but there is a theme of mistrust because they make different kinds of decisions. Is it cold and unemotional decision-making? Can machines have empathy for humans? These are inspirational questions but are not tackled by this research. This research does however explore questions of building and negotiating trust with a new type of agent entering our workforce. Debates about machines as neutral tools versus machines as actors that can engage with meaningful human-like actions such as learning and decision-making prove valuable for this exploration.

As a thesis about personalization and technology, this work uncovers a process of teaching machines to make decisions about users on their own. A look into emerging digital practices shows that these self-deciding machines are already here. We are increasingly giving agency and decision-making autonomy over our lives. For the most part, these are innocuous and suggestive. Algorithms suggest books based on our digital fingerprint for instance. Entire industries have been built out of algorithms that autonomously make guesses about users for the purpose of creating profiles that can be sold to advertisers. In this thesis, we explore cases of artificial intelligence and personalization used to support business processes through 'learning about users'. Often these algorithms make decisions without humans having any meaningful way to interpret *why* they made the decisions they did. This research looks closer at the design and implementation of technology-mediated personalization, and explores the journey of designers' understanding of said technologies as they build them and incorporate them into work processes. This thesis also highlights negotiation strategies being employed to manage or mitigate some of the emerging issues with these new technologies.

1.2 Setting the stage – Researcher background and motivation

This work is shaped by the author's ten years of chasing answers to elusive questions. What makes citizens happy? How do we know? How can our services improve this? What role can technologies play? This occurred across three sectors over this time, a non-profit Institute in Toronto that focuses on citizen-centred service delivery, then a large private-sector market research company servicing their public-sector clients across the Middle East. These organizations' service was to support businesses and governments in *knowing their customers* and using this knowledge to improve their service experiences. This knowledge about customers enables services to be reconfigured to meet diverse user needs or wants, including to support personalization. Giants in the industry of *knowing* customers had been established, many for over a century. These took the form of market research conglomerates and arms of management consulting companies. For these organizations to achieve scale they operated on an assumption that business processes had to be standardized. The tools used to *know customers* needed to be repeatable, predictable, and cost-effective. These took the form of surveys, often guided by models that track causes of outcomes like satisfaction (Parasuraman et al., 1988), or willingness to recommend to others (Reichheld, 2003). Other times it took the form of focus groups, guided by qualitative research and resulting in reports that highlight 'the voice of the customer'. These efforts proved valuable. Consider this excerpt from author notes when working with a country that made happiness a key national agenda:

> *"It all comes down to a smile", said the Director General of an executive office that fosters leadership across all government departments. A smile represented a milestone achievement for this government and its service delivery. It summarized a handful of years of coordinated effort and investment into improving staff politeness, knowledgeability, and confidence. I nodded approvingly because I had seen it unfold. I had joined this organization a few years earlier to support their efforts at improving customer-centricity by using technologies and measurement to better know their customers. My first task in this new environment was to support the executive office in the design, fielding, analysis and reporting of customer experiences in physical, digital, and telephone service encounters. A large citizen survey was deployed across all major government departments and a method called 'drivers analysis' using regression was conducted to find out what aspects of service delivery were most likely to be driving overall satisfaction. This is where I first appreciated the importance of technology in making tedious calculations about customers more accessible.*

For example, imagine a citizen rated staff politeness as 1 out of 5 but rated a series of many other questions as 4 or 5 out of 5, like timeliness, cleanliness of the facilities, or communication quality. And yet, imagine they score 1 out of 5 for the question, 'overall how satisfied were you with the service experience'. With a sample of one, it can be presumed that since they were happy about all other aspects except staff politeness, their low score of overall satisfaction is likely to be driven by this attribute. This is an easy enough decision to make manually, but we have long relied on statistical tools to scale this. Thanks to a set of technologies and analytical processes this calculation can be done over hundreds or thousands of respondents to find the key drivers of satisfaction across all citizens. It turned out for this government timeliness and communication scores, which had been the main focus of government service strategy for several years, were all performing strongly. Yet overall satisfaction was plateaued or even falling. The large survey uncovered a series of drivers around the quality of the staff. Questions like staff knowledgeability and staff politeness stood out as being in need of improvement. With the support of the executive office, departments made direct investments into staff training. Mystery shopping exercises were conducted to measure if staff were polite, if they could effectively direct customers to the conclusion of their service needs, and more. It did not take long for the departments to see the fruits of their labour. The happiness agenda had begun to take off across the region and service staff were building pride and confidence in delivering good experiences.

This was not the end of this government's service improvement agenda but underlined the impact of trying to *know your customers*, and the role of technology in supporting this. This research was shaped by these observations: greater service efficiency was gained by identifying channels of preference and improving channel experiences, and technologies played a fundamental role in this process. The way market researchers were gathering information about target customers had changed drastically even in the short time these observations took place. Surveys went from pen and paper to digital, allowing for faster data collection and fewer errors. This allowed researchers to begin the process of mining for insights faster, with less data cleaning. Surveys could be processed with increasingly advanced statistical tools, from frequency tables to cross-tables to step-wise regression to advanced random forest regression, allowing for more robust explanations of service outcomes.

This digitization coincided with a growing tension that could be felt across the market research industry. Those in need of knowing their customers to deliver more personalized experiences increasingly began demanding *better insights*, and *faster*. This amounted to a slow rejection of standardized research templates, models and frameworks. Slowly organizations requested actionability, which meant data and research needed to become more personalized to their needs and to their users, that is, more catered to the specific needs and challenges of the organizations and their customers. For a while, market researchers were convincing in the argument that such data cannot be collected affordably or quickly, and so clients acquiesced. But word was breaking out about the potential of *data science*, *behavioural science*, and *artificial intelligence*. And so, the stage was set. It was within this context that this doctoral research began. What role could these new technologies play in the context of organizations knowing their customers? What kind of opportunities will this give for service improvement? And what kind of relevant challenges? With the advent of artificial intelligence and modern software interfacing could we begin to revolutionize the speed and accuracy of our personalized experiences similar to how machines revolutionized factory work? Is there a machine-intelligence revolution taking place, and if so, could it be applied to the process of 'knowing' one's customers? The die was cast, this was the initial motivation behind this research.

1.3 Core definitions and concepts

Before progressing, this is an important moment to clarify core definitions that will be used throughout this thesis.

1.3.1 Defining personalization

Given the research context is around personalization, this is a good place to start. Having first been identified and elaborated in management and relationship marketing fields (Sunikka and Bragge, 2012), personalization is the tailoring of products and services to accommodate specific individuals. That is, to deliver the right services or products at the right time and place to the right customer. A second component of the definition is important to this research. This personalization and tailoring is based on what has been learned or is assumed about user preferences (Karat et al., 2000), such as customer information (Kobsa, 2000).

In the early years much of the literature on personalization began with debates about customization versus personalization. These debates slowed over time, settling on a view that both are personalization. Customization is the process of giving users more choice, with the agency in choice being in the hands of the users. Personalization was originally the opposite, strictly those cases where a business builds a different experience for users and decides for them based on what they know. This distinction has blurred with the rise of recommendation engines, which decide for users but give them the final say, as well as learn from their use. Personalization has also been called individualization, segmentation, target profiling, and one-to-one marketing (Sunikka and Bragge, 2012). Throughout this research any effort to give users a unique and personal experience, be it through giving them more choice, or through making those choices for them, will be considered personalization. Nonetheless, appreciation for these differences proved valuable through various stages of the research.

Value-generation through personalization. It is not controversial to say people have needs or wants. It is also not controversial to say that producers of goods and services have realized that meeting these needs or wants can create value, such as value that can be exchanged for capital or value that can be produced for the public good. As will be elaborated in Chapter 2, personalization therefore plays a role in the generation of a type of value because it enables the mediation of a wide range of needs or wants for a wide range of individuals. That is, there is a relationship between personalization and a form of value-generation.

Standardized personalization. To scale this personalization, or the tailoring of services to individual preferences, organizations make use of standardization. For example, learning about as many users as possible through statistical analysis and segmentation allows for a high-level view on groups of users. This is considered standardization because the needs of individuals are aggregated into profiles of similar people, where the needs of a group of individuals can be met at the same time or with the same product or service. Another way this is standardization is that the use of these tools and statistical techniques necessarily convert individuals into reproduceable data artefacts. Similarly, with mass customization, an organization can create a product with a dozen variations in colour to give users more choice, but the colours are still fixed and set within the context of a standardized manufacturing line. Understanding that personalization can be enabled through standardization informs this research.

ICT-mediated personalization. Recent advances in information-communication technologies (ICTs) (see the definition below) have created vast new opportunities for personalization (Salonen and Karjaluoto, 2016). Digital experiences can be fine-tuned, customized, and configured in nearly infinite ways. They are still standardized in that strict and reproduceable digital code is used to mediate the experience, but the degree of personalization is more powerful than ever. This ICT-mediated personalization also enables organizations to learn a lot more about users than ever before, by following and tracking their behaviour.

Given this, personalization that is made possible through the use of new technologies has occupied the attention of researchers of personalization (Fan and Poole, 2006). While an artisan designing a shoe to meet the tastes of a single client qualifies as personalization, this research is more interested in emergent mass personalization as mediated by ICTs.

1.3.2 Defining technology

If technologies play an important role in mediating personalization, it is also important to clearly define what technology means within the scope of this research. A traditional albeit superficial definition of technology are all tools, machines, instruments, infrastructure, communication devices, as well as the skills accumulated and required to utilize them (Bain, 1937). Since then, much debate has emerged around whether technologies are purely material, or whether they are inherently dependent on social conditions and context (Orlikowski and Scott, 2008). We will return to these debates in Chapter 3.

Information-communication technologies (ICTs). A subset of technology that occupies much of the attention of the fields of management are ICTs. This subset emphasizes communication and the interconnectedness of certain technological artefacts, be they interpersonal interactions or mass interactions (Mathur, 2017). This thesis is interested in how ICTs are used to configure and mediate personalization for organizations and their end-users, such as their customers, recipients, or citizens.

Algorithms. A further subset of ICTs are digital artefacts that are made out of well-defined programmed instructions and are facilitated by computers or mathematics to take some values as an input, process some value and then return it as an output (Yanofsky, 2011). Like the other definitions, there is debate about where an algorithm begins and ends (Knuth, 1996), as algorithms can include recipes for making pies. This research is explicitly interested in technological algorithms, hence the emphasis on programmed computer instructions. Generally, these algorithms are seen as requiring a specific execution and an ending, they do not run infinitely unless that infinite loop is part of a desired outcome. Computer programs are seen as configurations that include algorithms, and algorithms are further broken up into functions that facilitate the value processing (Yanofsky, 2011).

Artificial intelligence (AI). AI are ICTs that are said to exhibit aspects of human intelligence (Merriam-Webster.com, 2021), including representing knowledge, learning, language, sensual perception, desired motion, and social intelligence (Poole et al., 1998). These are executed using algorithms as opposed to through biological processes. This is a complex definition because human-like intelligence is also difficult to define. These definitions also change colloquially over time (McCorduck, 2004). Some tools like statistical regression were once considered advanced applications of computational intelligence, but after having become ubiquitous are often no longer considered AI. Thus, AI tends to be the most cutting-edge applications of human-like intelligence in machines, and this social-contingency tends to make defining AI controversial (McCorduck, 2004). For the purpose of this research, not all algorithms are treated as artificial intelligence unless they have a degree of autonomy in learning or executing their tasks (Poole et al., 1998).

Machine learning. Some AI is used to make guesses about sensory data, or others make predictions given certain pre-defined instructions, but not all AI modifies future predictions based on new observations. These are not said to be learning, thus are not *machine learning*. Alternatively, machine learning is a subset of AI that support learning problems in particular, and do so through learned experiences as conveyed through data (Mitchell, 1997). This implies that all machine learning is AI, and all AI include technical algorithms. But not all AI is machine learning, and not all algorithms are AI. These distinctions are important as we attempt to unravel the specific ways different technologies mediate personalization. This research elaborates how non-intelligent ICTs, generic applications of human-like intelligence and autonomy (AI), and machine learning algorithms each shape personalization in different ways.

1.3.3 Defining algorithmic explainability

How exactly do these ICTs need to be configured in order to create this kind of value? Designers make assumptions about user wants or needs, built on their understanding of what technologies can or cannot do (Leonardi and Barley, 2010). This is critical, because it underscores the importance of designer understanding, not only of users, but of technologies themselves. Technologies are not designed in a vacuum, they are borne out of, shaped, and distorted by human interpretation. This introduces an important complexity. What if we begin to use technologies that are difficult to understand?

Algorithmic explainability. Some algorithms are built using highly structured, routine, and linear logic. This gives designers an ability to understand why inputs lead to certain outputs. These algorithms can be said to be explainable. As will be explored in Chapter 6, some algorithms can be configured in a non-linear way, where components engage with data in a complex, live, and asynchronous fashion that makes it impossible to tell why a certain input led to a certain output (Samek et al., 2017). These are often called, 'black boxes' (Barredo Arrieta et al., 2020). As these tools make better use of increasingly dynamic social and behavioural data (big data), and as computing becomes increasingly ubiquitous, algorithms are becoming challenging to track. Yet despite this, the accuracy of these systems, their widespread adoption, and their autonomy continue to expand apace. This has led to calls for recognizing when algorithms are explainable versus not (Barredo Arrieta et al., 2020).

Algorithmic interpretability. When designers are not able to completely understand the inner workings of a series of algorithms, they may instead be forced to make interpretations about how they work and what they can do for users. Thus, while explainability may be an attempt to confidently understand why an algorithm made its decisions, interpretability emerges when the designer can only at best 'guess', such as based on their understanding of the underlying logic of the systems and the social data being fed into it. In Chapter 6 the differences between explainability and interpretability are explored with greater focus. The latter is not simply a poor-man's version of the former. Interpretability is explored as a degree to which the decisions made by an algorithm are understandable to actors like developers or users. There may be multiple interpretations of some algorithms, especially when using ever-changing data, or approaches that reveal outputs but not clues about what mechanisms led to those outputs. These can be distinct from algorithms that are explainable. The root words are helpful for understanding the difference. Interpretable algorithms can be interpreted, sometimes in different ways. Explainable algorithms on the other hand utilize functions that can be plausibly explained in a single way, with internal mechanisms exposed to human auditors. As we adopt more autonomous algorithms that are difficult to explain, there has been growing pressure to at least ensure systems and processes are interpretable (Samek et al., 2017).

Algorithmic transparency. Whether algorithms employed by an organization are highly explainable or not, easy to interpret or not, there are processes that can take place alongside their development that can aid the design and implementation of algorithms and can help designers be mindful or even accountable for risks. In Chapter 6 transparency is defined as and operationalized as the degree to which decision-making processes are revealed, including inner criteria. This is a focus on the process of retracing or reproducing algorithmic results. However, as discussed, some algorithms can be understood with one plausible explanation while others can have many interpretations and their inner logic not revealed. In these cases, transparency becomes about how to deal with algorithms that are less transparent (Waltl and Vogl, 2018). Unlike explainability and interpretability which focus on the algorithms themselves, transparency is a question of governance and reporting. Explainable algorithms can be highly transparent because retracing decisions can be straight forward. Algorithms that have many interpretations on the other hand may have these competing interpretations because they are inherently less transparent. In the face of less transparent algorithms, governance can shift towards tracing design decisions made, risk profiles, and auditable reporting. That is, transparency can be improved through a number of approaches, including making conscious decisions to adopt tools and systems that improve explainability (Samek et al., 2017). Another is to establish rules, guidance and governance around how algorithms are designed and utilized. This does not mandate the adoption of explainable algorithms, but calls for leaving a trail of decisions through which auditing can be facilitated. For example, in partnership with the Alan Turing Institute the UK government has begun to promote FAST track principles around fairness, accountability, sustainability and transparency (Leslie, 2019), and the Canadian government has begun mandating the use of a self-auditing tool, 'Algorithmic Impact Assessment' (Government of Canada, 2020) for internal departments that use AI. These do not ban the use of inexplainable algorithms, but heighten the need for reporting, tracking, and auditability.

Privacy risks. This research will regularly make reference to issues of privacy that emerge during the design and implementation of personalization algorithms. This is an area that has been well researched: the increasingly unsatiable desire for more customer behavioural data has led to concerns about user privacy being ignored in the name of service and profits, often called the personalization-privacy paradox (Aguirre et al., 2016; Awad and Krishnan, 2006; Karwatzki et al., 2017; Kokolakis, 2017; Weinberger et al., 2018). While the observations and arguments made throughout this research reaffirm these concerns, there is a stronger focus on more under-explored risks, such as bias amplification (see next).

Bias amplification. Literature around algorithms has made extensive headway in identifying important risks associated with bias. Bias can include statistical bias that over-represents findings, but throughout this thesis there is a more explicit focus on the research around the presence, detection, or mitigation of biased data, computation, or tool use that leads to discrimination for sub-populations of people, be it directly or indirectly (Barocas and Selbst, 2014; Venkatesh et al., 2016; Weller, 2019; Williams et al., 2018). Much of this has fixated on bias that is introduced by algorithms, especially as they make use of social data that is already biased. For example, training machine learning algorithms on social data means reinforcing gender stereotypes embedded within that social data (Hendricks et al., 2018; Kiritchenko and Mohammad, 2018; Leavy, 2018; Sun et al., 2020). In these examples, algorithms sustain human discrimination already present in data, or through the way they distort these data to make decisions. This can be considered technology-mediated bias. Less explored by the literature are biases that come not just from the algorithms and social data that are mediating experiences, but from decisions made by the designers themselves. The research in this thesis explores these biases with a great degree of intimacy. This is important considering the increasing number of autonomous systems being designed and deployed.

1.3.4 Defining agency and actors

A final class of definitions deserves special focus. This research observes how an organization develops diverse technology-mediated personalization processes and systems. As elaborated in Chapter 3 and in the introductory comments in this chapter, there has been an increasing metaphorical personification of technologies, with an attention to human-like characteristics around decision-making autonomy creating the impression of a seemingly rational actor. This has led to research questions around human and technology agency within the context of personalized service delivery. What then are actors or agents?

Agents and agency. Agents are broadly defined around a person or thing having a state of action or exertion of power (Merriam-Webster.com, 2020). While it is sometimes limited to the view of individuals, organization and management researchers have increasingly disassociated agency from the individual and instead towards a capacity or quality that stems from institutionally-defined resources, rights, obligations and social roles. These can include technological artefacts that are 'expected' to behave in a certain way within a human network and when mediating or facilitating human experiences (Abdelnour et al., 2017). Therefore, in a given service experience for example, there could be human agents such as users making decisions about what customizations to choose, designers who make decisions about how an experience should be structured, and technologies which have specific shapes and roles that distort and dictate service experience outcomes. These are all agents because they have a state of action or power on the overall experience.

Actors and their affordances. If agents within a network can be human or technical, are they all actors? A literature that has proven helpful in elaborating the boundaries between agents within a network is Actor-Network Theory. This will be more thoroughly reviewed in Chapter 3, but it is important to define early on because it shapes the research questions. Actor-Network Theory argues that nodes within a network are important for they have a capacity for power, action or influence over a relationship of actions within and across the network. This is similar to agency. These can therefore be technological or human. However, an actor view goes further in differentiating the specific roles and influences of these different actors, rather than treating them as all equal in footing. That is, technologies and humans may both be important actors in a network, but within a given experience different actors may convey different degrees of power or influence over the process. Some actors, for example, can have a license to dictate how others actors should behave, what processes they can use, and more (Leonardi and Barley, 2010). Beyond just dictation, other actors can *afford* different activities and possibilities. We will return to this concept of affordances in greater detail in Chapter 3.

It can therefore be said that within a value-generation network, like purchasing a good or receiving a public good, there are agents. Some of these agents have more power to influence the process than others, and viewing them as actors with different affordances can help with this. Some of those actors are end-users receiving a service, and others are service delivery staff or designers. As we make use of ICT-mediated actors within a value-chain, increasingly we outsource aspects of human decision making such as knowledge categorization, storage, recall, and more. Some of these technological actors are increasingly autonomous in that they receive information about end-users on their own, and then execute suggestions or recommendations to those end-users without any other humans involved. To the end-user, it does not always matter if a human was involved in the value-chain, and often they are unable to distinguish where decision-making was human and where it was algorithmic. That is, some actors are increasingly 'affording' value within a value chain independent of humans. Thus, ICT-mediated processes can increasingly be seen as actors affording value previously left to humans. This research explores how personalization in particular creates incentives for algorithms to be increasingly autonomous and intimate with value-creation with end-users directly.

Autonomous decision-making. Throughout this thesis there is reference to autonomy. For an algorithm to be considered autonomous, for the purpose of this research, it is said to be able to execute tasks or learning on its own (Poole et al., 1998). This can include advanced automation where tasks and learning functions are complex and interconnected. In these algorithms, the arrival of one piece of data can trigger a series of recommendation algorithms which in turn can trigger new services. Tasks and learning functions themselves can be dynamic and modify themselves to different data contexts. Automation is not limited to the most advanced applications however. More basic automation can determine when to run a task or learning function but have little or no dynamism within the calculations themselves. A simple script can be set to run when a certain field is filled for instance. These systems are still socially dynamic in that they are embedded within social networks and await live and asynchronous data. The degree of a tool's autonomy shapes the degree to which it affords or constrains interactions within social value chains.

End users as actors. One critical set of actors that cannot be ignored when understanding algorithms and their risks are the end-users. These actors receive the personalized system, be it a customized option when receiving a service, or a personalized recommendation, and this creates a value for them. This is, for example, exchanged for capital directly, their behaviour can be mined to be exchanged for capital between another party like an advertiser, or it could be considered a public good. End-users are referenced throughout this research as a secondary source of data.

Designers as actors. Another critical set of actors are the designers of the algorithms. This research dedicates much of its attention to this actor as decisions made when negotiating with algorithms introduce many opportunities as well as risks, and are a primary source of data.

Technologies as actors. This thesis explores the ways technologies are emerging as important actors in their own right. Whether these are metaphorical personifications or not, algorithms are being afforded autonomy in making important decisions about humans. This research studies not only the above human actors, but the technological artefacts themselves, which serve as another primary source of data.

1.4 Thesis research questions

There are many important actors involved in the functioning of an organization, each bringing different capabilities and enable different possibilities (Latour, 2005). As explored above, it has been valuable in recent decades to increasingly see technology as actors alongside people, albeit actors with different dimensions and properties. As will be elaborated in Chapter 3, these technological actors have a distinct underlying logic. While people carry with them subjectivities and a multi-dimensional logic of decision-making that escapes complete explanation, such as emotional decision-making that deviates from predictable utility-maximization (Demasio, 1994; Isen and Patrick, 1983; Pfister and Böhm, 2008), tools like ICTs are underpinned by explicitly mathematical logic or computational logic (Kallinikos, 2009). These distinctions are valuable for understanding how people and technology come together in the case of personalization. In recent years, personalization is being shaped by a proliferation of new technologies and data, especially machine learning and natural language processing (NLP). This thesis shows how these new technologies are changing the nature of machines as actors in our network.

Research question 1: In what ways do technologies, especially AI, and personalization influence each other?

The central argument of this thesis is that a drive for deeper personalization is resulting in the adoption of techniques like machine learning and natural language processing that have greater social relevance than technologies before them. In pursuit of personalization, these techniques and the use of dynamic social data to support them allow researchers and service designers to make better decisions about users than ever before.

Research question 2: Are these personalization efforts utilizing social technologies leading to new organizational actors? If so, in what ways?

Over three papers, this thesis shows that the efforts at bringing in more personalization into digital systems, especially through the use of techniques described as machine learning, has involved bringing in a new kind of actor into organizational operations. This actor has a different underlying logic from humans, but is increasingly making autonomous decisions about and on behalf of humans. The design of these systems involves adopting new strategies for dealing with technical features and for algorithmic interactivity. It involves being aware of new risks and challenges associated with these new technologies, including explainability and *technology-amplified bias*. It also involves being aware of the challenges and risks introduced by the core negotiator of the algorithms, their designers and *designer-amplified bias*.

1.5 Structure

This thesis has the following structure. In Chapter 2 we will situate personalization as both an evolving management paradigm as well as an explicit academic literature. We find that it is a rich and growing field, but has largely been held back by separating user-centric research streams from technology-centric research streams. Research around emerging AI-mediated personalization is also under explored. The field nonetheless shows excitement and promise about the impact AI-mediated personalization can have. This thesis intends to speak to these identified literature gaps. Chapter 3 establishes critical theoretical foundations that supported the research. This gap between user-centric and technology-centric research streams has also been identified across organization, management and information systems literatures. Attempts at reconciling this gap enable a research approach that is sensitive to both subjective user dimensions and objective/material technology dimensions. This chapter concludes with a discussion about the implications of emerging social and material interactions in the process of developing AI-mediated personalization. Chapter 4 summarizes the research journey and key methodological decisions made throughout. Chapter 5 was a paper co-authored with Antonio Cordella. A key motivation was to explore how, from a public value perspective, do emerging technologies like AI impact organizational capacity. A look at the development of algorithms and interfaces designed to assist employee selection shows the transformations and negotiations taken by the public sector. Personalization, we reaffirm, matters. It can increase organizational capacity-building by speeding up employee recruitment. However, the replacement of traditional processes with these new tools introduced challenges. Chapter 6 was a paper co-authored with Maha Shaikh and Antonio Cordella that built upon interesting discussions around the apparent persistence of black boxes even to developers. The dual nature of explainability and interpretability confounds even the developers of the system. An interactivity mapping approach is built over the course of this research. Chapter 7 broadens this mapping into a basic framework from which to understand personalization technical and sociotechnical complexity from the perspective of developers. Thus, across these three papers we show the importance of personalization, the particular qualities of social technologies, and emerging strategies for dealing with these social technologies.

Chapter 2: Literature review

In this chapter the literature around value will be considered, before moving on to the personalization literature.

2.1 A brief history of personalization, technology and related management paradigms

Personalization has emerged as an important field of research because of its apparent link to improved business or organization outcomes like customer satisfaction (Liang et al., 2006). There appears to be a relationship between personalization and the generation of some degree of value for individuals. This literature review situates personalization and technology as phenomena within a historical management trajectory.

2.1.1 Technological revolutions in value-generation

As explored, the literature defines personalization broadly as preference matching. If artisanal goods designed to meet specific customer needs qualify as personalized, then personalization has been with humanity since goods and services have been. Artisans must know their tools and materials well and must cater to the tastes or needs of their buyers. Shaping all of this, history is a tapestry of revolutions and new ideas. Many of these revolutions have material properties. Take the steam engine and later combustion engine as examples. We were transformed by the ability to produce rotational work from pistons. We no longer needed work-heavy tasks to be geographically limited to rivers and wind plains, or dependent on draught animals. These material innovations transformed the way we could structure society, industry, and commerce, and set the stage for the Industrial Revolution.

Standardizing manufacturing and the minimization of personalization. By the end of the 19th century and into the 20th a related revolution was taking place, one that limited personalization in an almost explicit way. Powerful new machines combined with a strategy of simplification and standardization led to industrialists like Henry Ford to mass produce goods that were previously out of the reach of lower-income consumers. This approach harnesses the efficiencies offered by the machines in the factory line and redirects human effort towards supporting and operating these machines. This had a profound influence on industrial strategy. Businesses were incentivized by being able to tap new markets of consumers if they are able to produce goods cheaply and efficiently. Standardization of goods became the name of the game. For services, a similar trend took root. Agents of service delivery may not have had new factory engines like the producers of goods had, but by delivering a unified and consistent service experience to customers they could harness predictability, efficiency in training, and could manage the experiences of the customers by standardizing them. These revolutions were not limited to the private sector. The public sector had begun to formalize what is known as traditional public management whereby bureaucrats including service providers are monitored in a top-down manner and expected to follow explicit rules and standards (Stoker, 2006). To prevent any biases that public servants may have when delivering services, discretion was minimized and public servants were bound to deliver services with as standardized a set of scripts, interventions, or routinized options as possible (Taylor, 2014). Across this period, personalization was at odds with a management philosophy that focused on standardization. The rise of technologies in the form of mechanized work allowed for vast productivity but reinforced a practice of minimizing personalization. The paradigm was long set. While there was money to be made in artisanal craft, the real wealth came from mass producing standardized goods and services.

Mass customization, tokenized personalization and segmentation. Over time machines and systems became more advanced, allowing for organizations to affordably offer diversity in products and services. Personalization was re-emerging in management mindsets in the form of standardized production like before, but with diversity introduced due to innovations and new efficiencies. Giving users choice in color became easy enough to enable through factory floor assembly, and this helped some mass producers attract new customers or retain existing customers. This type of standardized personalization was originally coined as *mass customization* by Stanley Davis's *Future Perfect* (1987) and was largely championed by Joseph Pine (1993) who defined it as "providing tremendous variety and individual customization, at prices comparable to standard goods and services" to enable the production of products and service "with enough variety and customization that nearly everyone finds exactly what they want." It has also been defined as "the technologies and systems to deliver goods and services that meet individual customers' needs with near mass production efficiency" (Tseng and Jiao, 2001).

It has been argued that these mass customizations as a means of personalization still fall within the standardization mindset (Zuboff and Maxmin, 2002) because the customization is largely token with minimal cost and still focus on top-down process where the source of value is being defined by the developers of services and products, rather than the recipients of service. While recipients are able to select from a wide range of options, these options are still in effect produced via managerial logic, and with an explicit aim for cost cutting and profit seeking. This aims to satisfy buyers with an increasingly varied product list and tailored service, which is more personalized, but most of the actual focus was on management principles and almost no analysis on the customers themselves. Many of the personalization techniques that have emerged from mass customization are quite token, such as simple gestures of personalization through the use of first names and welcoming personal greetings even if they are highly scripted (Li and Liu, 2017).

A related innovation enabled by technology such as statistical software and customer surveys was the rise of the modularization of service delivery along segments (Swaminanthan, 2001). The aggregation of individuals into groups of individuals allowed organizations to make better guesses about how to deliver unique experiences to some groups of customers over others. This is also within the standardization and managerial mindset, and could be seen as limiting choice to customers rather than giving them choice because marketing is used to encourage and spirit customers through products based on their demographics. Nonetheless, between mass customization, the use of token forms of personalization, and the rise of segmentation to more quickly understand groups of customers, a new mindset shift in management had begun to take root, one which challenged top-down assumptions about value-generation.

Web-based personalization. The literature around personalization went through an explosion in activity following the proliferation of new web-based technologies (Fan and Poole, 2006; Salonen and Karjaluoto, 2016). This is because these new technologies unlocked vast new opportunities for the delivery of personalized services. A first family of these innovations involved the development of adaptable user interfaces. The Internet and modern browser technologies have begun to emerge that allow for different pieces of information to be displayed for different users depending on a number of factors (Mobasher et al., 2000). This can include basic or token personalization, like giving users choice over website layouts or identifying user personal information, but it can also become quite complex, where entire windows or service experiences become available or unavailable depending on one's profile or context. A second family of these innovations are recommendation engines which collect information about users and then process them through algorithms to make guesses about what may be relevant for individual users. This can include a recommended list of similar items in a digital shopping cart, books similar to books you've read, or search results tailored to your geography and purchasing patterns.

Thus, we have reviewed how technological revolutions have shaped personalization. First, by discouraging personalization due to new vast value that could be generated through manufacturing standardization. Later, manufacturing prowess expanded and token options became available for customers and personalization began to re-emerge. This was aided by new techniques in statistics and customer segmentation. Then, personalization exploded with the rise of web-based services. With each wave of innovation, assumptions about customer needs and wants shifted from top-down managerial decisions to increasingly intimate measurements of real needs and wants through the use of systems that can efficiently provide personalization for users in a live and self-serve environment.

2.1.2 A brief exploration into the link between personalization and value

Before proceeding more formally through the personalization literature, it is important to reassess the link between personalization and value. Personalization assumes individuals have their own preferences. What is valuable for one is not necessarily valuable for another. Customer or citizen value has not been a well defined concept (Khalifa, 2004). Value can be multifaceted, multidimensional, and highly subjective. This thesis does not assume there is only one kind of value or that value is uniform across individuals. However, there is one type of value that may have a particular relationship with personalization. That is, the value that is generated through the use or experience of a service.

A strong tradition in applied market sciences focused on a unidimensional definition of value that was derived from neoclassical economic theory, which argues that individuals are rational and make choices that maximize their own utility (Sweeney et al., 1996). In this tradition, value is defined in terms of the performance of service delivery and can lead to trust and confidence (Institute for Citizen-Centred Service, 2012). This is a cognitive trade-off between wants and sacrifices, and is grounded by individual perceptions of what was received (Zeithaml, 1988). There is a direct link between personalization and enabling an individual to receive a positive service experience. The finding of their unique needs and mediating those needs is what creates this type of experienced service value. But if user preferences are deeply subjective and multidimensional, this implies that the value that is generated is done so when the recipient of that service actually experiences it, not before hand when a manager designs the system. This is because managers can only guess what users want, it is not until a user experiences it that their needs or wants are supported or denied.

Meynhardt (2015) stressed that there is an explicit relationship between a service and the user of that service. This sounds trivial, because it is. It is raised because management paradigms have long focused on practices that reinforce value coming from top-down managerial designers, rather than as being something intrinsically linked to the users themselves. Value is not produced exclusively by organizations, where consumers are simply passive users. A service value can instead originate from the interdependence between the recipient and the network of providers. This echoes a set of concepts introduced at the dawn of modern economics. Adam Smith (1776) argued for a differentiation between the utility a particular object has, and the power of purchasing other goods which the possession that object conveys. The former, "value-in-use", is not always aligned with the "value-in-exchange". For example, some objects which have the greatest utility in use for an individual can in fact have little or no overall value in exchange, and those with great value of exchange can often have limited value in use for an individual. One of the reasons for this is an embedded nature of value. What is useful for one might not be useful for others. This implies that one type of value is hiding within the interdependent relationships between individuals and providers, a value to be discovered. This also challenges top-down managerial standardization because it implies that there is a degree of subjectivity in what it means for something to be valuable. Delivering high-value products or services is not simply about the lowest possible cost with the highest possible profit. Dynamic and personalized service delivery can mediate personal value that otherwise gets missed by non-personalized services.

Moore (2012) considered this to be an important principle underpinning Public Value Accounting. Value cannot be detached from the societal context through which it is defined (O'Flynn, 2007). In the case of value derived from a service experience, it is this experience and perception that becomes essential to its determination (Vargo and Lusch, 2004). This suggests process becomes important rather than just outcomes when mediating this type of value (Vargo et al., 2008). This does not imply value-in-use is the only way to generate value. Bozeman and Sarewitz (2011) argue value is a complex and broad assessment of an object or set of objects, which could be concrete, psychological, socially constructed, or all of the above, and are characterized by cognitive and emotive elements derived from deliberation. Values are also privileged and embraced at different stages and different levels, with different values competing against each other (Baumgartner and Jones, 1991). Additionally, the way individuals prioritize values differs according to the degree to which it aligns with basic needs at one level and higher-order civil values at another (Nevitte, 2002). Thus, value can exist outside of the paradigm of service delivery and the value generated within the use or experience of a service. However, this experienced service does nonetheless appear to be one type of value, generated when personal needs or wants are mediated. And this has been sought after by managers for profit and by public service officials looking to promote broader overall public value.

There are implications. If service delivery agents are to mediate personalized value-generation, be it to generate profit or public value, then the source of this value needs to be recognized as coming from the subjective multi-dimensional needs of an individual and their interaction with the service experience. The idea of value being multi-dimensional is well expressed (O'Flynn, 2005; Cordella & Bonina, 2012; Chapman, 2003). Meynhardt (2015) for example has developed a framework for characterizing the mixed and varied dimensions that make up public value. Specifically, Meynhardt argues from a psychological needs perspective that this type of value is founded in individuals and is made up of subjective evaluations against basic needs, which are activated and realized by emotional states. These values are violated if a service seems unfair, unequal, and human dignity or respect is breeched. These basic needs act as subjective reference points to perceived reality, which can be 'felt' as discrepancies or deficits. If these basic needs are met, public value is created. If they are not, past experiences, routines, and more come into question and there is psychological discomfort. A social system becomes destabilized when many feel this discomfort. This has implications for personalization. Accommodating vastly diverse individual needs proves important, even if challenging.

Implicit within the realization that this type of value is defined by its use and is built from the subjective needs of different users is the realization that this value comes from a co-production between both sides of the service experience. To personalize effectively as much information about users is needed as possible. Co-production between users and designers helps improve understanding of preferences (Bovaird and Loeffler, 2012). From the bulk of the *public services* literature co-production is seen as an enhancement of value and service delivery by engaging with the service users and alternative service providers by meeting their needs or providing them with consultation (Osborne, 2010; Brandsen & Pestoff, 2006). In this view of co-production, the professional invariably retains control by designing and structuring the opportunities and mechanisms through which the service takes place (Simmons et al., 2007). In the field of *service management*, co-production in this context is not something that is a gift provided by a service provider. Rather, it is a necessary and fundamental aspect of the service encounter because of the inseparability of production and consumption of services (Vargo et al., 2008). Service science itself, according to Spohrer et. al. (2007), is the study of service systems of co-creation and co-value within complex system of integrated resources. From this perspective co-production is in fact unavoidable in services (Alford, 2008; 2016). Across the three papers in Chapters 4, 5, and 6 this thesis follows the co-production of services, and specifically the co-production of these services' personalization algorithms and features. Personalization was offered as a solution to the problem of not knowing what each individual user would find valuable. Users could be given choice, like customization, but also using recommendation engines their entire service experience could be altered and different from those of their peers. New social technologies are giving service designers more choice in how they can deliver personalized services, and in ways that can learn directly from users. Social technologies have emerged that allow for a seamless *learning* about user wants and needs. Personalization decisions are shifting from top-down learning and design to dynamic platforms that have autonomous learning and adaptive interfaces. That is, we are designing systems to engage with or learn directly from users about what they may want, for the purpose of creating value.

2.2 Personalization as an academic literature

Having explored the rise of personalization as a management paradigm and how it relates to the generation of a type of experienced service value, it is important to review the academic literature around personalization as the research within this thesis builds upon and enhances this work.

2.2.1 Defining personalization

Personalization is a growing field that crosses disciplines, including computer science and management (Adolphs and Winkelmann, 2010; Kwon et al., 2010). This thesis's journey through understanding this dynamic field began with definitions. Three influential literature reviews (Adolphs and Winkelmann, 2010; Salonen and Karjaluoto, 2016; Sunikka and Bragge, 2012) showed broad consistency in defining personalization as a process of preference matching (Chellappa, Ramnath and Sin, Raymond, 2005; Miceli et al., 2007; Montgomery and Smith, 2009; Sunikka and Bragge, 2012; Tuzhilin, 2009; Vesanen and Raulas, 2006). It can involve matching one's objective nature with one's subjective needs (Riemer and Totz, 2003). Personalized goods or services deliver personal relevance (Blom and Monk, 2003). This is largely facilitated by the collection of user data (Salonen and Karjaluoto, 2016). This data alone is not personalization however. It needs to be used to inform product or service changes for different users at different times. That is, personalization is the process of collecting information about users, to better know their wants or needs for example, and then utilizing this information to create a unique and improved experience. This seems to imply there is an actor who facilitates learning about customer preferences, and a technology that mediates those preferences with customers.

In terms of definitions, for over a decade the literature was split between company-initiated experiences, called personalization, and user-initiated changes, called customization (Arora et al., 2008; Bodoff and Ho, 2014; Fan and Poole, 2006; Ho, 2006; Montgomery and Smith, 2009; Sunikka and Bragge, 2012) A recommendation algorithm suggesting a song for someone based on their attributes was an example of personalization, while a user requesting a specific playlist they have curated was an example of customization. This divide has been less strict in recent years, as the use of data from user-initiated customizations are increasingly an active part of systems-generated recommendations (Kumar et al., 2019; Salonen and Karjaluoto, 2016) and vice-versa. It also turns out personalization appears to be beneficial regardless of whether it is user controlled or automatic (Fuertes and Lindsay, 2016). Personalization is understood as the tailoring of services to account for individual clients' preferences, allowing for some degree of user participation and co-production. The fluidity between these dimensions is explored in detail in Chapter 7.

2.2.2 A gap between user-centric and technology-centric personalization research

Below is a summary of a rich personalization literature. Many of the results found in this literature are unsurprising and/or self-evident. The purpose of this summary is to show the direction of the research and the ebbs and flows of optimism and criticism that have emerged as the literature has evolved alongside emergent technologies. The literature has largely been clustered as either user-centric or technology-centric (Adolphs and Winkelmann, 2010; Salonen and Karjaluoto, 2016), with the former focusing more on social dimensions and the latter focusing on material technical dimensions. User-centric personalization studies are interested in the impact personalization has on users themselves. That is, are personalization features worth investing in, when do they work or not work? For example, personalized ads have been found to lead to feelings of intrusiveness which ultimately can harm business performance (van Doorn and Hoekstra, 2013). There have been considerable privacy issues raised about unfettered use of data (Abu-Dalbouh, 2016; Aguirre et al., 2016; Awad and Krishnan, 2006; Jackson, 2018; Karwatzki et al., 2017; Li et al., 2018; Weinberger et al., 2018; Xiao et al., 2018). However, when done right personalization has been linked to increases in trust (Aguirre et al., 2016; Komiak and Benbasat, 2006; Mukherjee and Nath, 2007). Users are also more willing to provide profile data for web services than for ads (Awad and Krishnan, 2006). Effective use of personalization has led to increases in satisfaction (Devaraj et al., 2006; Ha et al., 2010; Herington and Weaven, 2009; Jiang et al., 2010; Liang et al., 2006; Piccoli et al., 2017) and service adoption (Krishnaraju et al., 2016). Personalization can also lead directly or indirectly to consumer loyalty (Chang and Chen, 2008; Che et al., 2015; Ha et al., 2010; Mukherjee and Nath, 2007). The effects of all of these positive outcomes are conditional (Sunikka and Bragge, 2012), such as on cultural effects, timing, or personal dispositions like motivation (Li and Liu, 2017). Preferences have often been viewed as static (Tuzhilin, 2009) when they should be viewed as contextual and in flux. For example, one's needs may shift according to timing (Bodoff and Ho, 2014), location (Li et al., 2014), and phases in the buying process (Lambrecht and Tucker, 2013).

Technology-centric studies focus on the technical implementation of personalization. A full understanding of design factors turns out to be critical to the development of successful web personalization for example (Salonen and Karjaluoto, 2016). Good design can increase trust (Li and Yeh, 2010) and loyalty (Chang and Chen, 2008) as well as shape preferences (Seneler et al., 2009), while bad design can inhibit it. These studies emphasize the technical construction of technologies that generate machine-readable artefacts representing persons and their needs. In the case of web personalization, this relies on the use of previously collected customer data (Arora et al., 2008) that can be inferred from consumer behaviour and transactions (Montgomery and Smith, 2009) such as search (Yoganarasimhan, 2015), product views and clickstream behaviour (Yang, 2010). The construction and extraction of user profiles is important for web personalization (Gajos et al., 2010), which can be used to infer complex dimensions of an individual like their personality (Arazy, 2015; Capuano et al., 2015), implicit needs (Chang et al., 2009; Qiu et al., 2018), or reputation and expertise (Martín-Vicente et al., 2012). This can be further supported by psychographic segmentation, or the aggregation of individuals into personality types based on answers to questions designed to model user psychology, as well as customer life-cycle stage assessment (Ahn et al., 2010). Recommendation systems have been increasingly interesting researchers (Li et al., 2014). These systems are incorporating complex social relations content that go beyond individual behaviours (Li et al., 2013). This can include community membership (Lee and Brusilovsky, 2017) and other clustering and causal mapping (Bernstein et al., 2019).

Methodologically speaking, user-centric studies are interested in the impacts personalization technologies have on users and utilize methods focusing on the voice of the user, such as through survey analysis, while technology-centric studies focus on specific technical features and often involves experimental design (Salonen and Karjaluoto, 2016). Few personalization studies emphasize both social as well as material lenses within the same investigation. As will be explored more thoroughly in Chapter 3, this thesis draws upon a body of theory that calls for both social and material dimensions to come together, and for measurement to be sensitive to these dimensions both for what makes them distinct from each other and for what relationships they have with each other. This is valuable for deepening our understanding of other gaps in the personalization literature. For example, personalization results are not guaranteed (Shen and Ball, 2009; Zhang, 2011). Personalization can be difficult to implement well (Chen et al., 2010; Fan and Poole, 2006; Sunikka and Bragge, 2012). Rather than viewing technologies as static tools that necessarily lead to certain outcomes, the literature has increasingly been studying personalization as a process (Vesanen and Raulas, 2006) that supports back and forth interactions (Adomavicius and Tuzhilin, 2005). Personalization requirements change as users do. Back and forth learning between users and systems has led to smarter interfaces (Gajos et al., 2010). Personalization has proven well-suited for heavily interactive service encounters (Mittal and Lassar, 1996). This research contributes to a deeper understanding of personalization by following the interactive evolution of personalization features by investigating both material dimensions of technologies from the code-first, alongside the voice of the user and the designer in interpreting and reconfiguring these material features. By bringing material and social views together, this work enrichens the personalization literature by deepening our understanding of the design and negotiation of emergent systems that are rapidly advancing personalized service experiences.

2.2.3 AI-mediated personalization: From opportunity comes risk

The literature is optimistic about emerging ICTs. Big data for example has been found to improve personalization and customization (Anshari et al., 2019). Machine learning provides for the ability to understand more from our users than ever before (Mullainathan and Spiess, 2017). The popularity of AI has been attributed to its high degree of personalization (Kumar et al., 2019), although has thus far been underexplored in the literature. Chapter 7 contributes to this research by comparing 34 individual cases of personalization, ranging from token to algorithmic, from adaptable user interfaces to machine learning. Chapter 5 follows a development team working with a government office in designing a new system to facilitate personalized employee talent search for HR managers. Personalization, as mediated by these systems, was seen as a way to modernize government leadership recruitment. However, from this case emerged questions about algorithmic explainability (see Chapter 6), fairness, transparency, privacy and bias. These observations corroborate concerns raised in the literature (Abdul et al., 2018; de Laat, 2018; Marino et al., 2020; Zerilli et al., 2019) including by international agencies (OECD, 2019), that influential new systems sometimes employ approaches that are not transparent or explainable. Consider the peculiar case of Stanford University experiments with deep neural network modeling to predict human sexuality. Features from pictures of faces from over 35,000 people were extracted and entered into a logistic regression model aimed at classifying sexuality. The research found that the model was able to predict human sexuality far better than humans, at a rate of 91% for men and 83% for women, versus 61% for men and 64% for women for human judges (Wang and Kosinski, 2017). This study led to a flurry of discussion in the computer science field (Gasser and Almeida, 2017), with researchers struggling to explain how the model was able to be so much more accurate than humans. The problem with some models like these is once built, they do not reveal explanations behind their answers. Instead, they often only return a result (Gunning and Aha, 2019).

What are the implications then if some models appear to be consistently more accurate than their human counterparts at deeply social and individual questions like sexuality? What are the implications if this technology becomes readily available to employers for example? Will this reinforce representation issues for sexual minorities? Or on the other side of the same coin, can these tools be used to find gaps in representation to improve them? If tools can be used to detect sexuality, can they be used to detect other complex social contexts? If we do not always understand how they come to the decisions they do, what safeguards do we have against wrong decisions?

These technologies are already being utilized to inform life-altering decisions. In 2017 the American Law Institute approved a proposal that aimed to modernize evidence-based decision-making in the penal code (Villasenor and Foggo, 2019). Legal systems are increasingly adopting actuarial algorithms to determine risk, and this is being used to determine sentence length. For example, in 2013 Eric Loomis pled guilty to charges of fleeing a traffic officer and operating a motor vehicle without the owner's consent. Loomis' personal details were inputted into a risk assessment system known as Correctional Offender Management Profiling for Alternative Sanction (COMPAS), which declared Loomis a high risk of recidivism. Loomis and his legal team appealed to the Wisconsin Supreme Court on the basis that the proprietary software did not reveal the reasons for its decision and thus this was against due process. The Supreme Court in 2016 ruled against Loomis, arguing that "if used properly with an awareness of the limitations and cautions . . . consideration of a COMPAS risk assessment at sentencing does not violate a defendant's right to due process" (Wisconsin, 2016). Thus, inexplainable algorithms are here and are consequential. Transparency has emerged as a critical dimension of AI-governance in the literature in response to issues like these (Abdul et al., 2018; Marino et al., 2020).

Bias is another set of risks identified by the literature. ICTs make all decisions based upon a core and underlying computational logic and are unable to pass judgement about bias. Advanced machine learning for example depends on bias. Machine learning seeks to learn from patterns and these patterns themselves can include and reinforce bias (Leavy, 2018). A widely recognized example of this comes from natural language processing tools that have been trained from general text corpora (Sun et al., 2020). These tools are trained from billions of publicly available texts, such as online encyclopedias, and identify relationships between words such as how likely a word is to appear alongside another word. We use this training to find synonyms or make guesses about similarity across texts. However, words like 'babysitter' and 'girl' tend to get reinforced, or 'CEO' and 'man'. In the literature, machine translation applications of English to Hungarian and back again found "He is a nurse. She is a doctor" to consistently became "She is a nurse", and "He is a doctor" (Douglas, 2017). A computer vision application was found to incorrectly identify agents as male when they were female because a computer was present in the picture (Hendricks et al., 2018). These biases range from degenerating, to stereotyping, to recognition problems, to driving under-representation (Crawford, 2017). Social data can be injected into rules-based logic, but the algorithms will apply mathematics blindly without awareness or sensitivity to bias present in the data. Biases in effect inform the calculation. Social data are necessarily biased. If bias exists in English language out there in the wild, then bias will exist and even be amplified in social technologies that are trained by it (Sun et al., 2020). These are risks that are top of mind for researchers and designers of personalized services (Leavy, 2018; Williams et al., 2018).

Another set of risks identified by the research is a possible incompatibility between ubiquitous consumption of personal data and *privacy* (Abu-Dalbouh, 2016; Aguirre et al., 2016; Awad and Krishnan, 2006; Chellappa, Ramnath and Sin, Raymond, 2005; Jackson, 2018; Karwatzki et al., 2017; Weinberger et al., 2018). Researchers have observed a number of possible solutions and strategies for enabling personalization while protecting privacy. The most promising examples include a focus in limiting the collection of unneeded data such as screening out what is called personally identifiable information (Schwartz and Solove, 2011) and reducing the possibility of privacy breaches in the first place (Li et al., 2018).

Dimensions like transparency, bias, and privacy contribute to whether or not users trust experiences generated by algorithms (Marino et al., 2020). Users judge algorithms based on factors like intention and competence when building trust (Devitt, 2018). It turns out users are tolerant to mistakes and desire getting more involved in the correction and training of algorithms (Stumpf et al., 2007; Thomaz and Breazeal, 2008). Over time as trust between users and algorithms build, users are willing to increasingly share decision-making autonomy (Marino et al., 2020). As this shared decision-making grows, users become willing to let algorithms make decisions entirely on their own. Thus, trust is shaped by co-production and through iterations of systems with direct user input, and are related, iteratively over time, to the increased autonomy of machine thinking.

Bias is always present in any technological system that uses social data because data contains and encodes human biases. At another level, algorithms can vary in terms of the degrees to which they offer explainability. Sometimes this explainability comes at the cost of accuracy (Guidotti et al., 2018), because powerful systems are available that perform exceptionally well on structured tasks, but the underlying reasoning sometimes escapes scrutability. Also, the most powerful personalization algorithms would logically have access to the most personal data possible. Users, organizations, governments and developers engage in a negotiation to determine the right balance between privacy and accuracy. The design and use of these systems become a negotiation of trust, and as trust increases users are willing to give greater degrees of decision-making autonomy to these systems.

Chapter 3: Theoretical foundations and discussions

As discussed in the previous chapter, the literature on personalization has been characterized by a divide between user-centric and technology-centric investigations. This closely reflects a methodological divide in studies concerning the role of technology across management, organization, and information systems literatures. Similar to personalization research, these fields have been characterized as having been split between two streams, one that focuses on the interpretations of users and the other that focuses on a perceived deterministic nature of technologies (Orlikowski and Scott, 2008). As will be elaborated below, the first stream has been criticized for delegating technology down to just another artefact without features or properties worthy of explicit analysis (Attewell and Rule, 1984; Dewett and Jones, 2001; Huber, 1990) while the second stream has been criticized for downplaying the role of users of technology (Barley, 1986; Zammuto et al., 2007). From this debate attempts have been made to create a third stream of research to reconcile the material properties of technologies with the dynamic and subjective properties of individuals and society. This attempt at reconciliation proved valuable for this thesis. This third stream has been called sociomateriality (Orlikowski, 2007). Research in this thesis draws from this debate to explore the interaction between material dimensions and social dimensions to better understand the ways technologies are mediating increasingly social interactions, and in particular, how they can mediate personalization.

3.1 The sociomateriality of technologies in our workplace

3.1.1 First research stream: user-centric operationalizations of technology

Technology, speaking from the field of information systems, is a healthily contested concept (Orlikowski and Iacono, 2001), and debates over definitions can help elaborate how technology is operationalized within this research. To many, technologies are among the institutional artifacts that individuals find themselves surrounded by (Orlikowski and Scott, 2015). Organizations and individuals engage, perform and find meaning from their interaction with these structures, and at the same time, interdependently shape these structures as well. An organization or individual becomes itself through this interplay between technologies and institutions on a metaphorical stage (Tsoukas & Chia, 2002; Langely & Denis, 2006). This perspective has classic roots, going back to Erving Goffman (1959) who, in a dramaturgic sense, described individuals as actors performing roles on a stage to an audience. In this sense, technology plays the role of the props. This research stream emphasizes user-to-user interactions, as depicted in Figure 3.1. This is a highly abstract figure that is meant to convey an actor-centric view of the world, or more specifically, a human-to-human view of the world, where props and settings are but contextual background information for the more important human actors and their relationships with other humans. Human-to-human interactions are primary, and human-to-prop/setting interactions are secondary.

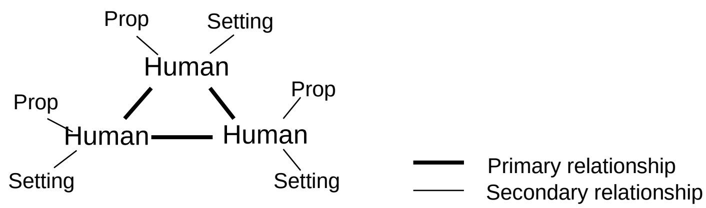

Figure 3.1. Actors and their settings in user-centric research

In this tradition, props and other contextual settings surrounding an actor shape and influence their decisions and meaning making. This context can include geopolitical realities, socially-defined institutions like cultures and more (Goffman, 1958). Human actors may use props to support their interactions and meaning-making with other actors and society around them. Many interesting studies have been built upon this tradition to identify deep and meaningful patters of social construction and interpretation between humans and objects like technologies in the organizational context. Consider Sauder and Epseland (2009) who explored the influence of technology through technology-mediated scoring and ranking of organizations and how these practices of scoring had an impact on the way performance in the organization was understood by its employees. That is, behaviour changed because of shifts in employees' internal perceptions about their roles, behaviours, practices and surveillance. In this study human perceptions became a central lens from which to understand changes in organizational behaviour, while the technology that mediated the ranking was in the background.

Human actors and objects like technology are not treated as analogous in this research stream. As depicted in Figure 3.2, in individual actor-to-object relations an actor ascribes value to the object while the object possesses physical properties which may or may not be relevant to the actor's social construction when making use of it (Arnold, 2003). In both the below figure (Figure 3.2) and the above figure (Figure 3.1), the primary force from which value is derived comes from the human actor. In actor-to-actor as well as actor-to-object relationships the actor constructs social meaning (Searle, 1995, 2008). In this social construction, an actor may choose to build meaning from physical properties of the objects, but they may also not. This view remains partial and incomplete, however. There is more to the story between an object and its value as ascribed by an actor. This will be completed when reviewing what objects can *afford* within value networks in the third research stream.

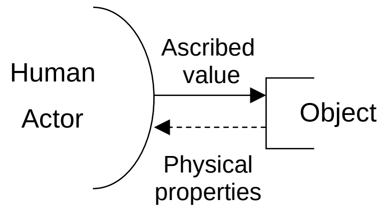

Figure 3.2. Human actor-to-object value ascription in user-centric research

Applying these depictions to personalization, an actor such as an artisan or service designer can build an object that they believe will represent an ascribed value to their customers. They can incorporate social dimensions in terms of understanding customer needs when developing products or services and could use the object as a tool and its physical properties to accomplish this.

Challenges:

This research stream has been criticized for delegating technology to the background of analysis in an almost 'taken-for-granted' manner, with properties that are 'foregone conclusions' (Attewell and Rule, 1984). Thinking again about Figure 3.1, objects like ICTs tend to disappear from view where user-to-user interactions take primacy. They can be presumed to be unproblematic once built and installed. Technologies are treated as independent variables in a calculation about human decisions (Orlikowski and Scott, 2008). The inner workings of these technologies and the dependency of these tools on interaction with users can be treated as 'black-boxes' that are not worthy of fully understanding (Latour, 1987). They can vanish from our view as social researchers behind our preoccupation with human-derived social constructions (Orlikowski and Barley, 2001).

The implication is that technologies are necessarily stable, settled artefacts that can be passed easily from one actor to the other without challenge (Latour, 1987). However, studies across the literature point to disparate, fragmentary, and inconsistent findings in terms of technology adoption and use (Arnold, 2003; Attewell and Rule, 1984; Dewett and Jones, 2001; Huber, 1990). It turns out technologies are neither *necessarily* stable, nor do they get uniformly adopted. This is not to say technologies cannot have stability, some mainframe systems have been used by enterprises for decades with little change. For example, the US Treasury Department/Internal Revenue Agency used the Individual Master File system, and the Veterans Affairs Department used the Personnel and Accounting Integrated Data system, for over 50 years (Moore, 2016). To an extreme, spoons or chop sticks have been relatively unchanged for millennia. Yet, software can change overnight.

A number of explanations emerged for these gaps in understanding, why technologies are sometimes stable and other times not. It has been argued that theories about technology which place them in the backdrop of our focus came before the emergence of modern technology (Huber, 1990). Technologies have become increasingly interactive (Goldin et al., 1998), ubiquitous, and intelligent, and as they do they are taking on more roles in our organizations than just being an object in the background. As will be explored in the next stream, it turns out that technologies have a few properties that set them distinctly apart from other contextual settings surrounding a human actor, like social institutions, and it turns out these particular properties have consequences for social experiences and social performance the more they mediate these experiences. The role of technologies needs to be brought forward.

3.1.2 Second research stream: technology-centric operationalizations of technology

Contrary to user-centric streams of management research, technology-centric streams have focused on the inner workings of technologies themselves as distinct entities that interact with organizations, and become particularly relevant or salient during moments of design, diffusion, implementation, deployment, adoption, use or breakdown (Orlikowski and Scott, 2008). These technological artefacts have qualities that set them apart from other institutional settings and have consequences that should be formally understood, especially given the increasing degree to which our organizational practices are mediated by technologies.

Two key attributes that set these artefacts clearly and distinctly apart from other social artefacts are the fact they necessarily *simplify* social information into standardized forms and have a tightly *closed* underlying logic structure. Functional simplification is a theme from the literature that emphasizes that ICTs standardize information from the social world into discrete and reproduceable objects (Kallinikos, 2005; Luhmann, 2005). For information to be machine-readable, they need to fit a logic that necessarily condenses reality. Even advanced context like voice or human gesture-based information needs to be condensed into code.

The centrality of the human in user-centric streams of research makes sense because human decision-making eludes our understanding. The way humans encode and calculate using information is not as well understood as it is with machines (Wang, 2003). Humans as unpredictable actors became a fixation of our social investigations in part because a long attempt to prove that human 'rationality' exists has been fruitless. Humans engage in decision-making that necessarily deviates from 'rationality' on a regular basis. A commonly accepted, even if not fully understood example is emotional decision-making. Empirical evidence strongly shows decisions can be influenced by our affection or emotional connection to our networks which can lead to decisions that deviate from utility-maximization (Demasio, 1994; Isen and Patrick, 1983; Pfister and Böhm, 2008). This is modeled in Figure 3.3. The arrow represents information about the world that is arriving to a biological being in the form of sight, smell, sound, touch, etc. This sensory data is then processed through neurological processes that largely escape scrutability (Wang, 2003). In this case, information is general stimuli from the natural environment, while sensory information is the neurological information passing from huma organs to the human brain and in many cases towards decision-making.

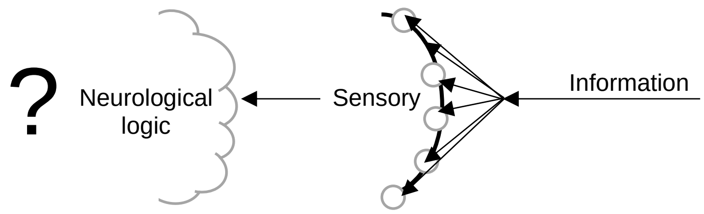

Figure 3.3. Human information encoding, an inscrutable logic

Unlike when investigating humans, with machines we start off on a stronger foundation in terms of understanding. The ways a machine receives information at a sensory level is well known. Data like text, numbers, visual pixels, and sound wavelengths can be digitized with ease. The process of encoding this information and making it available to algorithms is also well known and highly standardized. It can be said that technologies are judged by their ability to retain reproducibility as processes, or to be platforms of predictability (Bloomfield and Vurdubakis, 2001) for otherwise dynamic and ever-changing social realities. That is, we can utilize technologies to create durable (Latour, 2017) and reproduceable models of reality. It is the harnessing of this predictability that lets machines create repeated and sustained value. This is modeled in Figure 3.4, where sensory information is processed in a much more standardized, coupled, and understandable way.

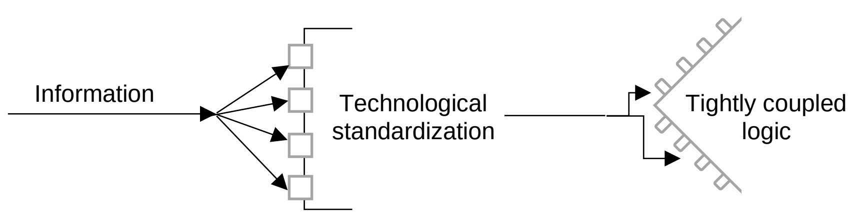

Figure 3.4. Technology information encoding, a scrutable logic

A dimension that enables this predictability is the tightly closed nature of the functions that are run over these simplified and standardized data objects. The complexity of the world is demarcated into an operational domain by reconstructing it into a simplified set of causal or instrumental relations (Kallinikos, 2005). Algorithms then engage in computation based upon these standardized versions of reality and do so by passing objects through mathematical and conditional logic (Kallinikos, 2009). The moment algorithms are run the social context surrounding the experience disappears. The algorithm cares not for time, place, or meaning-making; in a fraction of a second tightly linked logic runs mathematically, from one link to the next, over the simplified objects of social reality. Computations occur within a closed and almost deterministic paradigm. This determinism transcends time and place, and is a logic that is universally accessible. Before proceeding, care should be taken around the word 'determinism'. The literature does reference this word, both from advocates of technological-realism as well as its critics (Orlikowski and Baroudi, 1991). However, determining whether an algorithm is fully deterministic is debateable. Some algorithms can involve mathematical logic that is irrefutably deterministic, like basic algebra. However, others can make use of 'random' values, such as Monte Carlo simulation. This 'randomness' implies that these algorithms are not inherently deterministic because that random factor will change every time. Even in this case, there is also debate about the 'true randomness' of the underlying random number generation. A well-known challenge in computer science is that computationally-generated randomness is 'pseudo-random', and efforts at creating true randomness rely on drawing from natural phenomenon such as thermal noise, radioactive decay, and electronic oscillation (Wei and Guo, 2009). Whether algorithms involve pseudo-randomization, true-randomization, or no randomization, these algorithms still follow explicit instructions and cannot deviate from them. Throughout the rest of this thesis, references to determinism will emphasize that algorithms have *seemingly* deterministic properties, and that this differs from human decision-making which is not limited to instructed computer code.

Researchers in this stream treat technologies as independent entities with discrete properties and these properties have consequences. Consider Figure 3.5. Because social reality needs to be converted into a simplified form in order for the tightly coupled logic loops to function, context is necessarily stripped away (Ciborra and Hanseth, 1998). This can have distortionary implications. By enforcing artefacts stripped of context, true individual choice is limited when embedded within a local context. This has been shown to influence work practices. Choices about what to standardize and what not to standardize produce advantage for some and suffering for others (Bowker and Star, 2002). The more experiences are mediated by these technologies which necessarily strip out context, the more opportunity for distortions.

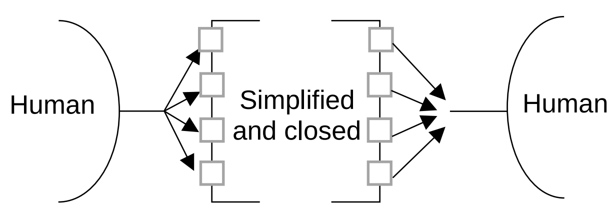

Figure 3.5. Potential distortion in object-mediated interactions

For personalization, mindfulness is needed in the design and monitoring of products and services as any number of large or small design decisions could distort the experiences of some in favour of the experiences of others. As these technologies utilize reinforced logic and machine learning, where categories and standardization choices are increasingly being made by the machines themselves, new pressures are emerging for designers and service managers to ensure distortions are mitigated.

Challenges:

The previous section highlighted criticisms that have been given to user-centric streams of research for being unable to explain gaps in technological adoption (Attewell and Rule, 1984; Dewett and Jones, 2001; Huber, 1990). Despite making an understanding of the inner workings of technology a main research aim, technology-centric streams of research have so far also been unable to explain gaps in our understanding about adoption as well (Grabowski and Roberts, 1996). By focusing only on the technologies themselves, and treating actors simply as passive users of the system, this stream of research has overlooked the importance of humans in shaping how technologies are designed, perceived, adopted, and more.

3.1.3 Third research stream: Sociomaterial operationalizations of technology

A third stream of organization, management and information systems research has been proposed to reconcile the gaps of user-focused research that downplays technological features, and technology-centric research that downplays the role of the user. Labelled sociomateriality, this paradigm calls for a recognition of both as interdependent when investigating organizations and the technologies they use to carry out their activities (Orlikowski and Scott, 2008). Both human actors as well as the material objects that surround them have critically important contributions for how technology is used and understood. Studying these phenomena means adopting methodologies sensitive to a) social construction of users, b) functional simplification/closure of technologies, and critically, c) how they interact with each other.

A shift this produces is a call for a two-way understanding of relevant social and material entities within a social web when studying a phenomenon. Rather than only studying the actor and treating objects as props, or only studying technology as deterministic objects, investigations can map social and technical entities together by appreciating what happens to users when facing seemingly deterministic properties, and how users in turn change or reconfigure these properties. In a related view, fittingly named Actor-Network Theory (Latour, 2005) actors in a network include humans as well as machines. This area of research further clarifies actors that are involved in the mediation of or translation of experiences as 'actants.' As abstracted by Figure 3.6 below, whether an actor is technical or human they can be seen as actants in a complex web of relations and interactions rather than background objects. Organizational or social practices can thus be understood relationally and as a nexus of connections (Nicolini, 2013) between social and material actors.

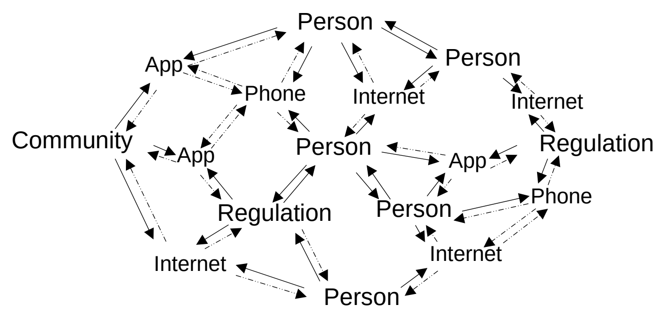

Figure 3.6. A network of actors with strong and weak connections

Actor-Network Theory does not mean to suggest objects have the same importance as humans. It does however suggest they do have particular contributions to their networks. It also suggests that technology and other social entities like institutions can co-exist, but that they too have different dimensions or properties. This mirrors technological enactment theories (Fountain, 2001), which recognize institutional artefacts like social routines, laws, and other institutions that make up a backdrop for decision-making around adoption, but that these influence adoption in different ways from technologies. Investigating enactment means creating a relationship web of social as well as technical artefacts, and treating them as distinct in function and logic (Cordella and Iannacci, 2010; Fountain, 2001). Purely social institutions do not have features of mathematical standardization or logic closure for example.

3.2 Actors of significance

The distinction between these mathematical objects and social artefacts goes beyond the two-way difference in their mediation of relationships. The objects in actor-networks can take advantage of reliability, scale, and they can be repurposed countless times. These nodes, unlike humans, seem to have few limits to the numbers of connections they can make. A human can only convey so much information to so many networks at a time. Digital algorithms, on the other hand, can at times process much more information than humans, like searching through hundreds of millions of articles at once, and can mediate more interactions at a time through scalable and standardized protocols.

When reviewing the first stream of research, user-centric research, users were seen as ascribing value to objects. At the same time, the objects had physical properties from which they could frame their understanding of its use or value. This third stream of research supports coming back to this phenomenon with greater analytical precision. An object's *affordances* are what they can do, based on the perceptions of the agents around that object. For example, users carry preconceived notions, shaped by culture and history, about what certain tools can do. Individual developers within a single team may have diverging views. One actor is able to configure tools to meet their set of perceived possibilities, but if a second actor carries different perceptions, that object now has an expanded set of outcomes that it can afford. Thus, what an object can do is determined by complex subjective negotiations contingent on individuals' socially-derived understanding of its seemingly deterministic properties (Leonardi and Barley, 2010). Affordances become a critical lens from which to observe the boundaries between objects and people, because it is where researchers can peel away at designer notions of objects and can help interpret the way objects are mediating value in a live network.

With affordances and Actor-Network Theory, this work can begin to answer the second core research question: is personalization leading to a new type of actor within the organization? In the literature, actants are differentiated based on the degrees to which they influence relations and experience in networks (Latour, 2017). The sheer scale of personalization that is enabled by new systems, and the rise of autonomous systems that are disintermediating human involvement, suggests that important new actors are indeed present within our value chains.

The boundaries between human and machine agency is increasingly blurring, and webs of actors within organization settings are growing complex (Suchman, 2007). Many actions are mediated by discrete algorithms which distort experiences through functional simplification and closure, and increasingly many actions are mediated by multiple algorithms connected to each other without a human involved. Algorithm-to-human, or algorithm-to-algorithm interactions occur in a digital manner. Reality is converted to data objects. The 'webs' or 'ecosystems' of relationships can be mathematically represented between objects with numbers representing relations for example. Human-to-human or human-to-algorithm interactions on the other hand cannot be readily recalled using repeatable data objects because they are influenced by social, psychological, affection and feeling, and other dimensions that are difficult to capture. This has implications for research: social phenomena within these settings cannot be fully explained without a two-way investigation that bridges machine information simplification and closure on the one hand and social construction on the other. As these machine actors take on increasingly distorting and influential roles in webs of interaction, investigations into these settings will benefit from an appreciation for social as well as technical agencies (Latham and Sassen, 2005).

Not only do machines encode information and conduct calculations using this information with a logic that can be fundamentally different than human information encoding and decision-making, there is distinct interactivity when they come together within an exchange. Humans react to machine material properties and can reconfigure them as needed. As machines get built, iterated, and integrated into work practices, they get reconfigured by human actors (Suchman, 2007). Increasingly as these machines do more than just recall and transmit data but begin to make inferences about users, humans are adapting new understandings of these systems and continue the reconfiguration cycle.

As these machines increasingly accommodate complex social data and employ techniques that are beginning to lose explainability and reproducibility, themes that emerged across this thesis research, humans are again adjusting their understanding and reconfiguring the systems as needed. But the process is not linear and simple.

In terms of informing analysis, knowing that machine sensing, encoding, and calculation can be assumed to be within an underlying computational logic, we can observe how social realities exist within a closed and simplified space. Computational processes can be mapped carefully by looking at these logic functions. Humans as actors, on the other hand, requires a different lens. As discussed when reviewing the first stream of research within management, organization and information systems literatures, humans define meaning from a network of experiences and relations. Within this network human awareness plays an important role. One's awareness of their place as an actor in a web of relationships with other actors means one's knowing or awareness matters for decisions like adopting technology (Danner-Schröder and Geiger, 2016). When human-to-human, these relations may be considered routines, relations with degrees of repeatability and predictability. We draw upon these routines when making decisions. With human-to-material experiences, humans build from assumptions about what a technology cannot do, *constraints* (Leonardi, 2011) and what they can do¸ their *affordances*, because they combine material potential with perceptions or awareness of that potential (Leonardi, 2013). One's perceptions are shaped by both material features that are inescapable, as well as social interpretations that have been reinforced by social interactions (Leonardi, 2009). We can map human reactions to technological features and behaviours, and how they reconfigure their assumptions about what can be afforded. This human-technology reconfiguration happens live and is emergent in practice (Pickering, 1993).

3.3 Sociomateriality and personalization

In the previous chapter personalization and the mediation of individuated experiences were similarly defined as phenomena that must be experienced (O'Flynn, 2007; Vargo et al., 2008; Vargo and Lusch, 2004). For ICT-mediated value, value-creation does not occur until a recipient and a technology come together. Designers of personalized services make assumptions about user wants and needs (preferences), then configure systems to accommodate these preferences. This design stage can involve rapid cycles of reconfiguration between designer and technology until reproduceable and reliable performance is achieved. Consider Figure 3.7. One actor of consequence, the designer of the service, draws from their own network to infer knowledge about user preferences. This could be from colleagues, best practices, past experience, etc. They can then build upon these assumptions about preferences and combine this with their knowledge about what ICT objects can do and cannot do. Designers can reconfigure until these ICTs until they are able to successfully mediate personalization to a network of users. For example, service designers can design an ICT-mediated service internally with early testers before releasing to the public. Once in the public and accessible to users, these ICTs can begin to be seen as successfully mediating value for users. The designer can continue to draw from this network including its users and the reconfiguration cycle can continue as preferences evolve. A network of value then exists between an ecosystem of users, designers, and matured ICTs.

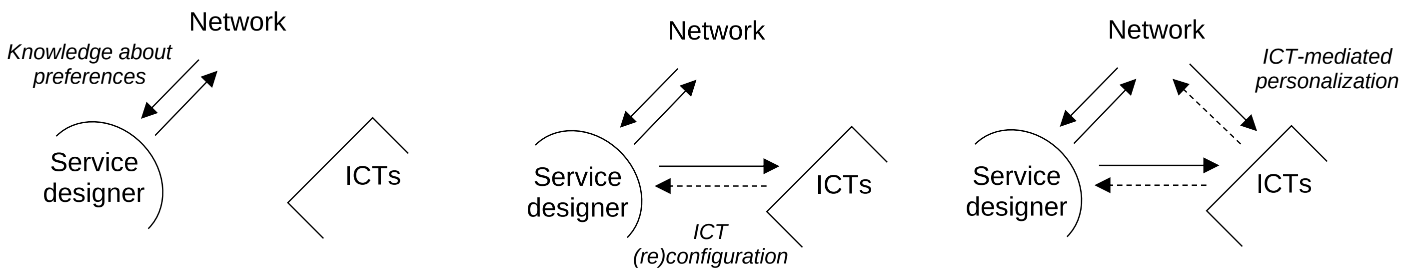

Figure 3.7. Personalization reconfiguration cycle

A key argument of this thesis is that a new type of actor within organizational settings is emerging. New technologies require a stage of design and (re)configuration just like older technologies, but are able to go beyond dependence on service designers. Designers can continue to configure and reconfigure the systems until they begin to take on greater decision-making agency in not only the delivery of personalized service experiences, but in initiating (re)configuration cycles of their own with their users (Marino et al., 2020). For example, machine learning allows systems to learn from users directly. Informally, machine learning algorithms are algorithms that learn from data (Goodfellow et al., 2016). Machine learning can be formally described as occurring when a machine can differentiate between experiences E that improve task T as measured by performance P versus experiences E that decrease task performance (Mitchell, 1997). The more experiences that can be fed into such a system, the better it can perform. The growing availability of data allows for greater experiences E, and the advancement of algorithms improve performance P, allowing service designers to expand the number of tasks T that can be tackled. A feedback loop of learning experiences can also exist between a user and algorithm independent from the designer, where experiences E lead to performance measurement P and use that to retrain itself. The agent of service delivery thus can shift from human-to-human, to human-to-object-to-human, to object-to-human, as seen in Figure 3.8.

This thesis draws upon the above discussions to make a case that the process of an actor like a service designer reconfiguring an object to meet the goals of personalization is leading to the adoption of increasingly social technologies able to make interpretations and reconfigurations directly with users. This reconfiguration is leading to personalization not only with shared autonomy in mediating, distorting or driving personalized experiences, but is often leading to entirely autonomous personalization (Marino et al., 2020). This evolution is summarized in Figure 3.8. Assumptions about user preferences in traditional ICT-mediated personalization can be reinforced through top-down managerial logic and conditional code that reflected that logic. Adopting machine learning to learn about users changes this dynamic. Today, choice can be offered to users autonomously by configuring ICTs with modern approaches like machine learning, where the role of the manager disappears into the background.

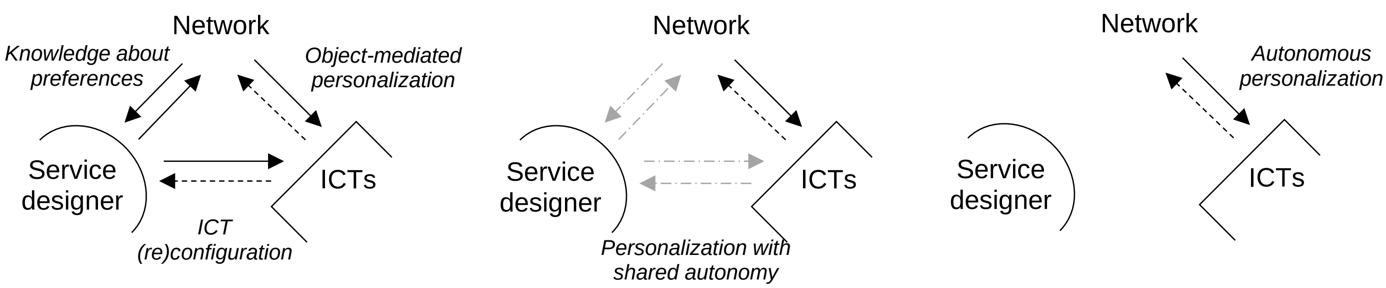

Figure 3.8. Autonomous personalization reconfiguration cycle

3.3.1 When autonomous personalization is risky

Unchecked autonomy: As autonomous systems are adopted, we are handing over inferential autonomy to some algorithms. For example, several classes of algorithms are employed to make guesses about users. Random forest regression analysis can be used to identify key features that may predict certain behavioural outcomes (Institute for Citizen-Centred Service, 2018). This is an inferential statistical analysis that previously required human agency. Today, this analysis can be activated automatically by a series of triggers. As far as reconfiguration cycles go, these inferential algorithms can go unchecked and can quietly influence important processes. When humans conduct inferential analysis, there are many checks, practices and norms around ensuring the results are valid, reliable, and ethical. Are the same checks in place for autonomous inferences?

Inexplainability: Another important dimension emerging from these new actors alongside autonomy is that of explainability, or rather, inexplainability. Earlier in this chapter we revealed that one advantage of the study of machines versus humans is we have a good foundation from which to understand their underlying mathematical and conditional logic. Even the most powerful and advanced systems still simplify information and run data objects through closed and tightly coupled logic. The data they run are still mathematical, even text must be converted into numbers in order for computation to be possible. But what emerges from autonomous personalization is a technology agent that is not entirely explainable.

To rely on complex social data means an advanced recommendation engine in a highly integrated social media platform may never be able to produce the same results twice for example, even for the same user. As time progresses, simplified objects representing data are also ever-changing. So even if technologies have seemingly deterministic properties and logics underneath, which we can understand, we are discovering that when we open up the black box of these tools there is still a black box that comes from our inability to predict the complex social data feeding into these systems. This reflects a debate that emerged in the field of computer science as well. For much of the history of the field algorithms were investigated as sole and discrete objects with an inherent computational logic. Interactive computation emerged as a paradigm challenging technology-centric views of computer science (Goldin et al., 1998). Researchers in this space call for a full appreciation for users, and the unpredictability that comes with their social data. This has also not been lost on some of the leaders in machine learning innovation. There is a growing call for user-centric machine learning design, to bring users into the center of algorithmic training (Amershi et al., 2014; Huynh et al., 2021; Zakaria et al., 2016)

Uninterpretable: Not only is an appreciation for the user and their unpredictable social data important for unlocking the full potential of machine learning, algorithms are also increasingly being used that inherently have no ability to present hints about 'why' they made the choices they did (Gunning and Aha, 2019).

Thus, even though technologies have an underlying logic that is mathematical, efforts to personalize experiences using these technologies are leading to actors that are social and inscrutable. These social and inscrutable ICT-actors are playing increasingly important roles in our social webs.

Amplified bias: One relatively well explored source of bias is the way machine learning can reinforce social bias as it already exists in social data (Williams et al., 2018). But there is another important bias that has not been as well explored in the literature. Returning to a formal definition of machine learning as an algorithm that can differentiate experiences E in solving tasks T as measured by performance P: one issue is these are heavily shaped by decisions made by designers. Incorrect or at least subjectively biased interpretations about how to codify performance or experiences can be amplified here. This is made worse when many competing and complex algorithms come together in a web of interactions, or when the design decisions are made by a network of different developers with competing visions of what an algorithm can or cannot do.

This can lead to interesting questions. Actor-Network Theory researchers call for a distinction to be drawn between actants that simply maintain connections from actants that modify connections (Latour, 2005). Is a recommendation engine more agentic than traditional ICTs that simply mediate relations, if they are now the actors triggering a change in relationship data with end-users? What about the sheer number of engines that may be intermediating our service experiences? If we are relying on more and more machines to make decisions about personalization to the point that human service delivery agents are often removed from the relationship entirely, what can be said about their importance as actors? As we've reviewed, ICTs standardize and distort reality, and are foundational to our perceptions of them because these have inescapable logic. We are inviting these actors to mediate personalization, and these are increasingly adopting dynamic social data, inexplainable processes, and learning. They are becoming more than just passive transmitters of repeatable and standardized information. They are increasingly actants of significance in our ecosystems of relationships.

This chapter began with a reference to an important debate taking place within the information systems, organization and management literatures. Research focused on users as the solely important actor, delegating technologies to the background, has held back a meaningful appreciation for the influences of seemingly deterministic properties of technology. Simultaneously, research focused on technology as a deterministic force with little regard for the importance of actor interpretation means not being able to answer important questions about why technologies are actually adopted, or in explaining the iterative development of systems with users over time, including the development of subjective dimensions like trust. Sociomateriality emerged as a paradigm that sought to bridge the divide between social dimensions and material dimensions. A meaningful investigation into deeply embedded processes requires a sensitivity to both (Leonardi, 2017). Researchers call for this sensitivity to inform epistemological and ontological research design decisions. For example, one cannot study technologies in a vacuum. This research continues this growing theoretical tradition. Applying sociomaterial lenses allows for a deeper understanding of the interactions between algorithms, designers and users. Drawing from Actor-Network Theory, we make the argument that the use of AI-mediated personalization is creating a new type of actor, one whose underlying logic calls for researchers and service designers to be conscientious. Lenses like affordances and constraints help researchers map designer interpretations across algorithmic co-production and human-machine reconfiguration. This allows researchers to bridge the gap in the personalization literature and answer richer questions about how AI-mediated approaches compare to more traditional technology-mediated personalization.

3.4 Discussion: The conscientious design of personalization

In the preceding paragraphs risks associated with autonomous personalization were explored, such as explainability, interpretability, and bias amplification from the algorithms but also their designers. Thus, decisions made by designers when adopting these new technologies matter and have consequences. This last section extends our discussion towards broader implications. If new actors are emerging in organizations through the adoption of AI-mediated personalization, there is an important call for the designers or managers of these systems themselves to be mindful and conscientious.

Enabling individuated experiences is a central component of personalization, as we discussed in the previous chapter. The pursuit of individuation and public value is redefining the role of managers (O'Flynn, 2007). This individuation needs to be sought out. The relationship between managers and recipients of services is not an agent-neutral relationship (Bozeman and Sarewitz, 2011). There is a call for active agency. Administrators need to help create and guide networks of deliberation and delivery to help maintain and enhance overall effectiveness, capacity and accountability (Bryson et al., 2014). In essence, public interventions and value creation needs to be defined by an explicit search for adding additional value (Stoker, 2006; Gains & Stoker, 2009). To realize relationship value, Zuboff (2002) called for a dispersed network or federation of providers that can act as advocates for customer or citizen needs. *Deep support*, she argues, enables psychological self-determination, or individuation. Organizations should aim to liberate the reserves of relationship value in individual space waiting to be turned into value through advocacy and relationships of deep support. The design of AI-mediated personalization can play a role in this deep support. As AI-mediated personalization actors in our networks become increasingly autonomous, they enable self-support directly with individuals, and can make fine-tuned recommendations built entirely from a recipient's social data. They can build interdependence, intimate knowledge, and repeated and robust relationships, which further support deep individuation (Zuboff and Maxmin, 2002). AI-mediated personalization emerges as a means for mediating deeply individuated experiences.

However, given what we have learned about the distorting nature of ICTs, to be supportive of meaningful personalized *deep support* that is mediated by AI, designers need to be aware of their risks, especially when giving more decision-making agency in the personalization process over to machines. Consider the follow example from Chapters 5 and 6. An executive office of a government charged with improving the capabilities of the public sector and its leadership engaged with a company to develop an interface and algorithm to allow for the searching and ranking of employees. The tool aimed to allow the office to match employees to new vacancies for example or connect employees to potential mentors. Personalization was defined as a key goal for this AI, with the hopes that it would allow HR managers to find talented employees based on their own unique search requirements and objectives. In this social setting, defining employees as 'talented' is complex. These are people with unique skill sets, experiences, and dispositions that are attractive to different organizations for different reasons. However, in order for this employee talent management system to effectively enable searching and ranking, they needed to codify employees into formats that could be manipulated and interpreted by the systems. To learn from the HR managers means to convert their interactions into these codified formats as well.

Throughout this research, examples of bias amplification and risk emerge as a product of designer decisions and their interactions with algorithms and their outputs. One straightforward but illuminating example of this comes from the decisions made around how a prospective employee are to be digitized in the first place. Objects reflecting employees were built directly from public servants from across this government who were encouraged to log into a platform and fill out fields that define themselves. They were asked basic questions like their name, work experience, education, volunteer work, and more. They were encouraged to upload a video to introduce themselves, and they were required in many cases to take a series of personality tests. All of this combined into a single data object that defined the individual as a searchable employee. A number of challenges emerged in the process of codifying or simplifying users into these strict formats. For one, it was imperative that users spend the time to carefully and accurately upload their information to the platform. In early versions the platform was clunky and a bad user experience. This led to limited adoption, and therefore early designs of the algorithm were limited by a small number of users. Over time, it also became challenging to keep employees logging in and updating their information. Gamification strategies were deployed to assist this, as well as the creation of specific features to support the users themselves, rather than just the HR managers. That is, if the platform remained difficult to use, the number of individuals being codified into digital artefacts would remain limited. Other issues emerged around the forcing or strong encouragement of certain fields to be filled in against others. For example, names became complicated because western first and last names fixed into the interface forms did not comply with much longer traditional names from the area. This meant users who wanted their full names to be represented were in effect left out. The video feeds also caused concern because many potential employees are not used to creating their own videos and are not comfortable uploading them. Especially individuals who have personal issues with exposing their face publicly. Similarly, the algorithms received far greater testing in English than the native language, which meant the tool was far more likely to retrieve a successful match for English speaking employees. This introduced ethical and bias issues that the office remained acutely sensitive to. These issues amplified as the developers incorporated more advanced algorithms to solve the searching and ranking problems.

The decision to add a feature or not to add a feature can significantly shape the material capabilities and impacts of developed technologies. These decisions can shape not only specific user behaviours but can cascade across an organization. As discussed, information technology necessarily simplifies information into computer-interpretable data, and this necessarily involves stripping individuals of their full context into formats like JavaScript Object Notation (JSON) or database entries. Therefore, it is important to understand the process of codification. There are consequences with how individuals and organizations define, preserve, and enact classifications (Bowker and Star, 2002), and this is especially relevant in the context of vastly expanding electronically mediated social experiences (Alaimo and Kallinikos, 2017). The above review of the codification of employees was used as an example of this consequentiality, but the literature is largely limited in how deep into the algorithmic design process this codification can be understood. As an additional contribution, this thesis sheds some light on the iterative design of emergent technologies that are utilizing adaptable User Interface (UI), predictive algorithms, machine learning, and natural language processing. Important risks emerge, like privacy, explainability, and bias amplification. This research is able to reveal an important process taking place between algorithms and their designers, and is able to do so from a code-first perspective that recognizes the seemingly deterministic properties of the algorithms, as well as the designer logics and the biases they bring and amplify throughout the process.

Going beyond codification decisions, in Chapters 5, 6, and 7 risk is introduced at several points in the development of more complicated algorithms. Many of these seem simple and innocuous, like the selection of a small function to make a range of numbers more interpretable, or the selection of a default parameter for an algorithm. From a user-centric lens, these can often only be detected if bias from these decisions causes alarming results at the output phase. Otherwise, their internal influence can go undetected. On the other hand, a code-first look at the experience may see it simply as an algorithmic default, and may not question why it was selected by designers compared to other parameters that could have been chosen. Each of these decisions nudge and shape the algorithm's behaviour, and are ultimately decisions made by human designers. Lacking conscientiousness can mean designers not being sensitive to these decisions, but it can also mean an organization has no way to go back and retrace decisions that were made. This explains the increasingly calls for algorithmic transparency. Autonomous actors are being introduced to our systems, and sometimes they will be inexplainable and uninterpretable, but even in those cases we need to be mindful of the design process and the reconfiguration cycle. What decisions were made and why?

In this research, we see evidence of mindfulness or acknowledgement of a number of risks associated with these decisions. But acknowledgement is not enough, the research observes how ongoing practices can emerge to reinforce mindfulness and enable actions for improvement. At the same time, this research uncovers significant gaps in this mindfulness.

3.4.1 Systemic dangers: 'Getting hooked on personal data'

Risks associated with explainability, interpretability, and amplified bias have been explored. These are the main focus of this thesis, given the existing gaps in the literature. However, it is worth underscoring that the well-explored risk of privacy also emerged consistently in this work. The nature of the algorithms themselves, their need for as many experiences as possible, create a tension with privacy. Engineers were consistently seeking out learning experiences from users, training data, and validation of their decisions made during the design of the systems. This desire for data has been characterized as a risk in its own right. As machine learning applications expand, the hunger for more data in order to support algorithms in making more accurate guesses about individuals will grow. AI-mediated personalization is a lucrative business (Zuboff, 2019). Organizations are able to make deeper and faster guesses than ever before as well. Behavioural data has become an important classification of data because the appetite for understanding users is more lucrative than ever. Larry Page described the vision of the growing Google and later Alphabet, Inc. empire, evoking excitement when he said their services will be "almost automagical because we understand what you want and can deliver it instantly" (Perez, 2011). The desire for certainty about users is not new, but revived emphasis on it is understandable in the context of incomparably vast, unfettered, and unrestricted access to the majority of an internet user's browsing, emails, pictures, locations, purchasing patterns, appointments, and more. In what has been meticulously described as a very human discovery, Google's co-founders stumbled accidentally upon the vast potential of behavioural data, under figurative duress from investor pressures to find profits (Zuboff, 2019). Advertisers are willing to pay for rich and indexed information about individuals. Industry titans began restructuring their operations around these practices, and the growing appetite to *know* everything they can about their users became an *economic imperative*. Their business models depend on a constant pursuit of predictability about user needs. Leading management consulting firms have joined the chorus, advising their clients towards "the transformation of business models from 'guaranteed levels of performance' to 'guaranteed outcomes'" (Pettey, 2016) by using predictive algorithms. Whether they succeed in achieving these guaranteed outcomes or not, it signals an effort by businesses to seek as much data as possible so they can control the experience and its outcomes. Government and business leaders alike are increasingly experimenting with Thaler and Sunstein's nudging (2008), using predictive power and machine learning to direct and alter user behaviour. That is, we are using specific formulations of technology not to accommodate user decisions, but to redirect them. Vast behavioural data makes it possible to predict and manipulate behaviour through targeted messaging, cues, and corralling interface design. Facebook was under fire (Hallinan et al., 2020) for their experiment on 'massive-scale emotional contagions' generated by their manipulations of social feeds (Kramer et al., 2014). How do we define what manipulation is reasonable? This question is not new. "The danger that a computer poses is to human autonomy. The more that is known about a person, the easier it is to control him. Insuring the liberty that nourishes democracy requires a structuring of societal use of information and even permitting some concealment of information" (Schwartz, 1989). These warnings appear more relevant today. If we are opening the doors to using behavioural prediction to direct citizen or customer decision-making, how will we be safeguarded from abuse? Will we need constraints on the way the businesses and the public sector can directly shape behaviour through new tools, much like how Traditional Public Management paradigms constrained the discretion of its pre-digital public sector (Taylor, 2014)? How will the businesses and governments balance the rising expectations for personalized services with the need to protect fairness, privacy, and transparency?

Titans have proven themselves willing and deliberative in their collection of behavioural data in pursuit of knowing, predicting, and nudging on behalf of advertisers, even if it means pushing the boundaries of privacy laws. As Google's Eric Schmidt stated, "old institutions like the law and so on aren't keeping up with the rate of change that we've caused through technology" (Gobry, 2011), at the same time the company was subject to litigation, fines, and regulations over Street View's breech of privacy laws across over a dozen countries (Rakower, 2011). Google also faced a loss in 2020 related to their cousin project: the cancellation of Sidewalk Labs in Toronto, Canada. This was a planned city built in pursuit of *ubiquitous computing*, a desire for digitizing every possible action in reality using voice, cameras, sensors, biometrics in order to build predictive models. Their extensive Toronto waterfront project attracted ire from city planners (Oved, 2019), privacy experts worried about invasion technology (Valverde and Flynn, 2018), and the Government of Canada's parliamentary ethics committee over concerns about access and transparency (42nd Parliament of Canada, 2019).

All of this underscores the risks that these emergent technologies present. Chapter 5 includes reference to how the public sector was coping with some of the key risks associated with the adoption of an employee management tool, such as through reconfiguring it. Chapter 6 takes a close look at issues of explainability, mapping a difference between when a designer knows how an algorithm works versus when they can only guess. Chapter 7 followed engineers in the development of 34 different personalization features from the point of conception to deployment. A number of negotiation strategies were observed when problems emerged. These negotiation strategies were observed across a single company, but this opens the door to future research about how societies will cope with a new actor in our workforce. Emerging tools are giving researchers and service designers a greater ability to know our customers than ever before. This is fueling an explosion in personalization. But developers and service designers ought to tread carefully. Value-inducing personalization does not happen autonomously, nor 'automagically'. It is built by developers representing managers of service design and delivery.

Chapter 4: Measurement and methodology

This thesis has been shaped by interweaving theoretical and empirical research narratives. It is important to underscore that the methodologies selected needed to be appropriate for analyzing users, and in this case the designers specifically. They also needed to be sensitive to the seemingly deterministic properties of the algorithms and programming code. As discussed, sociomateriality offered an epistemological bridge between user-centric and technology-centric investigations and provided a foundation from which decisions about methodology were made.

4.1. Methodology summary

4.1.1 Research aim

At a fundamental level, the *basic research aim* is to develop knowledge, theory and predictions around the relationships between personalization and emerging technologies, especially autonomous personalization. This began inductively with an initial notion that such a relationship *could* exist. After formulating a clearer understanding of the relationships, the research shifted to early deductive testing of a mid-range theory: that personalization is playing an explicit role in the emergence of a new actor of significance in organizational service experiences. As a minor *applied research aim*, this thesis also hopes to expand interactive mapping techniques to guide research around negotiations that take place between algorithms and designers.

4.1.2 Case selection

There were several options available for case selection at different phases of the research. With an initial focus on theory-building, a single software project from an AI development company was selected purposefully. This was not simply a choice of convenience, because there were several options available as the author had been in contact with public sector innovation offices in both Canada and the UAE. The AI company was selected purposefully on the basis of maximizing information utility (Flyvbjerg, 2006) because it offered more access than the other cases, and the company appeared to be moving faster with their innovation portfolio. A case study becomes appropriate for investigating empirical reality through the lens of a particular phenomenon when we are able to leverage multiple sources of data (Yin, 2003). The AI company afforded full access to the codebase, support systems, dialogue, developers and even clients. This proved to be a valuable combination for triangulating findings (Jick, 1979). This in-depth review into the AI company as a single case allows for the discovery of possible causal mechanisms that could otherwise go unobserved (Gerring, 2004). The long length of the observation period allows for an analysis of diachronic variation in this single unit, or, changes over time. This includes observed changes over time in both algorithms as they evolved and developers as they negotiated with them.

4.1.3 Research data

Throughout three papers, primary data was collected directly by the author. These data were collected via participant observation and were guided by three methods. The first is a code-first look at the algorithms, by tracing their code from the moment a user triggers an initial event, through their service experiences, until the process is complete. The second method for data collection involved turning to secondary resources like designer discussion boards or project management software. When this could not reveal explanations behind features or their influences on the process, a third method involved turning to key stakeholders with interviews, and particularly the algorithm designers. This is a descriptive process and can be characterized as mixed-methods with one focus on the retracing of underlying tightly-linked logic guided by a programming language, and another focus on designer interpretations of these and how they perform.

4.1.4 Data collection

The research involved several discrete periods, as summarized in Figure 4.1. Observation began to pick up a steady pace in the beginning of 2018 and continued through multiple projects until early 2020. There was little to no formal synthesizing of the observations until the middle of 2019 when a few identified prepositions emerged that seemed worthy of being tested. This shaped the development of Chapter 5. Insights and discussions from this led to another period of observation and another set of theoretical findings in early 2020 which shaped the development of Chapter 6. In both of these chapters there was a deep dive into a particular set of algorithms used in a specific project, the HR employee search tool. Finally, various dimensions were brought together into a framework that could be applied across an entire corpus of algorithms and projects beyond the HR tool, which shaped the development of Chapter 7.

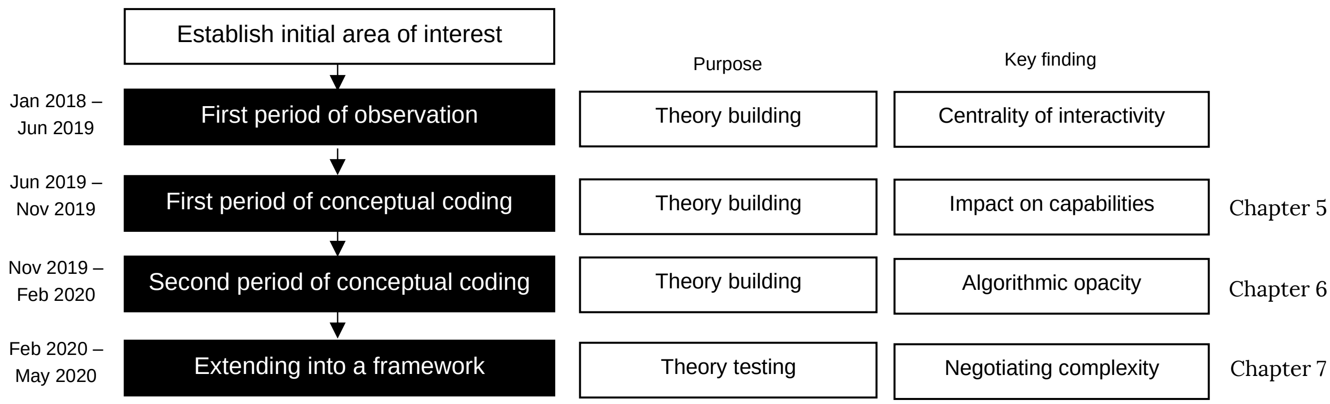

Figure 4.1. Overview of research purpose and findings over time

As summarized in Figure 4.2, throughout this research several projects are referenced. This research crosses 4 specific projects. Chapters 5 and 6 heavily reference an employee talent search project with the UAE government. Along the way the AI Company also engaged in three other projects, including a central bank algorithm that detects document sensitivity, a tool to support underwriters in tracking risk for a mid-sized insurance company, and the development of an interactive research analytics tool. Chapter 7 began when these projects were wrapping up and offered an opportunity to test 34 individual types of personalization across all 4 projects.

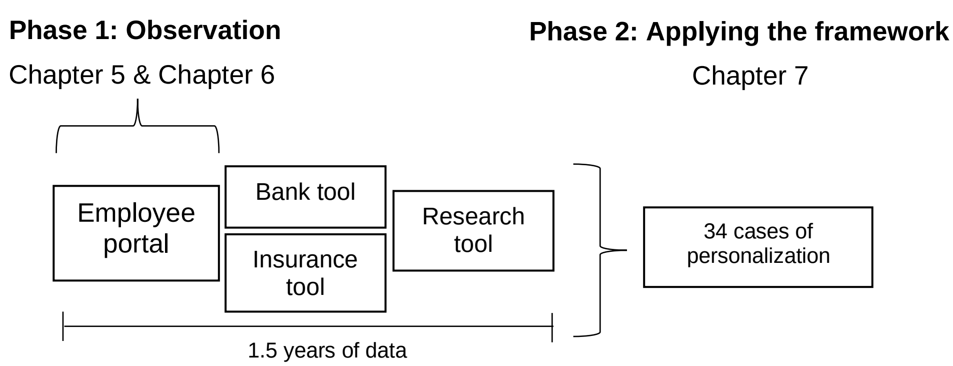

Figure 4.2. Overview of research cases

The first phase of the research looked closely at a sub-unit of analysis across the single case (employee portal), and can be considered an embedded case-study (Yin, 2003), while the second half of the research looked across the case over its many projects and algorithms which can be considered a holistic case study. Thus, the case enables taking advantage of the methodological strengths of both embedded and holistic case study research. The methodological decisions made for each investigation are covered in greater detail in their respective chapters.

4.2 The case in more detail

The AI Company was founded in 2016 as a group of technical experiments designed to showcase automation possibilities. By 2020 the company had grown from two fulltime engineers to five, as well as added operational support, corporate advisership, and institutional investment. The company incorporated in Canada and the United Kingdom and had formal activities across the Middle East with Abu Dhabi as a regional hub. The team was primarily remote, with the engineers communicating through support systems. Developers and clients met periodically in person to facilitate engagement and feedback, but most engagement with clients occurred digitally. The AI Company focuses on government and market research clients who are engaged in existing activities around measuring client needs. From within the Company there were several key actors listed below that played a formative role in the defining of client needs and in the design of systems. A few less important actors involved in projects beyond the scope of the analysis are left out. These names will be left anonymous because some in the team confirmed they would be more comfortable sharing deeper details about their entire journey as engineers if they did not have their names associated directly with it, citing for example a humble admission of being early adopters of some of these tools and not entirely sure how all of the features work. Engineer R was the data scientist who helped build the first experiments in automation. This engineer played the role of the original conveyor of the capacity of technology, or what we thought these new tools could afford. Engineer HH supported them in their early mathematical modeling. Engineer A designed and facilitated early client testing and user experience mapping. Engineer P joined the team, specializing in Python and the development of APIs. This engineer would go on to design the majority of Company's algorithms and AI. Engineer S directed overall platform compatibility and managed security requirements. This helped ensure key company values around data privacy were respected. Engineer N joined and became a lead designer of the system's frontend as a specialist with VueJS (JavaScript) and driven by a vision of promoting accessibility in technology design. Engineer K drove the original interface architecting for the research analytics project. Design and development activities were supported by Engineer HS, as well as several other contracted and operational actors. Engineer C was the author of this dissertation. Having lived and breathed market research, the author supported the projects through consultation by providing insight into market needs. The author helped the technologists configure their designs to the needs of the industry. The author was not a primary engineer in that the bulk of their code contributions were limited to early experiments. In the early stages, the author supported Engineer R, H, and P with code writing and code reviews. Over time the author's role evolved into higher-level technology architecture, acting as a mediator between user needs and engineer prioritization. For the bulk of the deployed code, the author did not experience or interact with their design or development until after their implementation into the production environments. This allowed the author to play more of a role of an observer-as-participant (Gold, 1958) than an active agent of code production.

As summarized in Table 1, the team therefore had a range of professional experiences, from 1 year to 10 years of programming experience for example, and across data science, interface design, user experience, platform architecture expertise and more.

Table 1. The Company's team: expertise and role

Engineer	**Expertise**	**Role**
Engineer R	2 years of AI programming, data science expert	Overall architecture and execution
Engineer HH	4 years of AI programming, data science expert	Overall architecture and execution
Engineer A	5 years of project management, and 2 years of user experience	User experience measurement
Engineer P	1 year of AI programming, data science expert	Data science execution
Engineer S	10 years of programming and 3 years of project management, platform design expert	Platform architecture
Engineer N	1 year of adaptable user interface programming, lead of accessibility	User interface execution
Engineer K	3 years of project management and 2 years of user experience	User interface execution and user experience measurement
Engineer C	10 years of programming, and 8 years of project management	Overall architecture

Several clients were engaged with extensively by the Company over the course of the observation period, these included three national level government departments, four municipalities, three market research companies, and two universities.

Across the observation period the technology stack remained consistent. A central enabler of the Company is their internal AI development platform. As depicted in Figure 4.3, from the perspective of the platform there are three primary activities that characterize development. First, when a new project is being developed, developers can 'fork' a set of frontend and backend boilerplates that the Company maintains, and then 'clone' it using a syntax called Git. These are default folder structures that allow developers to install a ready-to-go environment where both the frontend which gets rendered by a browser and the backend that gets hosted by an API server are hooked up. The second key action is developers will then download and update internally-developed core packages using the Git command ‘pull’. These are functional feature-sets that are designed to be easy to configure with the boilerplates. By downloading these, developers can have ready-to-go authentication features, user sharing and management, user project management, and data science functionality. These are expanded periodically, allowing multiple applications to stay compatible with feature updates. The third action is for developers to take this now fully-functional environment and begin expanding on them. They will add new folders and code that will modify this original boilerplate into their own product visions. After every coding session, users can submit their code back to a repository that will track their versions using the Git command ‘push’. When code looks complete, individual snippets are sent to different code reviewers to find errors and ensure team awareness of overall design logic.

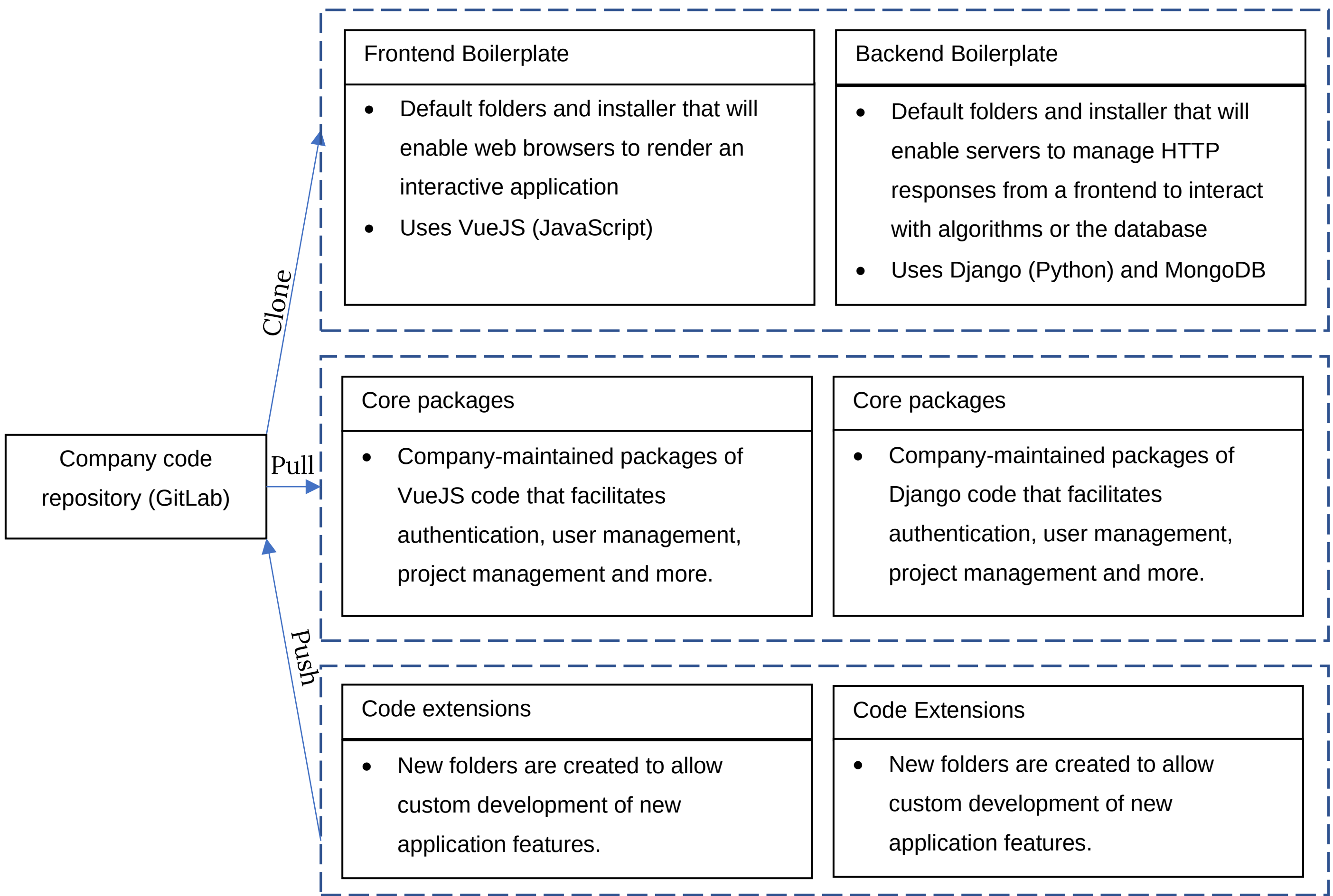

Figure 4.3. The Company's development environment

In addition to the code itself, this embedded environment gave the researcher access to the data necessary to triangulate the reasons for specific code artefacts and to measure the degree to which the programmers understood the algorithms in terms of explainability. For example, the Gitlab tool not only facilitated the cloning, pulling, and pushing of code, but it allows for entire branches of code to be split off from one another and remerge at later times. Engineers use this to enable experimentation or to separate short-term from long-term coding exercises. This also allowed the researcher to trace when individual lines of code were injected. This tool also tracked bugs, features, and requested code changes from users as interpreted and articulated by engineers. These issues are prioritized by high, medium and low priority, and are eventually labelled with the version release they are scheduled to be added to. Following these issues uncovers key interpretations, negotiations, and contestations by engineers.

Another invaluable tool is the Company's project management software called Jira which follows Agile methodology by facilitating the design of project scrum boards which allow engineers to identify specific tasks they intend to complete within a two week stretches called sprints. Access to these boards was also extended to clients so they could stay on top of project development. This transparent way to prioritize development alongside or sometimes with active client engagement was a good example of co-production. For researchers this software is invaluable because these individual tickets provide rich information about what tasks were tackled, when and why.

A final key source of insight beyond the code itself was the company's chat tool. Built on top of a software called Rocket.Chat, a strong majority of unstructured company discussions took place in rooms dedicated to various aspects of development. As a researcher the author made regular use of the generic rooms dedicated to machine learning, software engineering, as well as the project-specific rooms dedicated to client management, discussion about how to interpret client needs, and how to tackle technological complexity.

In addition to these systems, the author participated in the majority of key client meetings, sprint reviews, and team technical meetings. Along the way, the author arranged periodic informal interviews to begin scoping out and identifying the boundaries around interesting dimensions and patterns that were emerging in the observations.

4.3 Participatory observation - From grounded beginnings

The case offered a valuable opportunity to reveal the relationship between personalization and new algorithms by carefully following the development journey in a way that is sensitive to both technical and human dimensions. The author of this thesis also brought a few key advantages that could support this investigation. The author is not only a social researcher but has applied skill in each of the background systems utilized by the company, including Python, JavaScript, VueJs, MongoDB, and deployment technologies.

Case studies have been called strategies for empirical research involving a contemporary phenomenon with its real-life context (Yin, 2003). Because the research began with an area of interest but no formal models or hypotheses, the first half of the research drew from ideas and concepts in grounded theory methodology (Glaser and Strauss, 1973; Strauss and Corbin, 1998). Grounded theory goes beyond passive observation and thick description (Strauss and Corbin, 1998), as it can allow for the organic identification of meaningful patterns and enablers across social phenomenon. This research draws from the spirit of naturalism in that the closer we get to the conditions we are studying, and its people, actors, systems, and institutions, the more meaningful our descriptions might become (Becker, 1996). But for grounded theory research methods to yield meaningful patterns, researchers need to be careful not to drive observations too rigidly or with bias in focus that may lead to researchers creating self-fulfilling prophecies. Grounded theory research methods depend on as unbiased a set of observations as possible. However, pure grounded theory research approaches with zero previous awareness of the case or context is rare and may be impractical. The bulk of case study work is done by people who have intrinsic interests in the cases (Stake, 1995). The first half of this research therefore was a softer form of grounded theory, not entirely unlike the inspired works of Eisenhardt (1989) which further refined this tradition. Eisenhardt called for some *a priori* constructs and specifications, with research beginning with the establishment of general research questions that inescapably draw from our experiences. Over the course of this research however thick descriptions should begin to be clarified around conceptual density and meaning (Strauss & Corbin, 1998).

4.4 Mitigating bias amplification

The nature of this particular research amplifies a set of risks. The choice to adopt participatory observation is not without its trade-offs. In particular, when defining the relationship between human actors and computational actors in an Actor-Network-Theory we learned that nodes (humans or computers) can distort the process through subjective bias or strict and structured properties of the technologies. As outlined in the initial definitions in Chapter 1, bias here refers specifically to bias that creates discrimination for subgroups of people, be the bias from existing social data, the process of computing, or the practical use of tools, as well as from service designers themselves. Designer decisions around what technologies do or cannot do shape their functionality. For example, if designers have an incorrect understanding of an experience, a machine learning algorithm built around learning from those experiences could amplify bias. This research aims to measure the choices behind the adoption of these algorithms. To study these choices means to make inferences about the interaction between humans and technologies. This means the author themselves, being embedded, may also be amplifying biases, interpretations, and understanding. To allow for unbiased thick descriptions to emerge from the observations the author aimed to distance themself from formal discussions about the understanding of technological capabilities and the design of algorithms. The author aimed for a careful balance between an 'insider' perspective and an 'outsider' perspective (Becker, 1996). As such, the author played the role as an observer-as-participant (Gold, 1958). As part of maintaining reflexivity about the author's role in influencing the research, they were careful to separate work from observation activities, and ensured questions about the underlying observations were only conducted out of normal office time. The author would also write observation notes in the evening or outside of formal office interaction with the team. This distance from the actual moment of decision-making allowed the researcher to approach the question with a fresh mind, and to interrogate the designers for their decisions rather than having been embedded in the decision itself.

Moving from observation to coding involved a series of decisions as well. Pattern-matching (Yin, 2003) and congruence methods (George and Bennet, 2005) were used to support thematic coding that was guided by Miles and Huberman (1994). Over time, codes were reviewed and compiled into nodes (Mason, 2002) to combine or remove overlapping themes. The goal being to maintain internal consistency and external divergence (Marshall and Rossman, 2014). This was especially helpful during the first phase where new ideas were emerging naturally from the data. The process to codify observations into these themes was iterative. During the work around Chapter 5 and 6, these codes were then reviewed by external investigators who were the co-authors of these respective papers. This allowed for another introspection on theme consistency, and helped minimize bias amplification because observations and their subsequent coding were further scrutinized from outside perspectives.

4.5 Reconciling the epistemological divide

Sociomaterial research has been pushing the boundaries of understanding organizational practice and technology because it hones the analytical gaze towards moments of interaction between technology and user, rather than treating users or technology as background noise. This tradition has also played an important role in bridging seemingly antithetical notions of truth. On the one hand, users necessarily create understanding of the world through their own interactions with it, based on subjective experiences and interactive meaning-making (Hacking, 1999; Sismondo, 1993). This implies user-sensitive research methodologies need to appreciate that reality is understood through the social constructions of it employed by actors who are engaging in that meaning-making. That is also to say, in this view reality is contingent on this observation and this view rejects deterministic paradigms such as the idea that humans are rational and this rationality is independent from individual surroundings. On the other hand, digital technologies emerge with distinct properties that set them apart from social actors. They necessarily simplify information and are closed from social construction and interpretation in the moment they process logic through tightly-coupled links. This implies adopting a research methodology that is sensitive to seemingly deterministic properties and logic. By formal definition, an algorithm will run the same if given the same data.

Where these paradigms intersect is the moment of interaction between users who engage in meaning-making around the technology, and the technology's physical properties which necessarily distort and shape the information and outputs in a deterministic logic. As a result, even though algorithms are deterministic, the data they increasingly make use of are increasingly dynamic, socially-contingent, and in the case of personalization algorithms, occur increasingly in direct interaction with end-users with the designers of said systems taking on more distant roles. Deterministic properties and subjective meaning-making come together in live moments, both in the experience of the service and in moments of design. By focusing on a code-first look at the technologies that enable personalized experiences, analysis involves inferences over formal objects and logic. But explaining why these specific formal logics were adopted cannot be done formally. A look at interactivity and through the triangulation of different sources of data. especially by measuring the perspectives and assumptions of the designers themselves, became imperative.

Many exemplary sociomaterial research methods have inspired this thesis, such as Leonardi's (2009) measurement of the alignment and misalignment of material technology features and social interactions. Use and interaction was a central moment that shaped this work. This thesis intends to continue this tradition of bringing together research epistemologies that emphasize social construction and that emphasise technological-determinism. Important lenses that are employed to support this include functional simplification and closure to reinforce sensitivity to material properties, Actor-Network Theory to situate technological and human actors as mediators of experiences with distinct properties, and affordances which help operationalize technological features as perceived by designers.

4.6 Advancing interactive mapping

A minor applied research aim of this thesis is not only to apply sociomateriality to the research questions, but also to make modest advancements in sociomaterial measurement by expanding existing interactive mapping methods. Use and interaction have been central to these methods, but thus far there has not been extensive application of these to understanding the design of complex webs of algorithms and especially emergent technologies like AI. This is often because these are proprietary technologies and their inner workings are guarded intellectual property (de Laat, 2018). Given the case of this thesis offers unfettered access to not only the code, but supporting documents and complete access to the development team, this gives a valuable opportunity to investigate 'use and interaction' not only of a completed product, but over the evolution of a product from its earliest forms to their most complete forms. This look over time, and intimate access to data, has enabled this research to expand interactive mapping.

What this thesis calls *interactive mapping* began with Li and Jagadish (2014). They tackled the challenge of mapping natural language processing queries. Like other AI technologies discussed throughout, natural language processing is generally considered difficult to model and describe. They overcame this through careful interaction with users to iteratively model the architecture until they were able to correctly interpret this complexity in a generic manner across a range of domains. This inspired the thesis because 'code-first' analysis was easier said than done. For example, a code repository can contain tens of thousands of lines of code across multiple instances and programming languages. It can be challenging to know where to begin. The author initially faced this difficulty. Not knowing where to begin, an exercise in mapping and explaining the repository proved disorganized. However, following Li and Jagadish, the author decided to 'begin with the users'. For example, the first components worth mapping became the first experience a user would encounter should they initiate the program. Countless algorithms are involved, but by following the flow of the user, the author was able to carefully map the flow of data from interaction to method to function. Where an interaction led to a 'fork in the code', one direction would be mapped until complete. Then, the author returned to the 'fork in the code' and preceded down the other path as well. This was done until the entire code base had been investigated, and every scenario of data flow was understood.

In terms of 'understanding' each function, this involved first understanding what the code actually does. This is a deterministic investigation in that the code should have a specific set of properties that lead to specific outcomes. However, within algorithms and the code surrounding them are little artefacts that may not be understood in terms of 'exactly' what they do, like a parameter that is a specific digit. For example, in Chapters 5 and 6 a parameter with the number '20' as identified. Why 20 and not 19? What impact would 19 have instead of 20? These are questions that cannot be answered simply by looking at the code. Li and Jagadish answered questions like this by designing experiments with users. This thesis took inspiration from this, but opted to take a mixed-methods and mixed-data approach involving triangulation (Jick, 1979) instead of experimentation. This approach was chosen given the research questions are around understanding the iterative development of personalization technologies and the role of their designers in this process.

When the author could not fully interpret an algorithm from the codebase, they would turn to the GitLab, Jira, and Rocket.Chat to identify the time of the code and discussions taking place in and around that time. The author was initially surprised to see how much of the design involved negotiations in interpretation of either user needs, technological capabilities, or explaining how algorithms were working. Where the author could not uncover clear negotiations or interpretations, interviews with the developers were arranged.

This mapping also enabled a longitudinal look across the evolution of an algorithm. The support tools allow researchers to go back in time and see the state of an algorithm from its earlier iterations through to its more complicated structures. This coincides with support and evidence from developers that can allow for an understanding of why decisions were made and logics were added. It also allows for a separation between different components of an algorithm, some of which move between algorithms as data objects, some as standardizing interfaces that shape how users experiences are structured, others as interactions with databases, and others as calculations that draw upon methods, modeling, and learning that is often called AI. As Li and Jagadish also show, interactive mapping allows for complex architectures to be easily visualized. This thesis takes advantage of this in Chapters 5, 6, and 7.

Contributions of interactive mapping:

- Easy starting point into a codebase
- Combines technical and social elements
- Enables longitudinal mapping
- Supports the differentiation of technical and user components
- Can be represented visually

This formal mapping was applied to a small number of key algorithms in the earlier stages of the research. For example, Chapter 5 describes the evolution of an employee management algorithm, showing how interconnected processes come together to enable talent sourcing. This developed the preposition that personalization algorithms could change organizational capacity building, which in the context of the public sector could increase overall public value. This capacity building did not happen spontaneously. Interactivity mapping exercises uncovered a complex process of negotiation between developers and the public sector manager in charge of championing the new tool and its application to their executive office's leadership agenda. In Chapter 6, this interactive mapping supported intimate questions around how the designers are grappling with their own biases and differences in interpretation, especially with algorithms that are difficult to interpret.

In Chapter 7, this mapping exercise is expanded to cover the entire corpus of the AI company's personalization algorithms and interfaces. For example, beginning with user interactions every function and feature is mapped, and if a feature qualified as enabling a personalized experience in some way to a user, it was set aside for deeper investigation. After identifying over 30 cases of personalization, the author mapped every line to ensure a full understanding of their functioning. This holistic ‘across projects’ investigation was to broaden the analyses on algorithms to see if there were stable or even generalizable themes and patterns across different personalization features. This also gave a chance to better test the differences between features that are interpreted as AI versus features that were not. This could help better define AI-mediated versus non-AI-mediated personalization. If in Chapter 5 the research observed the consequentiality of personalization and in Chapter 6 the research observed the persistence of black boxes, it is in Chapter 7 that the research attempted to explore how black boxes are managed and negotiated with. Complexity in general became a theme that guided the development of a practical measurement framework. If personalization is important, what are the strategies we can employ to mitigate complexity in design? What strategies could we use when utilizing powerful new social technologies?

Those algorithms that involved AI or machine learning often proved to be the most challenging to interpret for designers, and this led to a different kind of negotiation and contestation, one involving a lot of guess work and constant iteration with users. That is, this mapping exercise uncovered different tensions in design when involving algorithms that were hard to explain. Issues of explainability became an important dimension, and this interactive research method allowed for a deeper investigation into this phenomenon.

Chapter 5: AI and public value creation: how algorithms shape organization action

5.1 Chapter preface

This chapter is based on a paper co-written by the thesis author and Dr. Antonio Cordella. At the time of this writing, this work has been submitted to an Information Systems journal and may undergo changes to its current form before publication. Nonetheless, this work in its current form played an important role in answering the research questions set out in this thesis.

This chapter follows the development of a discrete set of algorithms and related interfaces as they supported a government HR department in their selection of new employees. This case helped reveal the relationship between personalization and the adoption of emerging technologies, namely natural language processing and machine learning. This government made personalization an explicit goal. In order to improve the experience for HR managers to better find employees, a matching algorithm was developed to help search queries retrieve suggested employees 'smartly'. As of the writing of this chapter this HR department has expanded use of the algorithms, further integrating them as APIs with their leadership and transformation platform. Previously, when tasked with identifying future leadership for the government, HR managers would rely on third-party consultants or put out internal job announcements and would have to wait for results. By turning to a platform, HR managers instead begun proactively finding employee candidates based on their personalized experience with the algorithms and search interfaces. This case offered an opportunity to interrogate how these personalized experiences are mediated by these algorithms and how they were designed.

Public value was central in this work, with an interest in understanding how these personalization algorithms changed organizational behaviour and how it met the goals of driving government improvement. In Chapter 3 personalization was linked to value-creation. This does not imply there is only one way to produce value for people. Nonetheless, the mediation of individuated experiences has been seen as one form of value that has been sought out by managers in both the private and public sector. The work in this chapter continues the dialogue of the public value literature. This is not a core aim of this thesis however, and as a result for readers unfamiliar with public value this chapter may introduce more questions than it answers.

Another important theme that was highlighted in this chapter, one that is central to this thesis, was that of designer consequentiality. That is, given the risks that the designers of services can introduce, there is a call for designers to be more mindful and to emphasize awareness around issues of explainability, transparency and bias. Tracing each algorithm, certain design decisions were uncovered that would influence or nudge the algorithms' performance in various ways. Some of these decisions changed the fundamental architecture of the algorithms, and others seemed trivial and simple but tracing 'why' these were made proved difficult. A worthy commentary should be extended beyond the chapter's findings around how effective the tools really are, and whether or not these designer decisions had negative influences on the overall experience. In practice, the HR managers appeared to accept the tool and use the tool. Certain aspects of its functioning were at first questioned by users, but then later accepted. Further research into this case or similar cases should explore how much of the use is tied to 'accepting it for what it is', a resignation in negotiation, or if they were indeed fully happy with its features. Even if its adoption is evident, this work did not rule out the presence of bias. An issue of bias that emerged is the lack of appreciation for differences between characteristics of an employee and competencies of an employee, and the assumption employee skills can be accurately inferred from their digital resumes. Neither the designers nor the HR managers appeared to be aware of these issues. As such, the algorithm made no attempt to differentiate these differences or mitigate these challenges as well. Was this the right decision? Another was the decision to give 'existing job title' the largest weight in algorithmic computation. This was chosen through iteration with users to match intuitions about how the results should feel, but this was not a robust or empirically-driven decision. Yet another was the selection of a sigmoid function with a specific curve set by a specific parameter value. Why not a different parameter value? Decisions behind these were not always clear. Were these the most optimal choices that could have been made? This chapter did not answer these questions, about whether this HR system is optimized, instead the aim was descriptive. This research did however bring these issues to the forefront.

An important question related to this designer consequentiality is that of explainability. In many cases, designers could not fully explain their decisions, the impact of certain parameters, or 'why' algorithms made some of the choices that they made. Personalization did appear to change HR manager behaviour, and was considered a successful contribution to their transformation agenda. The process of building these algorithms, however, was not a simple or guaranteed process nor a process without flaws and risks. Following this design cycle proved fruitful and became the beginning of several phases of further research.

5.2 Introduction

Many governments are turning to digital innovations to provide better public services and hence better satisfy citizen's expectations and needs. Information and communication technologies (ICTs) have proved to help public sector organizations to achieve tangible improvements in the trust (Grimsley and Meehan, 2007), timeliness of service delivery, improved staff performance, or overall satisfaction with government services (Institute for Citizen-Centred Service, 2012, 2018). However, the process by which digital innovations foster these improvements is far from being linear and without challenges. To better address the complexity that shapes the impacts of ICTs on public sector and the value they generate for those who consume public services, both academics and practitioners have begun focusing on dimensions other than performance indicators. Public value theory has offered valuable insights to account for the multidimensional factors that shape the impacts of ICT on public sector organizations. Public value theory suggests that public organizations do not generate value per se but rather enable value creation by providing services to be consumed. Consumption is the actual process by which this type of public value is created. This does not imply this is the only way value is generated, but suggests one discrete and concrete value that can be facilitated. The achievement of citizen expectations and needs is what ultimately creates this type of public value (Bentley and Wilsdon, 2003; S Osborne, 2010; Osborne and Strokosch, 2013; Vargo et al., 2008). However, citizens expectations and needs change over time. Hence a public administration oriented to public value creation should develop competency that can adapt to these changes. In recent years, artificial intelligence (AI) tools emerge as having the potential to increase public services' abilities to manage complex and ever-changing citizen needs and expectations (Andrews, 2019; Wirtz et al., 2019) and hence public value creation. That is, AI can be used by public sector organizations to better configure their organization's capabilities to address changes in value creation. For example, AI is used by public sector organizations to better source employees that have capabilities and characteristics aligned with public value creation needs and challenges. For AI to be effective in supporting the sourcing of public sector employees to meet mutable social expectations, they must also support the organization in better understanding what employees are needed in different circumstances. To be effective in these tasks, AI must be adaptable. The algorithm must learn from past organization choices about what are the services that the organization must provide to deliver public value and about the employees that are needed to deliver these services. This leads to two inspiring questions: *How do learning algorithms impact public sector organizations ability to identify the characteristics of the employees that are needed to support the creation of public value*? *How do learning algorithms impact on how public sector organizations select employees needed to support the creation of public value*? For the purpose of this chapter, these inspirational questions were synthesized into a more practical question:

RQ1: *how might a system be built to find best qualified or suited employees, and what challenges, risks and negotiation strategies emerge throughout this process?*

To answer this research question the paper follows the adoption of a learning algorithm by the Prime Minister's Office of the United Arab Emirates designed to enable the public sector to better serve social service expectations through more effective employee scouting.

5.3 Literature

5.3.1 The rise of public value

The literature on public value suggests that value creation does not only occur within public sector organizational boundaries but can also occur with the consumption of the services that are created and delivered by public sector organizations. Public value creation is moulded by specific organization capabilities (how public services are created and delivered) but also by the legitimacy and support that the action of the public administration receives from society (i.e. those who will consume the services). Public value is created when public services are consumed and their consumption satisfies social expectation about the action of the government (legitimacy and support) but also the needs of all those who consume the services (Cordella and Bonina, 2012). Public value shifts the focus of the analysis of the value creation proposition from the organization that produces the services to the individuals who legitimise and consume them. This shift suggests that individuation matters and that this type of value generation is mostly occurring outside the organization boundaries (Meynhardt, 2015; O'Flynn, 2007; Vargo and Lusch, 2004). If so, this has significant implications for service delivery in the public sector. To enable public value, organizations need to assist in the fulfilment of individual preferences. This requires public sector organizations to have the right organizational configuration and the competences and skills needed to understand what the different individual needs are and to identify the right mechanism to support their fulfilment (Moore, 1995). The proper selection of the employees needed by the organization to identify individual needs on the one hand and to design personalized services on the other are organization challenges that can be supported by digital innovation strategies. Public sector organization can rely on digitally supported HR selection processes to better acquire the employees needed to become more responsive to individual needs and expectation and hence to better create public value.

5.3.2 Public value, value creation, and employees

The need to redefine the approach to value creation in government and hence the importance of the role played by employee selection in this value creation process is the result of a paradigmatic shift in the definition of what constitute value. A strong tradition in applied market sciences focused on a unidimensional definition of value derived from neoclassic economic theory. Accordingly, individuals are rational and their needs are *known*. Services are made available for exchange, purchased by another to maximize individual utility (Sweeney et al., 1996). Because value is presumed to be known and made available for exchange, the value generation proposition in confined within the organization that produces and delivers the services. In this tradition, the value of that exchange can be defined in terms of the performance of service production and delivery processes towards outputs. This coincides with the ubiquitous rise of service quality and client satisfaction measurement that occurred during the rise of New Public Management (Osborne and Gaebler, 1992; Aberbach and Christensen, 2005). Public value, on the other hand, stresses that there is an explicit relationship between a service and the user of that service (Meynhardt, 2015). Value is not produced by organizations, where consumers are simply passive users seeking utility but rather when a given service is used and by the context within which it is used. This type of individuated value originates from the interdependence between the recipient and the network of providers. This value cannot be detached from the societal context where the service is used and through which it is defined (O'Flynn, 2007). With respect to this type of value, there is no individuated value until an offering or service is used and experienced, as it is this experience and perception that becomes essential to its determination (Vargo and Lusch, 2004). This suggests that the center of this type of value creation moves away from *exchange* of utility to *use*, and therefore process becomes important rather than just outcome (Vargo et al., 2008). This has direct implications for how public sector organizations produce and serve services and the use they make of ICTs to support these undertakings. In this context to understand and enable value creation requires unique skills and competences to identify what services are needed and how to produce and deliver these services to fulfil social expectations. ICTs can support and enable this by helping organizations acquire and manage employees with these skills and competences (Alford, 2016; Cordella and Paletti, 2019).

Moreover, public value is multi-dimensional and ever-changing (Baumgartner and Jones, 1991; Bozeman and Sarewitz, 2011; Chapman, 2003; Nevitte, 2002; O'Flynn, 2007), with different values being promoted or privileged by individuals or communities at different times. Public value is not determined once and applied indefinitely. Over time, users of the service change their expectations with regards to what is valuable to them and hence what is needed to enable the creation of the expected value. Again, this implies a need for diligent engagement between a service provider and the user. Service providers should react to changes in these subjective, multi-dimensional values to be able to fulfil public value creation over time. Public value creation deals with inherent uncertainty. The point of interaction with a user of a service becomes an opportunity to learn about the user needs and their possible mutations. This exchange can be facilitated by ICTs (Zuboff and Maxmin, 2002) and by public sector organizations that rely on the right employees to manage a public value approach to value creation (Kelly et al., 2002).

O'Flynn (2007) reinforces the importance of the role of employees in public value creation warning that public value would redefine the role of managers within the public sphere. As already discussed, the relationship between public managers and recipients of services is not an agent-neutral relationship (Bozeman and Sarewitz, 2011). This is a common theme across the public value literature which calls for recognizing that public sector managers are not supposed to remain without an active role in shaping value creation (Cordella and Paletti, 2019; Moore, 2003). Public sector managers need to have the right employees to help create and guide networks of deliberation and to help maintain and enhance overall effectiveness, capacity and accountability (Bryson et al., 2014). In essence, public interventions need to be defined by an explicit search for adding additional value (Gains and Stoker, 2009; Stoker, 2006). This calls for a profound redefinition of employees, i.e. competences and skills public organizations must have to support and enable these new ways to create value.

How exactly can employees with the needed competences and skills be found, given the multidimensionality behind public value values? This is a public value dilemma and finding the right tools to support this concerns the identification of the right employees to support the organization's value creation needs. We revisit this theme throughout this paper. Can HR solutions enabled by AI service innovation allow the public sector to find the competencies needed to create value directly with users? Evidence suggests not all AI solutions help identifying the right competences in the same way. Earlier adoption of these tools suggested they can introduce complexity and less effective employee scouting which inhibits service efficiency for example, leading to less satisfied users (Surprenant and Solomon, 1987). Modern machine learning approaches on the other hand allow systems to make better predictions about the user expectations and needs than other technologies (Amershi et al., 2014). Can these emerging technologies therefore help public sector organizations to better identify and select the employee needed to play a strong and advocative role in the enablement of subjective citizen needs?

To answer this question we need to unfold how these technologies work and to unfold the mechanisms by which they create value.

5.3.3 Functional simplification and closure

The research recognizes that technologies have a few qualities that set them apart from other social artefacts (Kallinikos et al., 2013; Luhmann, 2005). Take a computer program as an example: a specific programmed code will run the same anywhere, if using the same software and hardware. Regardless of ideology, values systems, or interpretation, technology has a certain function that is closed from social construction and interpretation, again, even if just briefly: the code functionally closes what is processed making it not possible to change the outcome. This brief moment is important, for it implies some objective quality of technology needs to be necessarily understood in order for researchers to appreciate the ways in which individuals and organizations will derive meaning from it.

The forms and functions matter. Information systems manage information, make decisions about individuals, or make predictions about unclassified information, coupling predefined logical sequences of actions through seemingly deterministic coding structures (Cordella and Tempini, 2015):

that is, they functionally simplify the complexity of the world into the code of technology. Understanding these structures therefore become important for understanding the ways value is created around it. For example, the way in which a systems determines the forms and structures of classifications can have distortionary effects (Bowker and Star, 2002). They can force individuals into top-down and limited categories. These categories may or may not effectively enable value creation, depending on the user and the context. Alternatively, they can also unlock interactive potential. Consider the case study of PatientsLikeMe, a platform that enabled individuals to identify their own symptoms from a curated knowledge-base. Their own interactions with the system allowed medical researchers to find new categories of symptoms that traditional top-down research could not (Kallinikos and Tempini, 2014), enhancing health care provision.

The algorithms behind these interactions are nebulous but also built from finite and specific codes and measurable interactions (Alaimo and Kallinikos, 2017). These are becoming artefacts worthy of study due to the significant and in some ways generalizable way in which they influence the intermediation of services between citizens and government. This has measurement implications. Functional, tangible and discrete patterns and classifications can be observed and measured across computing (Goldin et al., 2006). Interactive computing as a paradigm calls for a recognition of a fundamental difference between computational features. Algorithms inside a computer for instance are formal, logical, and seemingly deterministic, but are highly dependent on inputs. Because of this dependence on inputs, computation as a whole, beyond just the algorithms, also involves parallel processes, time independence, and non-determinism. This can be observed in the form of user interactions, the changing states of data objects, or the influence some objects may have on others based on different user contexts. This paper includes these distinctions to help identify critical nodes, tensions or interactions within the case's AI. We recognize and underscore throughout our research the complex processes by which AI algorithms and the associated personalization mechanisms shape the decisions and choices an organization has when using these systems. We also recognize that technologies have forms and structures that nonetheless inherently standardize the experience.

5.4 Case methodology

5.4.1 Research methodology

This investigation originated from broader questions around exactly how users' configuration of the algorithm (hereinafter algorithm's personalization) is negotiated between subjective users' expectations and the functional structures of the AI algorithm. The research closely follows the development of an AI application which enables leadership across the United Arab Emirates' public sector to better identify and scout employees needed to deliver public value through the use of a recommendation engine. This primary data collection occurred between August 2017 and June 2018. The case was selected in part due to our access to the AI's development from ideation to deployment, from rich and detailed development documents, to key stakeholders, to the source code. The case is also interesting given the Prime Minister's Office put personalization as one of its key requirements, offering an opportunity to investigate a deliberate attempt at building AI to enable personalization.

As depicted in Figure 5.1, drawing inspiration from interactive computing case-studies emphasizing user-centered AI (Amershi et al., 2014) we adopted an iterative research approach that starts with the algorithms and technology themselves by reviewing the source code and software architecture documentation, but we then carefully note moments where users interact with algorithms (Li and Jagadish, 2014), and moments of algorithm personalization. For example, we look for moments where users have their profiles converted into data objects, or where an algorithm makes assumptions or matches based on these data objects. We then turn to supporting sources, such as the additional development documentation, version control notes, weekly project management notes, and interviews with the developers and project owners on the side of the government (Table 2).

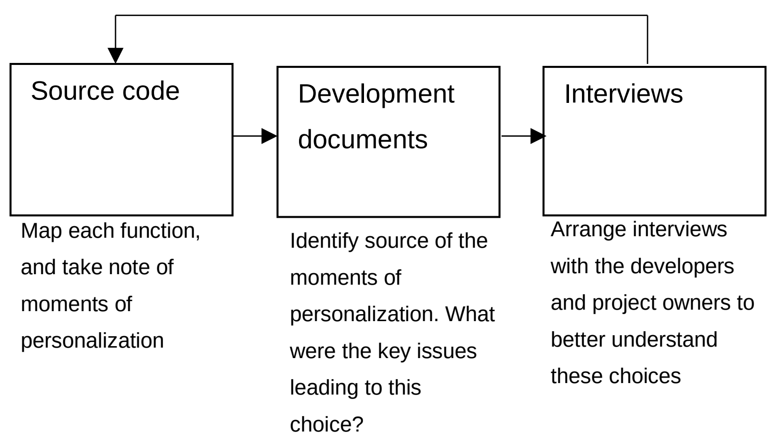

Figure 5.1. Methodological flow

Table 2. Data sources

Interviews	Development documents
Project Manager	Procurement and project initiation documents
Senior Technical Manager	Developer team chats (Rocket Chat)
Senior IT Consultant	Project reports to the client

Director of UAE Government Leaders Programme	Meeting minutes
Assistant Director General	Scrum sprint reports (Jira)
Managing Director for the legacy developer	Revision control logs (Gitlab)
Project Manager for the legacy developer	Source code
Chief Technical Officer of AI developer	Backend (Python)
UX Lead of AI developer	Frontend (VueJS)
Design Lead of AI developer	Database structures (API and SQL)

As an illustration of how this works, imagine the following flow of research:

- We review the source code and discover the following lines of code:

```
if word in tf_idf_bank.keys():
    w_w = tf_idf_bank[word]
else:
    w_w = 0.075
```

- We notice a conditional function that says if a word from a search query is in a pre-built dictionary called tf_idf_bank, load that word's tf_idf score. This is a score of rareness. But if the word is not in the list, give it a default value. What was the origin of this? And what is the default? Why was it selected?

- We consult the revision control logs to identify when these lines of code were added. We discover that they appeared after the initial launch of the algorithm, during periods of active tuning.

- We consult the sprint reports in the project management software (Jira) at the time of this line of code and discover that there was a ticket dedicated to the improvement of search results.

- We consult the developer chat (Rocket Chat) during this period of coding and discover back and forth conversations about some user feedback. We then arranged an interview with the lead data scientist. It turns out that users, when typing '*project manager with digital skills*', expect the word 'digital' to be the most important, but the search results focused on 'project' and 'manager' as they are more common. This is where we learned about the tf_idf score. A rarer word, it was determined, may be more important in a search query. The default was selected based on a suggested default from the broader community of data scientists but was nudged to find results that were optimal.

We repeat this process for each instance of algorithm personalization, interaction, or additional layers on top of the algorithm that we discover. We then thematically code our data. Codes are read interpretively rather than explicitly, to allow for deeper meaning-making. From codes we elaborated larger themes. Themes and codes are cleaned periodically (Marshall and Rossman, 2014) and iteratively (Mason, 2002). After feeling confident with the defensibility of our themes we also employed the use of narratives and stories to guide our collaboration. For example, pulling from our sources we re-told some of our themes as stories that follow specific characters like developers, project owners, and users.

This development project involved a range of actors. On the government side, the project owner is the individual in charge of managing the day-to-day development, managing the relationship with the vendors, and is responsible for the project timelines. The Prime Minister's Office also has its own IT team and security department, which manages challenges related to policies, privacy and infrastructure. With respect to this project they fill the role of technical managers. The government also has a third-party developer who produced the legacy system. Senior to the project owner is the Assistant Director General, who provided high-level vision and final approval on all features and deliverables. The last relevant direct stakeholder is the AI company, which brought with them a technology lead, a project manager, a user experience lead, a designer, two data scientists, and a front-end developer. In total, twenty-two interviews were conducted across these groups. The interviews mostly followed a similar format, beginning with asking the stakeholder to describe the project, the problem it aims to solve, and their own roles. From there, depending on their role or interest, the conversation moved into domain-specific issues. Interviews with the AI developers were less structured due to the close interaction and high engagement between the researchers and this team, and instead focused on clarifying artefacts detected in the source code.

5.4.2 Background: Original aims and the legacy system

The Prime Minister's Office of the United Arab Emirates launched the Global Leadership Programme (GLP) as part of larger e-government reform objectives under the UAE 2021 vision. The GLP manages and organizes UAE government leadership development programs, the strategic relationships between leadership and government management centres, and the initiatives needed to build specialized e-government capacities throughout government. A key initiative under this mandate is the development and improvement of an innovative digital platform to better manage recruitment, career development, and other related career activity. This would eventually take the name Qiyadat.

The GLP is guided by the key strategies set in place by the leadership in pursuit of organizational and government improvement. One of these is the Global Leadership Model which identifies ten core competencies the UAE wishes leadership to foster in their employees. Upon its launch in 2015 Shaikh Mohammad Bin Rashid Al Maktoum, Vice-President and Prime Minister of the UAE and Ruler of Dubai stated, "the Leadership in government work is not about titles or positions, but about continuous development and efficiency, as well as looking ahead and being proactive in facing challenges," and, "the UAE journey towards the future requires continuous development efforts that focus on UAE nationals, build their capabilities, enhance their skills, and equip them with the right tools to face the future challenges" (UAE Cabinet, 2016). These ten competencies (Table 3) reinforce a call for openness to ever-changing citizen realities. For example, employees should be *open-minded* to new experiences, *futuristic* in being able to anticipate and plan ahead, *agile* in promoting new and efficient ways to deliver change, *innovative* in being a catalyst and enabler of change, and *technological* in being familiar with emerging tools. These competencies also call for employees to play advocative role as an *enabler of people* to reinforce human capabilities, and to be a *role-model* in actively promoting happiness. All while adding value through achievements to promote and represent the nation positively. Given their recognition of changing citizen needs, as well as an advocative role employees can play in adapting to these changes while pursuing added value, these efforts can be seen to align leadership and employee strategies with public value generation.

Table 3. Global Leadership Model competencies[1]

Competency	**Definition**
Enabler of people	Inspires, encourages, and motivates others; reinforces human capabilities and talents through empowerment, effectively leverages others' capabilities and demonstrates emotional intelligence.
Role model	Shows values of integrity, humility and respect; embraces and promotes the concepts of happiness and positivity; makes substantial contributions in representing the country in a positive way.

[1] This government uses competencies generally. It can be argued that there are critical differences between characteristics broadly speaking and specific competencies. Some of the competencies listed may be more characteristics than actual representations of skill and competency. The designers of the tool, and the HR managers asking for it, did not differentiate between the two. It is therefore no surprise that the algorithm makes no effort to differentiate these. This is a worthy point of future research, could an algorithm be adapted to better handle these differences? What does it say that the HR office itself did not care for these differences? Bias is amplified not only by the designers by the requests of the users themselves.

Open to the world	Open-minded to different experiences; embraces the values of peace, tolerance and coexistence; enjoys an extensive network of relations and is well-versed in global culture.
Futuristic	Well-informed about global trends; able to imagine the future; anticipate and analyze opportunities through developing future scenarios and proactive plans
Innovative and disruptive	Catalyst for change at the individual and institutional level; entrepreneurial, Risk Taker and adventurous for whom nothing is impossible.
Well-versed in advanced technology	Awareness of new technologies and trends such as the fourth industrial revolution and artificial intelligence (AI) and how to get the most benefits out of these technologies which will transform the way we live and work in the future to achieve people happiness.
Life-long learner	Seeks self-development in order to acquire and enhance diverse skills to meet future needs; passionate for knowledge, research and exploration.
Focuses on the government's goals	Strong advocate in achieving the government's objectives; adds value in all aspects of work performance relating to national goals.
Smart, effective, and efficient	Adopts a critical, analytical style of thinking, is mindful and gutsy of all decision parameters in achieving the most desirable outcome.
Agile and fast	Creates an environment which promotes and empowers change, achieving goals in the quickest possible way and makes efficient use of available resources with self assurance in different situations.

This Global Leadership Model directly guided the exploration of new tools by defining attributes that they should foster. The aim of this platform is to enhance the way the government manages employees as well as enables citizens to enter the public sector. As a key function the tools aim to enable HR managers to find individuals with relevant experience that match the needs of new projects, programs, and job vacancies emerging in association with the public administration modernization efforts at better fulfilling citizen expectations and needs. Moreover, the platform also aims to direct meaningful notifications and personalized suggestions to support employees' personal development and hence empower the public administration to better serve expected services.

Prior to the AI enhancement the platform's legacy version was designed to invite individuals, or employees, to enrol in the system if they are looking to access the Prime Minister's Office for career advancement or to enter the public service. These employees were then asked to fill in a range of fields that intended to capture their education and work experience. The fields are similar to the fields you would see in any other employee or professional profile platform, like LinkedIn, with past and current job titles, dates of employment, or descriptions of work. To add further data employees applying to particular GLP activities are mandated to fill out a number of internally-developed personality tests. These are a battery of questions designed in partnership with psychologists that explore psychometric behavioural, competency, and leadership dimensions of the employee. All of these come together to create a 'profile' about individual employees. This information is meant to be used by the HR managers, such as personnel from human resources departments, leadership programs, or advisory offices. These HR managers look for quality employees to fill programs, projects, or departments. They also want to have deeper insights about the talent pool, such as ranking of employees based on how their profiles match perceived excellence.

The high volume and poor quality of these data generated and collected by the legacy platform made the matching process difficult, cumbersome and time consuming. The overly standardizing nature of the legacy tools stood as an impediment to identify the right candidates in a dynamic and fast changing environment the platform is designed to support. To deal with these challenges the government decided to put out a proposal for an AI company to develop a series of interfaces and algorithms to better support the government's employee acquisition and management program. The goal of these technologies according to procurement documents is to simplify the use of the platform, personalize content to better support personal development, make automated profile recommendations, reduce manual work associated with employee management, enable predictive analytics, and classify user content in meaningful ways. In the end, the design of the AI solution not only provided a more flexible solution to support different users' needs allowing the tuning of the system to the local needs, but also structured and defined how algorithm's personalization was delivered and what was possible to personalize. The AI solution, instead of being a neutral tool, became the key agent in shaping employee recruitment. To explain how and why the functional characteristics of technologies gained this leading role, it is important to detail the design process and the functions of the AI solution before discussing how and why it became the key agent for employee scouting and value creation.

5.4.3 AI process and functions

As will be discussed later, the initial proposal involved building a standalone python-based AI service that would occasionally search employees for Global Leadership Model attributes which were pre-defined as searches. If employees had some of these attributes the tool would tell the HR teams. But conversations with the AI development team and project owners led to the recognized need for new interfaces so HR managers could make searches themselves. This led to the development of a basic ecosystem (Figure 5.2) where an internet browser frontend using JavaScript manages users like HR and employees, while a separate python backend processes queries. Note that throughout this chapter grey boxes denote algorithms or code that processes calculations, white boxes denote data objects, and white circles denote users.

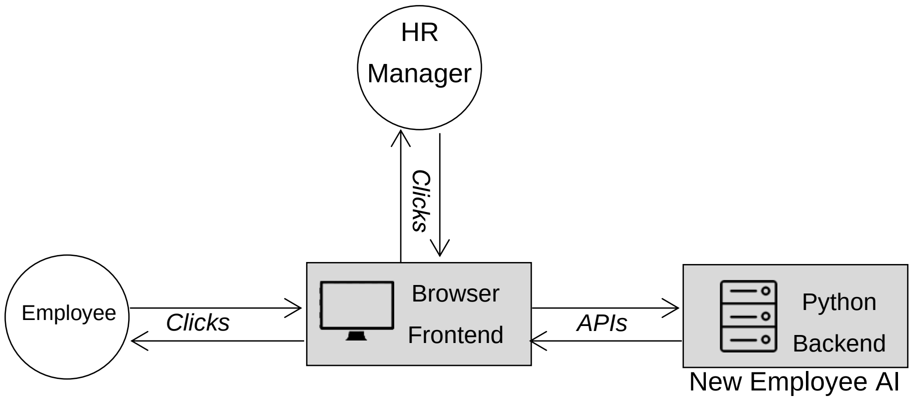

Figure 5.2. UAE employee management portal users and system

The users, in this case HR managers, make decisions that strongly influence or even govern employee acquisition strategies and hence how the public administration will identify priorities and respond to citizens' needs. Their use therefore of algorithms that automate the identification of employee characteristics makes this case particularly interesting and revealing. This investigation follows the process of how recommendation engines define employees, and how HR managers requests for increased personalization of these engines led to reconfigurations.

5.5 Observations – Identifying employee characteristics

The GLP pursued a goal of having every employee registered in Qiyadat which then builds a digital representation of each employee. The AI platform would then draw from this pool of digitally-represented employees to run queries and make employee selection recommendations. Employee recommendations would not come from a direct interaction with an employee, but an indirect interaction with them. Algorithmic decisions and the subsequent reactions of the HR managers happen over the digitized objects that were created by the employee interacting with the profile submission page. As seen in Figure 5.3 when employees log into the platform they are asked to fill out a series of fields which create several SQL database tables.

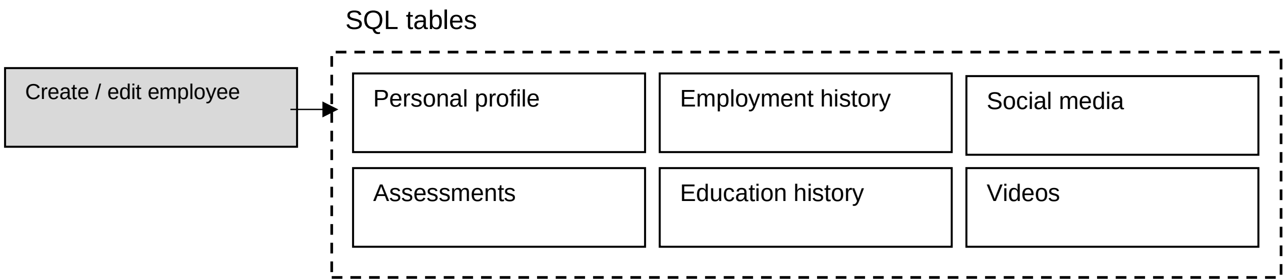

Figure 5.3. The storing of employee profiles

This database constitutes the data source for the employee acquisition AI system.

5.5.1 Pre-processing employees

An important process occurred nightly to pull from these user profiles to create special objects designed to enable algorithmic matching. Developers called this the preprocessing of employees. At midnight every new user or every user who has updated their profile in the last 24 hours has specific field-types from their profile extracted and placed within a new data object. The purpose of this object is to convert keywords from their profile into a format that can be interpreted by the matching algorithm later.

Preprocessing involves a number of distinct exercises treated section by section. The first is to create tokens out of the strings of text, as seen in Figure 5.4. Using a package called NLTK individual words are outputted from the lines of text from each profile section. At this point the object ceases to consider the position of the words relative to each other and are stored in an array. Other cleaning includes the removal of non-alphanumeric characters and the converting of all characters to lowercase format. The most common words like, ‘the’, or ‘and’ are also removed, thanks to globally maintained lists of common words worth ignoring. These are called ‘stop-words’. This all happens automatically.

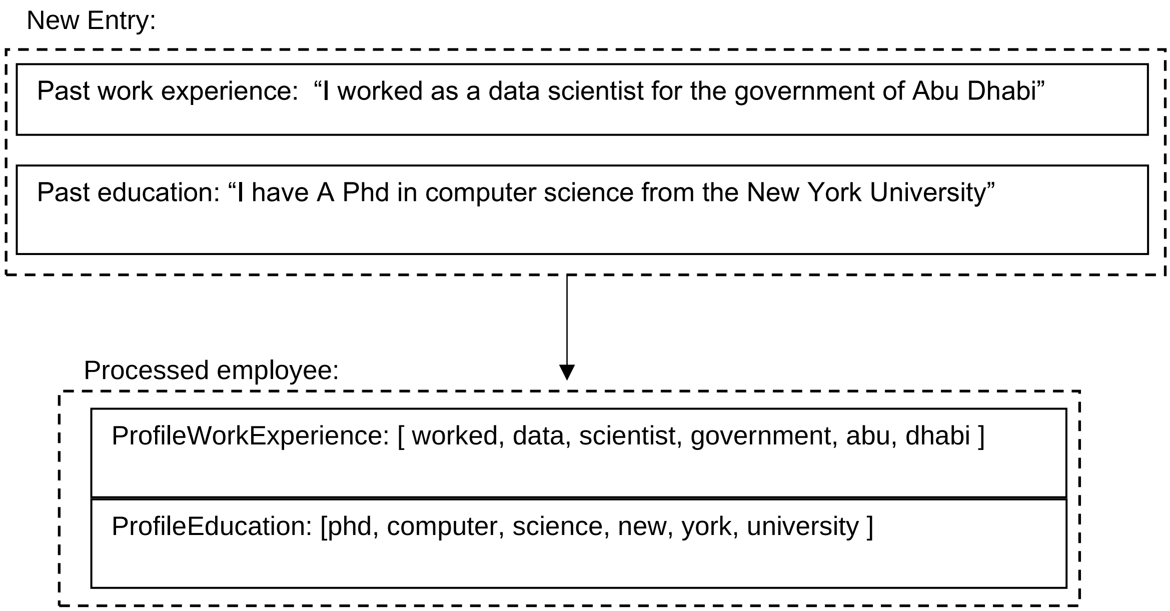

Figure 5.4. Tokenizing profiles

Following this, the same words are processed to detect lemmas. Lemmatization involves reducing a word down to its stem. In the case of ‘computer’ it would be ‘compute’. Then it involves finding related words using the same stem, like ‘computed’ and ‘computing’. Arabic algorithms follow a similar pattern, although they use additional internal and external packages. These help accommodate the rich inflectional and cliticizational morphology, high degree of ambiguity, and the many dialects of Arabic (Monroe et al., 2014). Clitics are unstressed words, typically functional words that cannot stand on their own. English mostly only makes use of clitics in colloquial speech, like, ‘em in “go get-em”, whereas Arabic makes much more extensive use of these.

Industry-specific stop-words are also removed. These are words that throughout testing users and developers agreed were not relevant and were diluting the quality of the results. An example was the word ‘government’ because it was a platform of government employees. This meant the word appeared in search fields too often and made the task of distinguishing employees more difficult. As seen in Figure 5.5, profiles take on increasingly modified forms thanks to these natural language processing techniques.

Processed employee:

ProfileWorkExperience: [work, worked, working, works, data, scientist, science, scientific, abu, dhabi]

ProfileEducation: [phd, computer, compute, computing, science, scientist, scientific, new, old, york, university, college]

Figure 5.5. Lemmatization and removing stop-words

In early testing the system generated results from words that seemed uninteresting, while not giving special relevance to unique or interesting attributes. For example, a processed talent may have keywords and related lemmas ranging from irrelevant like ‘welcome’, to more relevant hints at attributes like, ‘direct’, ‘experience’, ‘enjoy’, ‘manage’, to unique profile attributes like, ‘bilingual’, ‘scientist’, or ‘certified’. The latter keywords were of interest to users making search queries for example. To give these latter keywords greater relevance in the system an additional layer was applied that checks the rarity of each keyword to a globally-trained score of rarity. Using TF-IDF (Ramos, 2003), a multiplication between a ratio (TF) and the logarithmic of a ratio (IDF) returned a score between 0 and 1, but in practice the majority were lower than 0.5. The lower the score, the less likely a word is to appear from a corpus of text, or in this case a globally-trained corpus. The system would favour rarer scores, but common scores were still present.

As simplified in Figure 5.6 these pre-processed talents, in completed form, include keywords with rarity weights applied to them that attempt to represent their experiences and profile. It makes no specific reference to when the keywords appeared in a profile except which section in general it came from, although results can be filtered by numeric or categorical fields like number of years of work experience. Filter logics are less interesting for this investigation because they do not operate on social learning and are far easier to explain. Once completed, these nightly employee objects sit idle, awaiting a direct user search.

Processed employee:

ProfileWorkExperience: [work:{0.84}, worked:{0.82}, working:{0.89}, works:{0.87}, data:{0.56}, scientist:{0.37}, science:{0.36}, scientific:{0.41}, abu:{0.13}, dhabi:{0.11}]

Figure 5.6. Weights added to employee keywords based on rarity

5.5.2 Setting dynamic section weights

When a search is initiated an algorithm is fired that sets a weight for specific sections of the profile. This was not an initial part of the design of the system but it was decided that in order for the results to make sense for HR managers certain sections needed to be more important than others. For example, finding matches between search queries and a user profile in the 'past employment' section was deemed more important than 'education'. Education was nonetheless also deemed more important than matches found in 'achievements and volunteering'. These were human decisions, especially at first. The original weights can be viewed in Figure 5.7.

```
prior_knowledge = {
    ProfileAchievement.ProjectTitleAndEvent: 40,
    ProfileAchievement.Description: 30,
    ProfileAchievement.Role: 20,
    ProfileEducation.FieldOfStudy: 70,
    ProfileEducation.Organization: 10,
    ProfileEducation.Country: 5,
    ProfileEducation.Title: 20,
    ProfileEducation.DegreeItem: 20,
    ProfileMembership.Organization: 2,
    ProfileMembership.Role: 3,
    ProfileTraining.Organization: 10,
    ProfileTraining.Title: 20,
    ProfileWorkExperience.Organization: 15,
    ProfileWorkExperience.OrganizationType: 2,
    ProfileWorkExperience.Country: 5,
    ProfileWorkExperience.JobTitle: 70,
    ProfileWorkExperience.Industry: 50,
    ProfileWorkExperience.WorkField: 8,
}
```

Figure 5.7. Initial pre-defined section weights

However, developers were wary of setting a static weight for these sections so an additional layer was designed to allow these weights to learn from similar past searches. As seen in Figure 5.8, a search is first compared to past searches. If any past search included a similar keyword its computed weights are included into the calculation. Combining static predefined weights for sections with weights from similar past search involved going through each profile section and multiplying the static weight by 0.3 and then adding this to the average of section historical weights multiplied by 1 minus 0.3. This value was chosen after experimentation with different ranges until results began to appear sensible. The team did not want the weights to be overly determined by past searches, as some HR users may bias results. But they also did not want the influence of past searches to be so minimal it could not be detected. Because of this process each search has a slightly different set of initial section weights than other searches if there were matches to past searches, because they slightly impacted each other. Over time, search results began to differ and change over time by the use of the tools by HR managers. Whether influenced by a past search or not, these weights are used in the matching process.

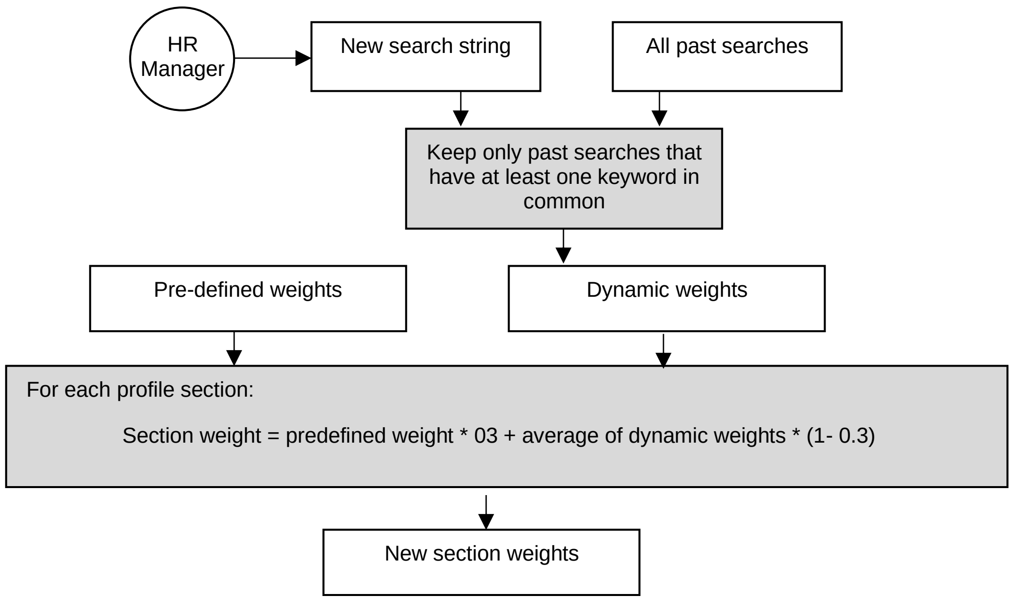

Figure 5.8. Adjusting section weights from similar past searches

5.5.3 Creating the search object

When an HR manager makes a search query on the system the text from the search query will be tokenized and these tokens will be lemmatized. Unlike employee profiles however, a few additional layers are added to the search query. Rather than limit the final output to individual keywords relevant bigrams are searched for as well. A bigram is when two tokens are found next to each other. Every pair of words are compared in terms of how likely they are to appear together in a corpus of text, similar to the rarity score of the employee objects. If there are bigrams that return a high enough score, they are added as standalone keywords even if made of multiple tokens. A common example that was often tested for was 'data scientist'.

So now upon initiating a search, section weights have been generated and a search object has been built (see Figure 5.9). At this point the search object is ran through a matching algorithm that looks for keyword matches from the search object in any section of an employee profile. Because it matches all search words, including lemmas and stems, there are a high number of matches to employees. The developers sought a way to have a dynamic matching score for employees so that the system would recommend certain matches above others. That is, the recommendation engine would begin to include ranking scores for employees.


```
“searchObject”: {
    "params": { "numberOfCandidates": 25 },
     "Search" : { "Header": "A senior programme manager" },
       "AdvancedOptions": {
          "EducationLevel": { "value": "Bachelor",  "mandatory": true },
          "FieldOfEducation": { "value": "Pedagogy", "mandatory": false },
          "Industry": { "value": "Education", "mandatory": false },
          "Location": { "value": "Dubai", "mandatory": false },
          "LeadershipPoints": { "value": 1000, "mandatory": false },
          "YearsOfSeniority": { "Min": 2, "Max": 10
          }
      }
    }
```


Figure 5.9. Example search object

5.5.4 The matching algorithm

To recommend and rank employees the algorithm takes the list of keywords from the search and looks for a match in an employee profile. To speed things along the matches prioritize the most important sections as determined by the pre-determined section weights. If no matches are found in the early sections the algorithm may skip an employee and move to the next. If there was a match found in the section, e.g., a query included the bigram “data science” and a profile included “data scientist” in the past experience section, then it will add a score to this section, ‘past experience’. It will add a score by the value of the keyword weight determined in the pre-processing stages, which is a float and may have many decimal points as seen in Figure 5.10. Consider the query, “I am looking for an employee with: ‘experience managing data science projects’”. If a keyword in the pre-processing scored very rare for say ‘data science’, it will result in a higher increase in the ‘past work experience’ score than a match that is less rare, like ‘manage’. Special rules are added to the algorithm’s processing to give extra weight to bigrams, and to give extra weight to matches in the work experience labelled as ‘current’ instead of ‘past work experience’.

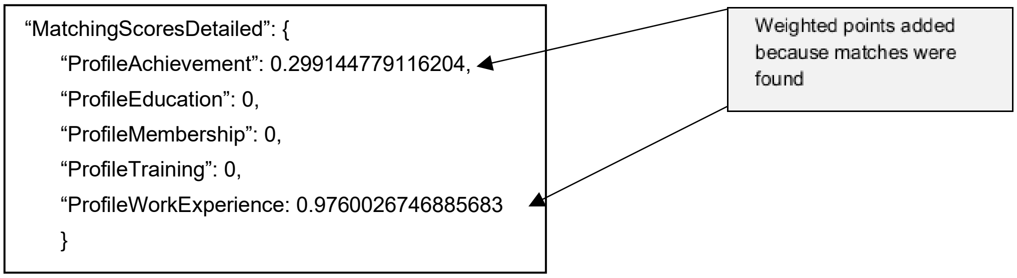

Figure 5.10. Section-based scores

Sections that have too low of a section weight from the pre-processing stage of the search query are dropped. Finally, after all matches have been found and section scores increased by rarity weights, the scores are combined to create a 'final' score for each section. A specific search query could have a higher match in one section but a lower match in another. The algorithm will return any number of results provided they have at least one section score with a high enough result. For the purpose of the HR search engine, the interface requests a maximum of 15 recommended employees at a time.

To facilitate the overall ranking across all sections one last scoring layer was created that sums weighted section scores from above, then applies a sigmoid function that maps this average into a number in the interval from 0 to 1. The closer the score is to 1, the better the match. This implies the actual value of the suitability score is an abstract representation of the matches, for it combines distinct attributes into summary scores and distorts them using a method to force them to fit within a range (0 to 1).

Similarly, a separate but related algorithm was developed to fetch the 'exact' keywords between a search query and a profile. This result was then added directly to the resulting employees from the recommendation engine. This was built due to users requesting a better sense of "why" the employee was selected. Having a score for each section like, 'past experience' was not descriptive enough, they wanted to know what words from that section in particular appeared. This was an abstraction like the suitability score in that it only returned direct matches, but did not return matches with stems or lemmas, and ran on a logic or computation separate from the original matching algorithm. This layer of 'matched keywords' nonetheless supported HR managers in interpreting the quality of the employee matches. It also allowed users to remove keywords dynamically from the "key reasons" which would become new stop-words for that particular search. Users could both improve the quality of the matches to their personal needs and teach the tool for future uses in the process.

5.5.6 Machine learning

While the tool sets section weights based on past searches, this did not qualify as machine learning because no formal model was trained. The HR managers and the developers wanted to expand the quality of the learning so that the tool could adapt to individuals using the tool at a quicker pace and in a more direct way. A machine learning layer was implemented much later in the development of the algorithms. As seen in Figure 5.11 this algorithm takes the initial section weights generated at the moment a search is queried, but waits for feedback. A new interface was designed to facilitate direct 'training'. HR managers are given a range of, by default, 15 top employees from the matching algorithm. These are sorted by the suitability score discussed above. However, HR managers are given a chance to further shortlist this 15 into a list of their favourites. The interface was developed to encourage picking around 1 to 5 candidates on average. If the system generated 15 employees and the HR manager shortlisted this to say 3 employees, then a model was developed to learn from this. For each selected employee the system set error to be 1 minus the candidate suitability score. This would inform the model that a high suitability score that was not selected was a more egregious error than a low suitability score that was not selected. Section weights, which once started with a statically defined value before being nudged by past searches, are now no longer static. Section weights are modified by a back-propagation algorithm and then are normalized to total a sum of 1. As a result, each new query will be different from the last not only from past searches, but from past successful matches. That is, the decision to shortlist a recommended employee informs the model to give greater preference to matches from certain sections.

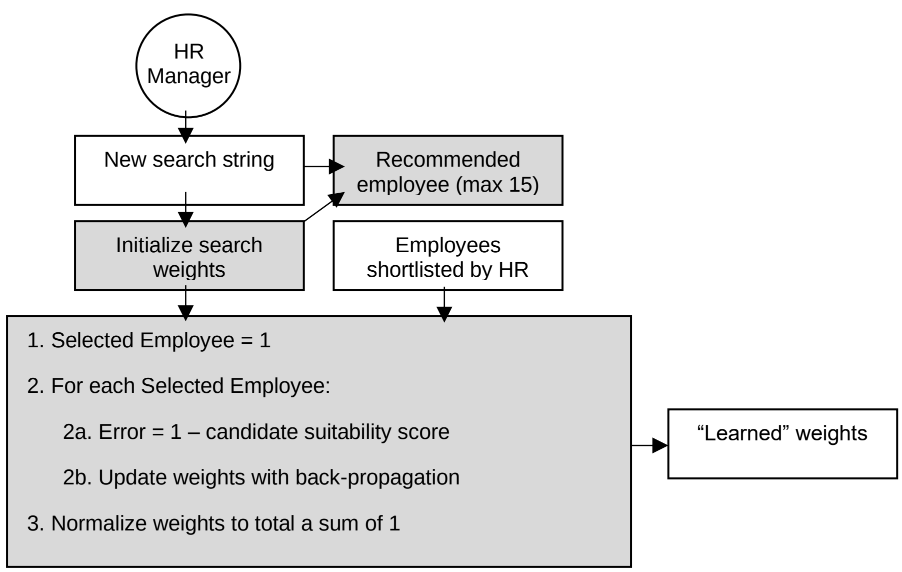

Figure 5.11. Machine learning algorithm

5.5.7 Digitizing ‘difference’ across employees

The process to upload information about oneself as an employee with employment history, education and interests is the process of digitizing the self as an employee. This digitization serves the purpose of standardizing individuals into formats that are ready to be interpreted and calculated upon by algorithms.

In this case, employees are arranged in a way that allows the system to detect and emphasize differences across individual employees. What makes one person a better match than another during employee selection, it was assumed, comes down to textual characteristics that differentiate employees. Some characteristics match specific needs better than others. We observed that in this case ‘difference’ between employees, which helps define them for the purpose of selection, are digitized through section weights for example. Designers of the system assumed an individual with characteristic matches that fall within one profile section over another, like ‘work experience’ instead of ‘interests’. This was facilitated by what was at first a pre-defined weighting determined by developers and the HR managers in user experience sessions. But over time these weights were further modified by machine learning.

Another way employee characteristics were digitized and managed explicitly by algorithms was an assumption that characteristics that involve ‘rarer’ words will be more relevant for a recommendation engine than assuming all matches are equal. This ‘rarity’ was determined by a globally-trained package, meaning this particular algorithm would operate under logic and data not available or explainable to these users. These are highly social data models because they are trained using a vast corpus of publicly available English words. Another assumptions that was configured into the system was expert-derived stop-words that were deemed not appropriate for this tool and are filtered out.

In the earliest iterations of the tool, a model was chosen that favoured accuracy over explainability. For example, scale vector models were explored to facilitate the match between a search query and a profile. The challenge with these models is that they return a result, match or not, but do not have interpretable ‘reasons’ or interpretable scale. ‘What makes one employee a better match than another?’ asked HR managers when using these early versions. Instead, an entirely different architecture was designed, built around models that release some degree of explainability. Scale vector models could not do this. As discussed, this was managed by giving scores to specific sections, based on word rarity, past searches, and machine learning from successfully identified employees. This is not to say all decisions to favour accuracy are necessarily inexplainable. Algorithms have strengths and weaknesses in different contexts. This happened to be a trade-off in this particular instance.

While the algorithm configures matches in highly structured ways, the decisions behind their design were highly iterative and involved data models that were trained by complex social language. Upon the completion of the tool, even if the architecture was no longer being modified by developers they remain ever-changing due to the fact that HR interactions further refine how ‘difference’ across employees is identified and operationalized into recommendations.

5.6 Analysis

5.6.1 Mapping interactivity and standardization

The case allows for the exploration of the application of a methodology from interactive computing (Goldin et al., 2006) to better shed light on how organization capabilities selection designed to enhance public administration’s ability to create public value is shaped by AI. This has allowed for the better finding of key moments where seemingly deterministic algorithms are fed data objects, the key interactions with the user, and the specific ways these data objects are transformed or distorted. Mapping out these components enables a fuller understanding of how the identification and selection of the needed organization capabilities is mediated by AI. Consider Figure 5.12, a flowchart of the identification and selection of the needed organization capabilities that has been inspired by Li and Jagadish’s Systems Architecture for Interactive Natural Language Interfaces (2014). One the public sector civil workers (element 1) are presented with a user interface (element 2) that encourages them to provide inputs about their personhood as employees. That is, they are guided through a series of pages that ask for different fields to be either typed, uploaded, or selected. This is the first aspect of functional simplification directly impacting the user experience. The way the information is presented constrain how the user will fill this in. But this extends beyond the frontend. Some backend systems, for example, cannot accommodate unstructured data. This informs the frontend to limit the type of acceptable entries. This may include limits on characters being used, impacting less known languages, or limit the number of entries. Another way functional simplification can limit personalization was revealed in the scattered and unstructured nature of the legacy system. The project champions complained that users created many alternative versions of the same entry, due to spelling mistakes or uses of abbreviations. For example, some employees typed, “Prime Minister’s Office”, as a place of past employment, while others, “PMO”, and others still, “Prime Minister Office”. The frontend was not able to automatically detect typing or send a query to the backend to identify suggested matches. But given this created difficult data that impacted the entire service experience, the frontend was changed to adopt a framework that allows for ongoing calls to an API engine.

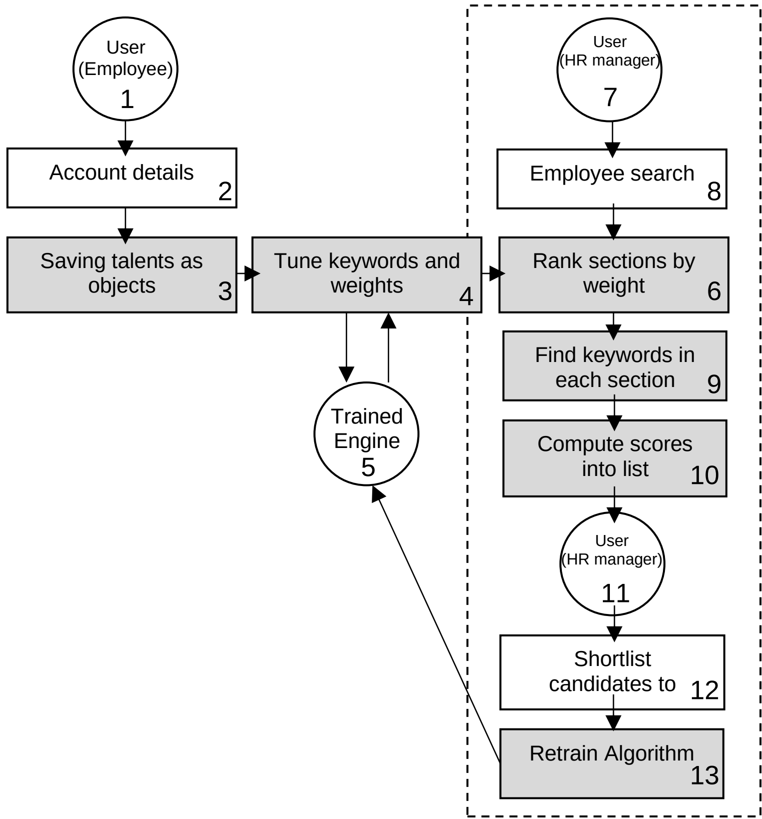

Figure 5.12. Algorithm / User interactions in the matching algorithm

Once a set of data is accepted by the browser and successfully sent to the backend, the information needs to be passed through a series of standardization and processing algorithms (element 3). These look for certain information in certain fields, and recombine them into a standardized object, in our case JSON. While these objects did allow unstructured data, like long text fields, the specific fieldnames like education or work experience needed to be identified. This functionally simplified object is both powerful but also significantly shapes the rest of the employee identification and selection process. It is powerful because developers, algorithms, and browsers can all interact with the same objects and readily find information within it. Apps are built to these objects, and their flexibility and smartness is only possible due to this processing. Within these algorithms a number of assumptions are often made, for example certain characters like apostrophes are often stripped out, to make later natural language processing easier. But this implies that not every stroke left by a user makes it to the final object. In fact, certain fields are filled by users but not processed at all, because no clear use case had been developed to make use of it. What is processed is functionally closed by the algorithm. These uncoded aspects of the user experience are lost. That is, any aspect of the employee competences and experience not directly identified and coded for, are not accounted for by the algorithms.

With this information saved, the later algorithms can now do their work. One process occurs nightly. This process (element 4) involves collecting all of the sections from the employee, and then applying a series of calculations to give a rank to each section based on hard-coded parameters which are internal to this algorithm. These weights were set by the developers. However, the ranking is also influenced by past learning that has been taught by users through their interactions (element 5). While the hard-coded parameters are interpretable, interestingly, the learned engine is not a simple element to just open up and read. This is often where the term black box is perpetuated. All employees are then not only processed into data objects, but custom ranks are also applied nightly (element 6) which change based on the users' feedback and based on other employees that have been uploaded, as the weights are relational.

The platform is now ready to make calculations upon these objects representing employees. Imagine a platform user (element 7), such as a leadership manager, is ready to make a search in the database. They are looking for employees that have certain skills, or certain employment histories. They want the search to be smart enough to find similar searches, manage spelling mistakes, and more, rather than just be limited to direct queries. They can do this through an interface (element 8) that has similar limitations and constraints as element 2. The search details, such as text or filter requirements, are sent to collect the nightly ranking that was conducted previously. The search then gives greater priority to those sections from the trained engine that are higher weighted than others.

Finding matches were the end of the algorithm originally, but over time an issue of explainability emerged extensively. Platform users, in testing, were wondering 'why' the algorithms made the choice they did. Without knowing the source of matches, users were less able to judge the algorithm and then meaningfully continue interactions with it. The AI development team then spent a great deal of time developing a supplementary set of algorithms that look through each ranked employee and find keywords representing the search matches (element 9). If the match was made through smarter connections, such as word associations instead of direct matches to the search query, then this information is also sent. This is then displayed to the user with each employee, giving the user an opportunity to understand the cause of the match.

But this explainability question did not end here. The focus on understanding the algorithm's decisions, specifically the ranking, led to platform users wanting to be able to meaningfully sort and rank the users in a more intuitive and human way. While the ranking algorithms from the matches do provide a sorted list, there was no way to understand the difference between the first employee in the list versus the last. Or the first versus the second even. An additional supplementary algorithm was developed that intended to create a single score for each matched employee, that could be interpretable by the platform users. This has been called the 'suitability score' (element 10). This itself is an interesting supplement and is worthy of deeper review. For example, the developers opted to use a mathematical function called a sigmoid function to convert uninterpretable scores that the ranking algorithms were giving, but convert them to a 0 to 1 score. This gives users a 'sense' of the match quality, with some employees returning 0.8 versus others returning 0.2. One could be interpreted as an 80% match and the other a 20% match. Although, we stress the term interpreted, because it is actually a model representing a match, and not an actual percentage score. Similarly, the original responses were clumping results below a threshold that made sense. The team then decided to apply the same function twice. A quirky choice from an outsider's perspective, but these choices reveal what the developers often call the "art" of data science. Trial and error involving the use of various mathematical and computational functions and features. Functional simplification and closure have now left the realm of simple functional limitations, and has moved into a more complex mathematical and programmatic ecosystem of applied cuts, negotiations, and distortions to the data from the developers, who were themselves shaped heavily by user feedback.

The learning does not end here though. The team was keen on finding a way to get user interactions to train the ranking, so eventually the algorithm will respond to the kinds of needs and priorities that its own leaders have, rather than being constrained by the design stage of the algorithm. That is, it needs to show stable agency in making decisions about ranks. To achieve this, the team developed a new process. The algorithms may return up to 15 matches for example, now with key reasons and a suitability score. But what if these technical distortions do not match the users actual ideal search result? The system asks the employee searchers (element 11) to 'shortlist' the 15 into their own favourites on the interface (element 12). This shortlisting is an interesting process. Mathematically, a learning algorithm assumes that those who were shortlisted were an ideal match, and that those that were not shortlisted are not. This creates an opportunity to train an engine (element 13) to continue to find employees with similar parameters as those that were shortlisted, and avoid employees with similar parameters to those that were not. This is then fed back to element 5 to be used in the next nightly recalibration of weights. Thus, user shortlisting in one day shapes the ranking and future results of the next day. While this learning is indeed occurring via the subjective interactions of users, we have shown extensively how various stages of the algorithm apply different functional simplified constraints and distortions on the functioning of the platform. That is, AI offers a greater ability to take deep contextual and unstructured details into ICTs which were previously much more static and standardized. However, to do so, systems need to be developed around the forms and limitations of the technologies enabling these contextual computations.

All of this revealed evidence about the extensive way in which algorithms shape how employee identification and acquisition occurs within AI-guided e-services, even if these new emerging tools are themselves deeply dependent on individual interactions. This is interesting in and of itself, validating perspectives raised by advocates of interactive computing. For example, it implies that even if each individual algorithm in the process is predictable and deterministic on their own and in isolation, there is nonetheless no way to predict actual outcomes in the system, because individual and subjective search experiences can happen at any time, asynchronously. And concurrently, these subjective search experiences are enabled through the functional standardization of the ICTs used.

To answer our initial research question we can conclude that learning algorithms impact public sector organizations' ability to identify the characteristics of the employee that are needed to support the creation of public value, and do so through the structuring and ranking of employees and search queries. The way in which these algorithm function creates an opacity which hinder the ability of the public sector organization to fully control the selection of the best employees needed to support and enable public value reaction. Hence the algorithm takes control over what values the public sector origination will be able to create and inherently how these values will be created.

5.7 Conclusions

Our case reveals that AI-guided employee identification and acquisition, the combination of algorithms, data objects, and interactions, are always evolving, and this evolution is shaped by functional realities which do not necessary overlaps with the contingent needs of public value creation which drive the design and adoption of this technology.

As the public sector adopts new tools that can help predict the needs of citizens, and the employees needed to fulfil these needs, clear implications emerge. Given public value requires a tireless pursuit of user needs and employees' competences which are ever-changing, systems need to be ever-changing as well. Machine learning offers an opportunity to harness ever-changing needs in a way technology could not enable before. A mapping of the evolution of AI algorithms enables a greater awareness and defensibility in being able to articulate the ways in which AI adoptions can impact public value creation. In this case we observed the development of a recommendation engine that helps with employee selection across the public sector. Personalization and AI it turns out can help with the identification of values and traits in potential employees that are aligned with political values. This helps the public sector retain control over the way the AI defines which employees to scout and which ones to not. But inherent within this is the risk of bias embedded within the AI, bias of the designers, as well as broader privacy considerations. Effective utilization of personalized tools through the use of AI and machine learning does not happen automatically, it requires attention and care on the part of the public sector. As a result, the role and nature of public managers appears to be shifting towards managing and monitoring how aligned these tools are with ever-changing public value needs. This led to a need for highly interactive tools and an intimate collaboration and co-production between algorithmic developers and the public sector managers procuring them.

Interestingly, much of this demand for interaction was driven by the developers themselves, always eager to shape their data towards real needs. As the use of data becomes increasingly ubiquitous, we can expect the needed skills and competencies of the public sector to shift along with it. The public sector has the opportunity to play a particularly important role in ensuring this data is used to drive public value while minimizing risks associated with its collection.

Chapter 6: Learning algorithms in organizations: a process view of algorithmic explainability to algorithmic opacity

6.1 Chapter preface

This chapter is based on a paper co-written by Dr. Maha Shaikh, Dr. Antonio Cordella and the thesis author. At the time of this writing, this work has been submitted to an Information Systems journal and has received several rounds of feedback. This paper is undergoing another round of edits, thus there will be changes to its current form before publication. Nonetheless, this work in its current form played an important role in answering the research questions set out in the thesis.

This work follows from the findings of chapter 5. The work in the previous chapter opened up the black box of emergent technologies, particularly natural language processing and machine learning, as it was used to recommend employees on an HR management portal. Interactive mapping was used (as discussed in chapter 4) as a way to interrogate the code and then the decisions made behind the code by key designers. As the black box was opened and the design process mapped, a few interesting observations stood out: the persistent lack of explainability in some emerging algorithms, and the ever-present issue of designer bias amplification. The authors of this paper intended to unpack the former, issues of explainability, but contributed to the latter indirectly as well.

The authors of this chapter set out to understand how designers seek to make algorithms more explainable. The users in this case, HR managers, raised questions to the designers as the algorithms were configured and reconfigured. 'How did it make this decision?' This question was active in the early cycles of algorithmic design, but as the developers added new interfaces, modified algorithmic parameters and logic, and activated machine learning, this question reduced in number. Users began to trust the tools and use them. However, through interactive mapping it was discovered that as these new tools were integrated and as answers to the question, 'why did it make this decision' deepened, a comprehensive understanding of the algorithms remained elusive. If anything, understanding got worse, even to a team of skilled and experienced data scientists, developers, and subject-matter experts. In seeking out how to make algorithms more understandable to users, layers and abstractions are added to help build an intuitive sense of 'why'. This is similar to reverse engineering, except that when engaging with components utilizing machine learning and natural language processing there is too much uncertainty for a full answer to be found. As reverse engineering occurred and layers or interfaces were added to 'explain' actions, the ability to fully 'explain' what was happening was replaced with 'interpretations' with how they behave. Even though these are ICTs with a seemingly deterministic underlying logic, the use of dynamic data mixed with the properties of the designed algorithms resulted in an inability to fully know. Some degree of transparency was achieved, in that the users were satisfied with the algorithm in terms of understanding 'how did it make its decision'. This was achieved by creating separate logics and new algorithms on top of old algorithms, such as returning specific talent keywords to give users a 'sense' of why the algorithm picked the employee that it did. The result satisfied the user, but was without scrutability because it was actually separate algorithms coming together to create an abstracted hint about 'why'. This led to lingering questions. Do all algorithms become inscrutable after designers seek explanation? When does algorithmic explainability get replaced with interpretability exactly? The work in this paper reviewed a single case of algorithms, but fueled a curiosity for more generalizability (chapter 7).

Deeper dimensions of bias are also explored indirectly in this chapter and are worthy of extended discussion. At times decisions are made that look as if designers are forcing the algorithm to behave in ways they would like. For example, the selection of an arbitrary attribute of '20' seems to lack concrete justification. Similarly, at times it is apparent that designers are well aware they are not subject matter experts when it comes to HR and are uncomfortable about making decisions on behalf of their users. Yet, at many stages in the algorithmic design, user input did not prove useful for solving the problem or informing the decision. Ultimately, the algorithms became a product of designers and their admitted biases, including subtle decisions made throughout, but are also the product of the requests for deeper explainability by end-users, HR managers. This aim of this research was not to answer whether or not this particular arrangement is free of bias. If anything, a recurring observation has become reinforced. When it comes to using natural language processing and machine learning to build a matching algorithm, bias is proving inescapable, as is inscrutability. Efforts at better explainability, in this case manifested in the question, 'why did it suggest this employee?' added more opportunities for designers to introduce bias through imperfect experimentation and guess work, and deepened the adoption of inscrutability in the process.

6.2 Introduction

Organizations are racing to adopt artificial intelligence (AI) (Furman and Seamans, 2018; Raj and Seamans, 2019). Yet, it is increasingly evident that relying on algorithms and learning algorithms more particularly, is giving rise to a need for algorithmic accountability (Gillespie, 2017; Pasquale, 2017). How learning algorithms behave and what criteria affect their decision-making process is still rather opaque (Burrell, 2016; Stohl et al., 2016). While scholars have argued that human decision-making can be equally non-transparent (Zerilli et al., 2019) there is a distinct need for technology to be transparent. Technology needs to be adopted and the adoption process is dependent on human trust in the system. There appear to be few if any checks and balances with learning algorithms but by contrast human decision-making such as employment panels are always open to scrutiny (Liem et al., 2018).

Scholars in different domains seek to understand algorithms and discuss algorithms in the context of algorithmic interpretability (Totaro and Ninno, 2014), algorithmic explainability (Abdul et al., 2018; Adadi and Berrada, 2018; Barredo Arrieta et al., 2020) and algorithmic transparency (Kellogg et al., 2020; Stohl et al., 2016; Waltl and Vogl, 2018). These terms are often used interchangeably. In our study we look to a rich source of data, our case study of AI developers working on cutting-edge learning algorithms to work on top of an HR system, to understand how the needs of understanding an algorithm are distinct for developers that create them and for the users of the algorithms. Both developers and users are baffled by how and why the algorithms behave the way they do but the processes they employ to reach for understanding are different. We are thus able to define and refine our appreciation of the distinction between interpretability (how the algorithm works) and explainability (why the algorithm works the way it does) to deepen current scholarship on learning algorithms.

Algorithmic interpretability refers to the ability of humans to understand the operations by which the algorithm reaches a specific decision (Biran and Cotton, 2017). It implies understanding what mechanisms have been involved in the decision-making process. Interpretability is not universal (Lipton, 2018). An algorithm can be interpretable in its entirety - all the possible outcomes can be explained - or only a specific subset of outcomes can be interpreted and hence understood (Guidotti et al., 2018). Moreover, not all users or designers have the same knowledge and skills. As a consequence different interpretation among users and designers can coexist (Weller, 2019).

Algorithmic explainability refers to the extent to which it is possible to understand the mechanism by which a decision is taken by the algorithm (Gunning and Aha, 2019). This goes beyond interpretability, which stops short of knowing decision-making mechanisms. Explainability requires an understanding of the functioning of different mechanisms that structure the outcome of the decision-making process. Learning algorithms are complex and difficult to unpack and hence to understand (Faraj et al., 2018). Interpretability and explainability are increasingly requested by users to overcome the opacity of algorithmic decision-making (de Laat, 2018) and support user adoption and use (Taddeo and Floridi, 2018).

Given the intrinsic technical complexity which surrounds the design of these systems often the users do not fully understand the how and why of the system's functioning. Hence designers and developers are asked by users to explain the functioning of the system so that users can confidently embrace the systems (Mercado et al., 2016).

To provide explanations designers and developers need to understand the how and why mechanisms of algorithms for themselves. Scholars have pointed to many reasons why learning algorithms can be opaque to designers (Burrell, 2016). For example, opacity results from the complexity of the algorithmic code (Faraj et al., 2018); it is also a result of the use of multiple algorithmic components which are not all known or under the control of the designers (Sandvig et al., 2014); and there are intrinsic characteristics of learning algorithms that redefine parameters and relationships with each additional data point which make it hard to explain how and why a decision has been taken (Faraj et al. 2018). The many reasons that produce algorithmic opacity need be resolved by developers of the algorithm to make the algorithm interpretable and explainable to themselves before being able to offer the needed interpretability and explainability to their users.

However, for the reasons mentioned above opacity cannot be always resolved which leads to our research question: *How can an algorithm be made explainable?*

To answer this question, we unpack the process by which developers question the way in which a learning algorithm unfolds in its use to manufacture interpretability and explainability. The findings in this paper sheds light on how the algorithm at the centre of an AI recommendation system (learning algorithm) is used and works alongside users. Our study offers a useful perspective to the working of machine learning algorithms because we observe and interview the developers creating such algorithms. We map the struggles that developers face with learning algorithms and interrogate the code of the algorithms, thus allowing for a way to open the black box. We can, through such work, make sense of how explainability of algorithms is achieved (Miller, 2019). The main contribution of this work is a process perspective and explanation of how learning algorithms continue to elude the developers that build them and the users of systems. We show that two processes of interpretability and explainability work together to strive for algorithmic transparency yet in the unfolding of the processes both developers and users working, amending and changing the algorithms, manage to increase algorithmic opacity instead.

This paper is structured as follows; the next section offers the theoretical underpinnings of our work followed by detail on our research methods. We then elaborate our findings followed by the theoretical development section. This work ends on the discussion and implications section with a clear conclusion.

6.3 Theoretical underpinnings

Technology can fundamentally reshape decision-making processes within organizations (Simon, 1973). There has been persistent scholarly interest in technology adopted and implemented in organizational settings to change decision-making abilities (Ciborra, 1996; Daft and Lengel, 1986; Galbraith, 1974; Huber, 1990; Markus and Robey, 1988; Orlikowski, 2000; Orlikowski and Baroudi, 1991). In organization studies the focus has been on technology's transformative impact on the way in which people structure their decision-making processes and control their activities (Bloom et al., 2014). In this body of work technology is considered a black box that impacts, shapes, and redefines organizational practices. In information systems the understanding is that technologies are designed and used to support decision making processes and coordination (Orlikowski and Barley, 2001). Information systems thus recognizes technology as the outcome of a process of black boxing that is influenced by organizational practices and technological use.

In recent years, advancements in technological innovation have transformed the functional characteristics of many technologies enhancing their *informating* capabilities (Kallinikos, 2006; Sharma et al., 2014; Zuboff and Maxmin, 2002). As a consequence, technologies expand their impact on decision making processes and organization practices by increasing the amount of information available to decision makers to the point where only "intelligent machines" are able to process and analyse the available information (Brynjolfsson and McAfee, 2016; Faraj et al. 2018; Zuboff, 2015). Given the pervasiveness of these information technologies in organizational decision making processes, several authors have proposed novel explanations to shed light on how emerging technologies redefine organizing (Alaimo et al., 2020; McAfee and Brynjolfsson, 2018). Scholarship has at the same time unpacked distinct emerging technologies like robots and 3D printing to make better sense of how decision making is affected in organizations (Barrett et al., 2012; Leonardi, 2013; Masli et al., 2016; Zuboff and Maxmin, 2002).

6.3.1 Machine learning and artificial intelligence algorithms

The advent of AI has offered innovative ways to analyse the impact of technology on decision making processes and on fundamental organizational task performance (Brynjolfsson and McAfee, 2016). AI indeed offers novel mechanisms to automate organizational tasks concerning the collection of information; information processing, and provision of solutions - i.e. decisions - that build on these different pieces of information (Ford, 2015). For example, the automation of organizational tasks and decision making processes redesigns the boundaries of authority within the organization delegating tasks that were previously under the responsibility of specific organizational actors and functions to AI algorithms instead (Von Krogh, 2012). Such delegation of organizational tasks to AI algorithms reshapes fundamental organizational functions, processes and relationships challenging established paradigm of organizational design (Shrestha et al., 2019).

Moreover, AI algorithms redesign decision making processes and organizational practices into codified processes that transforms inputs into outputs -i.e. causal mechanisms (Markus and Rowe 2018) that "make decisions based on computational optimization" (Shrestha et al. 2019) which is designed into the objective function of the algorithm. From an organizational design perspective these causal mechanisms can be used to optimize decision making processes that can be explicated into the code of the algorithm. Optimization here implies immediate processes and not the overall outcome or final decision. However, their use to support decisions that cannot be made explicit into objective functions increases the rigidity of decision making which instead of following human judgement, follows rules blindly which can lead to biased and unfair decisions (Taddeo and Floridi, 2018; Wu, 2019). In fact, human decision making can exercise judgement and intuition and hence handle ill structured decision objectives better than computational optimization (Cohen et al., 1972). To address this fundamental problem, Shrestha et al. (2019) suggest the use of hybrid models where the AI system can either automate the decision-making process or support it. In the latter case the human is in charge of the decision making process validation (Samek et al., 2017) taking responsibility and hence accountability over the process and its implications.

To support decision making processes and organizational tasks that are ill structured, AI algorithms cannot solely rely on computational optimization. A subset of AI algorithms, learning algorithms, are designed to learn from decisions taken in the past to better predict or proxy what current decisions ought to be to fulfil expected organizational tasks (Faraj et al. 2018). Learning algorithms rely on advanced computational features and capabilities to permit the algorithm to change and re-write itself to adapt and respond to variation in the environment (Faraj et al. 2018) thus overcoming the limitations of traditional AI algorithms which rely on static causal mechanisms to process inputs into outputs. By so doing learning algorithms can enhance predictability, transparency and accountability in organizational decisions and tasks that are ill-structured. Inscribing predictive frameworks of action accounted for by algorithmic code – i.e. algorithmic induction (Puranam et al., 2018) - learning algorithms black-box complex and unpredictable organizational actions and patterns into complex but formally specifiable interactions (Luhmann, 2005). Algorithmic induction eases algorithmic predictability, transparency and accountability. Black-boxing complexity into pre-determined processes that transform inputs into outputs standardizes and stabilizes decision-making into predictable, transparent and accountable processes. In fact, the design of an algorithm reveals how relational processes are structured in the scripts of technology and, at the same time, what other possible relational processes or interdependences are excluded by the very same scripts

However, once in use, both traditional AI algorithms and learning algorithms can involve inscrutability and can generate opacity in organizational decision-making and task execution (Faraj et al. 2018). Processing and analysing unprecedented amounts of data AI algorithms can make decision making inscrutable due to the complexity of both data and the algorithm. Learning algorithms can increase the opacity of decision making due to the complex and inscrutable mechanisms used to enclose and exclude causalities while shaping the algorithm to the evolving context (Samek et al. 2017).

However, inscrutability and opacity are not only the result of endogenous characteristics of the algorithmic function. The increased diffusion of these algorithms reveals the importance of the inputs provided by users for an effective design outcome (Negnevitsky, 2005). The users interactively provide feedback to help, tuning parameters, tweaking features, and providing more data to improve target setting. (Amershi et al. 2014). Given the role users have in the shaping of learning algorithms it is increasingly difficult to define clear boundaries between the effects of the algorithms per se and those of organizational practices and needs on the design and development of these algorithms. The social and the technical become tightly coupled blurring the boundaries between the two (Amershi et al. 2014). AI and learning algorithms have become intertwined with social and organizational practices that make it impossible to separate the two (Orlikowski, 2007).

To fully understand how the algorithm works and shapes organizational decision making the algorithm needs to be opened and unveiled – the black box needs to be opened.

6.3.2 Algorithmic interpretability

AI and learning algorithms have to be modified to accommodate the specific and sometimes contingent needs of the organization. This modification process involves adjusting the weights of different parameters and network of connections (Faraj et al. 2018) to align the output of the algorithmic process to the specific needs of an organization. The modifications, or *tuning* as software developers often call it, includes the developers of the algorithm, the code of the algorithm, and the users of the algorithm. In essence, what modification or tuning implies is a number of steps that include the opening up of the algorithmic black box, setting different parameters, amending weights and network relations to align the algorithm delivering the expected results, and finally, closing the black box to allow its use.

For example, an AI system used in organizational talent recruitment needs to be modified according to the specific characterises of the talents required by an employer in its potential employees. Developers define the different parameters and their relationships in the code of the algorithm to reflect the criteria that have been identified as key for the selection process. Once these adjustments are done, the system is used to select the applicants. However, the users might require the algorithm to be re-tuned if the results do not match their expectations. The retuning is now based on the new input from users plus the information of what was wrong in the initial search.

Thus, the modification process is affected by specific technical characteristics that determine the choices and functions (i.e. abilities) that the algorithm offers to the users. These abilities determine how the user understands the algorithm and what modifications can lead to effective tuning (Norman, 2013) as well as engagement of the users (Karahanna et al., 2018) in the tuning process. The technical features of the algorithm shape its abilities and reveal what is understandable by the users but also the potential algorithmic inscrutability and opacity (Faraj et al. 2018) that might impinge upon users engagement and participation in the tuning process. Algorithmic *interpretability* defined as "the ability to explain or to provide the meaning in understandable terms to a human" (Guidotti et al., 2018) depicts these attributes.

Where the results provided by the algorithm do not necessarily meet anticipations, the users expect to know 'why' the algorithm made the choice that it did (Kulesza et al., 2013). However it is often the case that developers do not understand why the system results in the outputs it does (Samek et al. 2017). The embedded functionalities of the algorithm - what the algorithm black-boxes in its code - can be so obscure and opaque that it becomes impossible to explain or make explainable the results it produces.

Since most AI and learning algorithms share the same problems of interpretability and explainability (Samek et al. 2017) an obvious question emerges: how can we make learning algorithms interpretable and explainable? In our study we focus on software developers building a system where learning algorithms are interpreted and explained to offer users a clearer understanding of how and why the algorithm works in the way it does.

6.4 Research method

The purpose of this research is theoretical understanding of learning algorithms in design and use. In order to open the black box of learning algorithms we collected different sources of data pertaining to one organizational setting. The need to reveal the inner workings of the leaning algorithm underpinned and guided this research which involved a novel ethnographic exercise of meaning-making around the evolution of the code and its design logic. This led to the adoption of an extended mixed-method approach.

6.4.1 Research setting

As part of the UAE's Global Leadership Programme, the Prime Minister's Office made a request for proposals for an AI provider to develop a series of tools to enhance organizational leadership capabilities and connect new talents to new opportunities. The AI was intended to sit on top of their existing system, a portal that collects user employment history, education, and psychometric scoring. The researcher was given unfettered access to source code, development documentation, and interviews with relevant stakeholders. Guided primarily by a desire to understand the logic behind the design of the algorithms, the researcher began collecting data about the negotiations taking place between the developers, the users, and the algorithmic outputs themselves.

6.4.2 Data collection

The data collection involved an analysis of code, the developer chat forum, tickets for amendments to the code, and interviews (see Figure 6.1). The primary focus of the research was the code. Code is supposedly predictable and simplifies information in determinate ways (Kallinikos et al., 2013; Luhmann, 2005). Despite this, code cannot be understood quantitatively. The complexity behind algorithms is determined through a negotiation between designer and the design of the code.

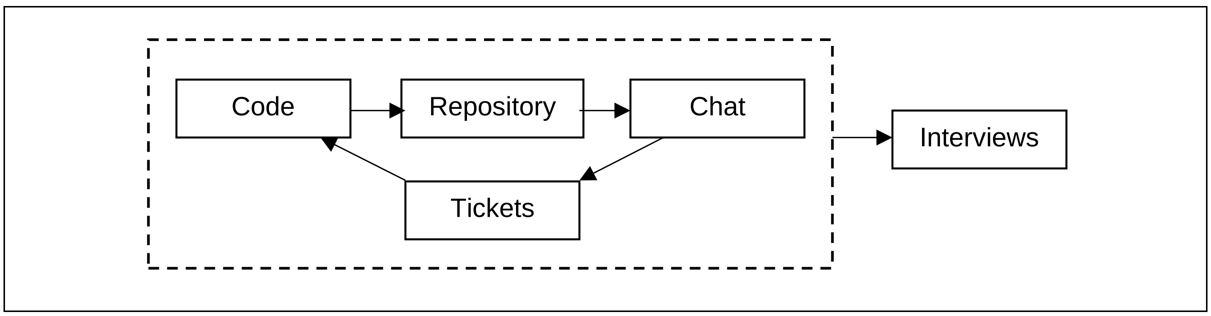

Figure 6.1. Data sources

As an author of this paper was closely connected to OrgX he decided to observe the developers at work. In this manner he could get closer to the conditions of study, the people, actors, systems, and institutions, and build a more meaningful understanding (Becker, 1996). Further care was taken to explicate what kind of role the researcher would take. The primary role for this particular project would be as an observer, but that necessitated occasional participation across the life cycle of the project. This observer-as-participant (Gold 1957) approach allowed for access to data while minimizing the disruption of its occurrence. That is, it enabled a balance between an insider perspective and an outsider perspective (Becker 1996).

Novel data collection methods were employed during this early descriptive phase of the research. All routes of inquiry began with the source code, with a goal of mapping out the logic behind each function. Drawing on Li and Jagadish (2014) particular attention was given to moments of interaction between users, the data that represents them, and the algorithms that iterate over this data. This guided the researcher on where to begin the journey through the code: following user actions. Following a user action, the data that is manipulated around those actions, and the algorithms that interact with them, slowly and meticulously the entire range of algorithmic steps were mapped out.

Code alone, however, only revealed so much. Even with competence in both core programming languages, the researcher could only guess why certain decisions were made. If interesting nodes or interactions were discovered, there was not always an obvious reason why they were chosen over others. There were default values or hard-coded weights set by designers. This was because the more tuning decisions made by the designers, the more challenging the understanding of the origin and purpose of each decision. To properly describe the algorithms and their interactions with users, the researcher had to turn to supporting documents, tools, and interviews. This created an iterative research cycle. As an illustration, consider the following:

- The researcher followed user flows until reaching the algorithm that returns the suitability score of a talent selected by the recommendation engine. In this algorithm the following line was observed (Figure 6.2):

```
low_matching_scores[section] = matching.keywords_sigmoid(section_score, 20)
```

Figure 6.2. Excerpt of machine learning algorithm showing scoring transformation

- The researcher noted two design choices. First, the selection of a sigmoid function when finalizing section scores, and second, the use of a parameter of 20 which seemed specific and human-selected.
- Curious to understand the origin of these choices the researcher turned to GitLab, the code repository. This allowed for a temporal view, seeing when developers pushed new changes to the code. Here, the researcher noted changes were made to this line a few times. It originated during the user testing of the early version of the recommendation engine and was modified when the learning algorithm was activated.
- The researcher turned to the developer chat, a tool called Rocket.Chat. Here, the researcher observed the following line:

  "*It seems to me that as more profiles with more complete data are being added, suitability scores might have become same/similar for a lot of profiles because perhaps when we tuned the parameters of the sigmoid, a function that transforms raw score into human readable out-of-100 score, we used old data. we can retune the function & perhaps impose a restriction on it to always make some difference between the scores of top talents or something if this is desirable.*"

- From this quote a number of interesting observations were made. The origin of the suitability score was to convert raw algorithmic scores into a human readable format. It is also revealed that the parameter was set and modified by tuning exercises. When old tuning led to unexpected results, the developers adjusted the parameter. The final value of 20 could not reveal the intricacies behind its selection. This was not understood until turning to supporting documentation.

- For further assurance of the connection between this development chat and the observed lines of code, the researcher turned to the project management software Jira. The ticket "PMO-100: Tuning for matching algorithm" aligns with the timing of the chats and the changes to the code.
- The above data revealed why the parameter was modified but did not reveal why 20 was chosen. An interview was arranged between the researcher and the AI lead. The interview that even the developer did not fully understand 'why 20', and that they often revert to defaults recommended by the broader data science community. Decisions made to further tune this parameter were done with a causal experiment, where a single change was introduced and then observed.

It is important to note that this process can become more challenging if the number of designers on a single feature are high, the work is spread out over time, and/or the organization has poor tracking. However, modern organizations making use of an agile methodology will tend to keep development cycles tight, well documented, and using focused groups of smaller teams. Even in these cases, there may be gaps in the full understanding.

Each iteration began with the source code. When interesting or unexplainable interactions were uncovered, the GitLab repository was used to identify the timing of the changes. The developer's chat revealed discussions taking place during the development of the code changes. Project management tickets from Jira would then confirm which discussions translated to specific coding actions assigned to individual developers. Through this cycle, observations emerged that could not be fully understood despite the triangulation of code, documents, and discussions. Interviews were then set up with the respective developers with directed questions designed to understand the social context that shaped coding decisions.

The code and repository observations involved reviewing the backend of the application, which housed the algorithms and database calculations, and the frontend of the application, which generated the internet browser interfaces that the users interacted with. The backend comprised 100% python code, made up of 1734 commits of code changes by developers averaging 2.1 per day. The frontend comprised 79% VueJS code, which was a modern framework for enabling complex internet browser interactions, 19% javascript, and 2% other including HTML and CSS for styling the internet browser interfaces. The developer chat was hosted by an open-source team collaboration application called Rocket.Chat. Table 4 summarizes a dedicated channel focused on the case project. Development discussions remained low in the first few months of the project but peaked between September and February. This overlapped with the period of ongoing user testing. Exposure to user feedback generated new tuning.

Table 4. Dedicated developer chat room

Month	Wordcount in AI development chat	Relevant description of the month's activity
May-18	484	The AI team started as consultants, supporting the PMO's RFP process
Jun-18	5559	The AI team submitted a bid for the RFP, seeing alignment between their capabilities and the PMO's needs. The rest of this month was focused on writing the proposal. This is where the first mention of the architecture emerges
Jul-18	454	After submitting the proposal, there was an opportunity to further clarify the proposed approach.
Aug-18	5468	With the project now initiated, this first month went over design requirements (shaped by user stories, personas, existing data, and other brainstorming).
Sep-18	10430	Development shifted to designs, wireframes, UX journeys. These reviews of the user journey dramatically shaped the future forms of the algorithms.
Oct-18	13944	A month of heavy UX testing, emerged issues include explainability in particular. E.g., suitability score emerged on the front-end, and then discussions about matching keywords
Nov-18	11105	A month of extensive UX testing involving 'gut checking' which also slowly shapes the questions of explainability, and changes the nature of the algorithms.
Dec-18	16656	A month of incorporating UX into new code.
Jan-19	17051	The launch of the machine learning resulted in a lot of interesting observations.
Feb-19	11721	Deployment and getting project approval for completion. There was mild tuning , but also interesting discussion 'describing' the explainability
Mar-19	7366	A month focused mostly on deployment, not a lot of new engineering otherwise
Apr-19	1305	More deployment, and a shift to supporting APIs not directly tied to searching/scouting. E.g., "more like this" matches talents to similar looking talents.
May-19	9251	Mostly deployment, as well as the testing of new translation services. Although there is some discussion/clarification of older algorithms, as well as an enhancement of the ML to incorporate further feedback from users who can enhance the power of individual keywords
Jun-19	8024	More deployment, and introducing a new Designer to the PMO to be dedicated to improving their other complimentary platform components
Jul-19	4011	Slight tuning to the suitability score, to deal with ties
Aug-19	1903	Exploring restricting into more generic APIs

As an interpretive exercise, links between chats and code decisions could sometimes only be presumed. Turning to project management software allowed for the linking of developer discussions to code changes. Table 5 summarizes the tickets or project actions assigned to developers. The tickets peaked between October and January, which was the heaviest period of coding. Tickets following January focused mostly on tuning and issues of explainability that emerged after the machine learning engine was activated.

Table 5. Machine learning tickets

Month	Total tickets	ML task descriptions
Aug-18	16	User stories and gathering of security requirements. Then ML research, followed by early ML proposals and setting up basic versions of the ML
Sep-18	19	Create first data objects (manually create talents), and begin building ML (Specifically a matching algorithm, by assigning early weights and simulating learning). Also explores other ML options, and begins wireframes, design and UX

Oct-18	53	Integration with live data, and heavy tuning of the ML (multiple successive tuning tickets). Begin to generate talent objects instead of manually creating them.
Nov-18	31	Building of explainability. And further tuning.
Dec-18	43	Further explainability (keywords, matching sentences, associations). And further tuning.
Jan-19	30	Further tuning. Addition of spell check service
Feb-19	18	Visual updates to enhance the user experience
Mar-19	25	Additional learning experiments (user like, remove keywords). Explainability, visualizing the 'why' to the user
Apr-19	13	Enhanced code for stability. Advanced filtering added.
May-19	13	Creating of dummy-data generator.
Jun-19	11	Implement new learning
Jul-19	41	Enhanced code, new AI services (not directly connected to the search feature), better deployment and data integration

Table 6 summarizes the 19 interviews that were conducted. Interviews were each 45 minutes long. Interviews conducted at the beginning of the AI project followed pre-scripted and highly structured questions designed to understand initial expectations about AI and the project. Later in the research, interviews became less structured and focused on specific algorithmic features and attributes that emerged during the mapping of the algorithms across the user flows.

Table 6. Interviews

Contact	Interviews	Minutes
PMO Project Manager	3	135 minutes
PMO IT Manager	2	90 minutes
PMO IT Consultant	1	45 minutes
AI technology lead	5	225 minutes
AI data scientist	6	270 minutes
UX lead	2	90 minutes
Total	*19*	*855 minutes*

A defensible understanding of the algorithms escaped not only the researcher but was also difficult for the developers themselves to articulate. This underlined the value of the mapping exercise. The process of mapping the interactions across the user flow enabled a richer understanding of computation (Li and Jagadish 2014). It can bring the social context and algorithm together. An outcome of this exercise was a modelling of the ecosystem of users and algorithms (Figure 6.3). Circles represent users or agents, white squares represent frontend environments where the user interacts, and grey boxes represent computations and algorithms.

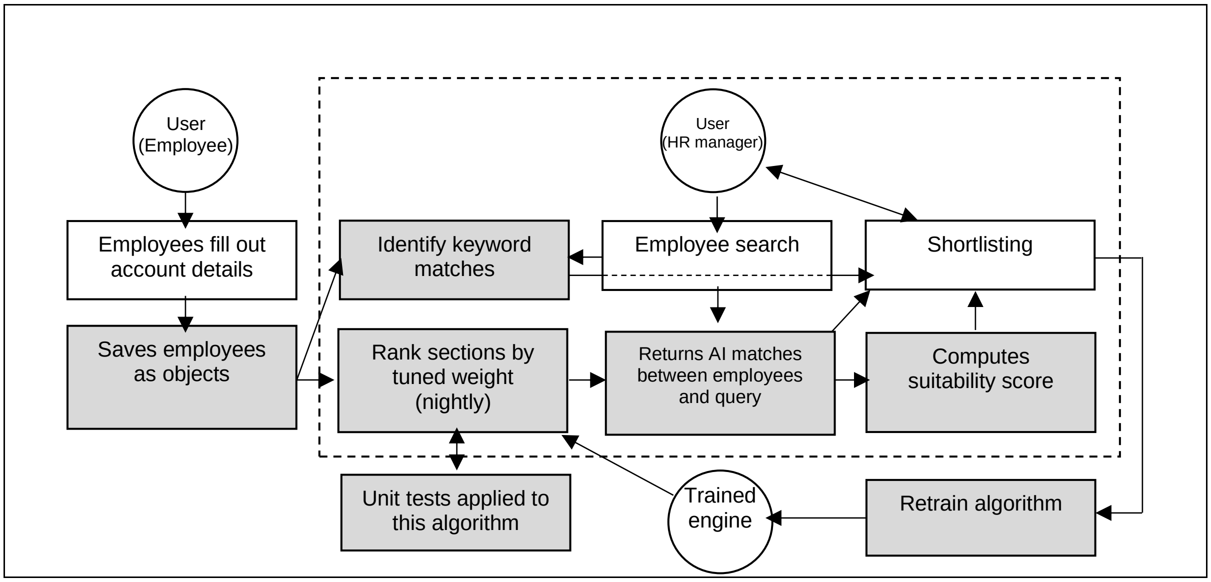

Figure 6.3. Mapping the algorithms across a user flow

As a note, the step 'identify keyword matches' is a separated logic. When a search is made by an HR manager it triggers a matching algorithm that returns employees. This then feeds into a suitability score and the retraining of the algorithm. Simultaneously, this query triggers a separate algorithm that returns specific keyword matches. The logic of this algorithm is not connected to the logic of the matching algorithm. This was added to enhance explainability, but in effect complicated the architecture with two different logics interacting with the users at the same time.

6.4.3 Data analysis

The thick descriptions (Geertz 2008) from the mapping of user flows through the triangulation of observations created a rich corpus of material. The focus of this study was the interdependence of users and algorithms. The grounded coder in the research team focused on letting the data speak for itself and became immersed in the user flows, team chats, and code description. Empirics and data became a primary source of discussion and further inquiry. We adopted the tools of Grounded Theory (Glaser and Strauss, 1973). In order to create robust explanations, we questioned each other's' interpretation of the data to move it beyond thick description (Strauss and Corbin 1998). The coding process guided the participant observer to return to the data and sources and recombine, aggregate or summarize key themes. To better understand the data all three researchers immersed themselves in the coding process. The coding process also involved an explicit use of process-tracing and pattern-matching (Wendt, 1999) which is beneficial for theory-development (George and Bennett, 2005).

The first stage of theory-building (Eisenhardt, 1989) began to establish and identify a mid-range theory, rather than identify explicitly deterministic causal and universally generalizable effects for the universe of larger cases (Gerring, 2004). The theory-building process involved the three researchers creating narratives to understand the ways in which algorithmic interpretability emerged. These narratives were initially parsed by the participant observer. The theory-builder also worked with the grounded coder in routine checks to ensure alignment between the emerging compelling theory and actual data.

We adopted systemized strategies (Miles and Huberman 1994) to develop our thematic coding. This began with a pre-defined code book shaped by initial research questions, but as the analysis continued, the code book changed and evolved. Overlapping themes were either combined or removed, with a goal of internal consistency and external divergence (Marshall and Rossman 2014; Mason 2017). We coded till we reached theoretical saturation (Gasson, 2011). The emerging themes that we wrote thematic memos for included algorithms being interpreted, algorithms becoming explainable, tuning, weighting, filtering, relative scoring, suitability scoring, and gut checking. Each of these ideas were fleshed out further during the course of coding across the data sources. The emergent theoretical understanding from this data was the process of how developers build their own interpretation, which is multiple and open-ended about the algorithm but how this process later merges with the process of users demanding a single explanation of why the algorithm produces any given results. We detail these processes in the next section.

6.5 Findings: developer interpretability to user explainability

The small AI team of OrgX had managed to secure a highly coveted UAE government project. The UAE government needed a system that could automate and improve the process of staff selection for its human resources (HR) staff. The adoption of AI and machine learning (ML) algorithms was demanded because the belief was that decisions about staff hires made by intelligent algorithms would lead to relevant and better hires. The OrgX project had been underway for a little over a year when questions began to emerge that baffled the developers of the algorithms. What is noticeable at this point of the narrative is that the algorithms discussed during the period of October-November 2018 were not ML algorithms, yet they were still seen as opaque and indeed offered unexpected results. At times the developers struggled to interpret the results offered by the algorithms they were building and testing. However, once the ML algorithm was initiated and users became involved understanding the workings of the ML algorithm became more complicated.

6.5.1 The process of developer interpretability

The developers of OrgX described algorithmic interpretability as “the capacity (offered by an algorithm that allows a developer) to understand a model, that is, given a certain output of the model, what determined that output”. The learning algorithm is written by developers, but the same developers found the results of the algorithms inscrutable because they led to unpredictable results. Comments such as “honestly, I don't know how it works” were common. In order to enhance their control over the algorithm and how it works developers designed small experiments. The aim of the experiments was to enhance developer understanding of the algorithms. The internal working of the ML algorithm was a black box to even the developers, so they tuned the algorithm to understand it inductively. Tuning was a process of changing only one variable of the input to the ML algorithm and then letting it run. Once the results were available developers would make note of any changes in the results from previous occasions where the algorithm had been run. Any difference in the results could then be assigned to the change in input.

> “*It is a way to codify and share knowledge. If I make a change just based on some reflections I make but then I do not provide tests for such knowledge to be consistent, my knowledge will go hidden and people in the future will not know if they are taking it into account or not. We could have better tests and methodologies, that is for sure, but they are not easy to develop. What began as tests are indeed becoming methodologies for us*”.

This process, it was hoped, would shed light on the inner workings of the ML algorithm. An example of a tuning experiment (below) shows not just the current tuning but also plans for additional experiments that could help enhance developer understanding:

> “*I am doing some work on tuning and demoing the matching algorithms .... Whilst I think this is a great baseline, ... we need to add a new ticket to improve, tune the algorithm, specifically:*
>
> *- perhaps include assessment mock data?*
>
> *- check on weight updates systems issue: for example, when a candidate discarded by the user is strictly better than a candidate selected by the user (i.e. has matching scores higher for each section) this forces all the weights to go to zero, because is the only mathematical solution. We need to address this*”.

Another approach that developers adopted to interpret the algorithm under design was to add filters to the algorithm. Filtering involved adding layers of categories that made certain criteria more important than others in yielding results when the algorithm was run. This entailed that certain criteria could be made more important thus tweaking how the input to an algorithm would purposefully modify results. Developers debated how certain criteria should be filtered into the important category. Such debates made the developers rethink how the very basis of their requirements for the ML algorithm were questionable. What should be made important through a filter was clearly key to the results but the truly important criteria would need to be built into the design of the algorithm (see Figure 6.4), as can be seen through the discussion amongst the developers:

```
return apply_filter(filter_type) if filter_type in search_filters else None
```

Figure 6.4. Examples of hard filters

> *"SeniorityLevel was part of the major (no advanced options) part ... what was the logic that made us shift towards putting it into the advanced options? I was thinking that SeniorityLevel is something we would probably want for any matching search, not just for the advanced ones".*

A by-product of this and other such discussions was the creation of what was eventually termed as hard and soft filters (see Figure 6.5). A hard filter was an extra layer in the algorithm that was purposed with verifying that the results met all mandatory conditions and ranges. An example of a hard filter for the HR system under design was years of experience of each possible candidate.

```
prior_knowledge = {
        'Achievements.ProjectTitleAndEvent': 40,
        'Achievements.Description': 30,
        'Achievements.Role': 20,
        'Education.FieldOfStudy': 70,
        'Education.Organization': 10,
        'Education.Country': 5,
        'Education.Title': 20,
        'Education.Degree': 20,
        'Memberships.Organization': 2,
        'Memberships.Role': 3,
        'Trainings.Organization': 10,
        'Trainings.Title': 20,
        'WorkExperience.Organization': 15,
        'WorkExperience.OrganizationSector': 2,
        'WorkExperience.Country': 5,
        'WorkExperience.JobTitle': 70,
        'WorkExperience.Industry': 50,
    }

    prior_importance = 0.3


    weights_from_historical_searches = [
        {k: search['weights'][k]
            if k in search['weights'] else prior_knowledge[k]


    return {key: prior_knowledge[key]
            * prior_importance
            + (reduce(lambda x, y: x + y[key], weights_from_historical_searches, 
0)
            / len(weights_from_historical_searches))
            * (1 - prior_importance)
    if len(weights_from_historical_searches) else prior_knowledge[key] for key 
in prior_knowledge}
```

Figure 6.5. Examples of soft filters (machine learning weights)

The soft filters, on the other hand, were aptly referred to as the 'smart' filters by the designers. The soft filters involved the ML algorithm taking the search text added by the user and offering unique and interesting options of candidates for the job. The developers explained how filters that were designed into the algorithm made the algorithm stronger, yet this reduced the ability of the search and results offered by the same algorithm. In essence, if the algorithm could be varied through different search parameters the results were more nuanced and closer to the actual needs of the user. Pre-designed filters in the algorithm would restrict any alternative and reduce the user option to customise the search to her specific needs. The user experience sessions showed that the search parameters were sometimes long and repetitive while other users offered short words only (not sentences) (see Figure 6.6 for an example of a long search). Crucially, building hard filters into the algorithm reduced the power of soft filters. The fact that the ML algorithm had multiple layers of changeability added to the capacity it showed to offer effective results. Simultaneously, it allowed the developers to better interpret the inner workings of the algorithm. If the algorithm was hard coded with any filter it made the results less possible to disentangle from the multiple stimuli in any context of use.

> *"The whole mandatory/preferential system is a bit weird for the algorithm. Because it they are mandatory they become filters, if they are preferential they become items of the algorithms (thus they gain their own weight and matching score). From an algorithm perspective the more search parameters we have the better, thus if many of these "preferences" (such as experience level) are missing because they are hidden away into the advanced options (where my expectation is that it would be mostly used by User that have mandatory requirements) the search is weaker.*
>
> *Also, we need to agree on what we do for the other side of the problem – are we grouping their sub-matching scores in categories that we show (e.g. grouping all of them in ProfileWorkExperience? Unfortunately, this would make the algorithm weaker as it would reduce the variance of the weighting".*

Enabler of people - inspires, encourages, and motivates others; reinforces human capabilities and talents through empowerment; effectively leverages others' capabilities and demonstrates emotional intelligence. Role model - shows values of integrity, humility and respect; embraces and promotes the concepts of happiness and positivity; makes substantial contributions in representing the country in a positive way. Open to the world - open-minded to different experiences; embraces the values of peace, tolerance and coexistence; enjoys an extensive network of relations and is well-versed in global culture. Futuristic - well-informed about global trends; able to imagine the future; anticipate and analyze opportunities through developing future scenarios and proactive plans. Innovative and disruptive - Catalyst for change at the individual and institutional level; entrepreneurial, risk taker, and adventurous for whom nothing is impossible. Agile and fast - creates an environment which promotes and empowers change, achieving goals in the quickest possible way and makes efficient use of available resources with self-assurance in different situations. Smart, effective and efficient decision maker - adopts a critical, analytical style of thinking, is mindful and gutsy of all decision parameters in

achieving the most desirable outcome. Focuses on the government's ultimate goals and achievements - strong advocate in achieving the government's objectives; adds value in all aspects of work performance related to national goals. Life-long learner - seeks self-development in order to acquire and enhance diverse skills to meet future needs; passionate for knowledge, research and exploration. Well-adversed in advanced technology - awareness of new technologies and trends such as the fourth industrial revolution and artificial intelligence (AI) and how to get the most benefits out of these technologies which will transform the way we live and work in the future to achieve people's happiness.

Figure 6.6. An example of a search query by a user

Along with tuning and filtering developers used another approach to grasp the workings of the algorithm. This approach was *matching*. Developers were keen to be able to score individual candidates that the HR related ML algorithm searched for independently of any search terms. This required the algorithm to attach a suitability score to each talent of a candidate. Such a mechanism was useful because the developers were then able to match the independent score given to each candidate to the score offered by the algorithm through an actual search and make sense of '*how close*' the candidate was to any ideal desired by a user. The discussion amongst the developers related to their ability to define a '*successful candidate*':

> "*How to define "successful candidate", is it "got the job?" and "role model?"*"

Matching involved attaching a relevant and useful score for each talent held by a potential candidate and then using the scores to compare different talents (and their configurations) across candidates. An actual example can be seen in Figure 6.7 where the ML algorithm has created scores for different talents. These scores make little sense to the user, as we will see later in the narrative, but they are an important first step for the developers of the algorithm.

> "*http://127.0.0.1:8000/algorithms/talents/match should give you profile['MatchingSentenceWeight'] & profile['MatchingSentence']. This needs to be further tuned but would need your feedback on stuff like what should be the maximum acceptable length of the sentence*".

Developers even go so far as to suggest that if the algorithm is not understandable to them through different processes then perhaps dropping its use is the only option. If they cannot understand the algorithm, they cannot tune it:

> "*If it cannot be tuned we may want to reconsider using it entirely*".


```
"MatchingScores": {
            "ProfileEducation": 0.06708774125319938,
            "Organizations": 0.056467308549537965,
            "suitabilityScore": 0.7733833435156914,
            "ProfileWorkExperience": 0.8969138944135691,
```

```
            "ProfileAchievement": 1.0,
            "Industry": 0.0830183903086915,
            "ProfileTraining": 1.0,
            "YearsOfExperience": 0.7238794746946816
```


Figure 6.7. Excerpt of a machine learning algorithm showing 'match strength'

Again, such an approach allowed developers to refine their own understanding of how the algorithm functioned in real world settings. Tuning, filtering and matching were all different approaches embraced by developers to parse and interpret the ML algorithm that continued to elude them. It was of little help that they were the designers of the algorithm. The opacity of the algorithm became more prominent when the users, through user experience sessions, began to ask probing questions. User questions could not be given a response where there was any ambiguity. They wanted a clear and decisive answer. Whereas developers had been content with different interpretations of the algorithm, users demanded clear, and single answers for how and why the algorithm offered any solution. This led to the need for explainability rather than interpretability:

> *"we need Explainable AI to show why the system picked a certain profile"*

Table 7. Analytical terms in findings

Analytical Term	Definition
Tuning	The process of opening the black box of the inner workings of an algorithm through experiments involving the changing of one input at a time in order to trace the change through to the solution.
Filtering	Filtering involved adding layers of categories and/or weights that made certain criteria more important than others in yielding results when the algorithm was run.
Matching	Matching involved attaching a relevant and useful score for each talent held by a potential candidate and then using the scores to compare different talents (and their configurations) across candidates.
Gut checking	The use of unsophisticated yet logical measures to gauge and test the results of the ML algorithm. This seems analogous with 'trial and error' but underscores that the optimal answer is still judged by a 'feeling.'
Pruning keywords	Removing and/or amending the filters designed into the ML algorithm in order to provide the user with more freedom to search with short phrases rather than lengthy description statements.
Relative scoring	The adjusting of weights of the frontend filters to fine-tune the ML algorithm's search process such that the algorithm could then show the strength of each talent of any candidate relative to other talents.
Suitability scoring	Gauging the implementation of the ML algorithm enough to be able to create artificial bare minimum scores of suitability for all talents so that the user can have a threshold to work with, and even amend through relative scoring if needed.

6.5.2 The process of user explainability

Once the user began to be involved in the process of testing the prototype version of the software interpretability shifted into a language of explainability. What was noticeable though was how this was not simply a cosmetic change of language. Whereas interpretability involved multiple and different interpretations by the designers, explainability was a process of creating a single, plausible explanation for how and why the algorithm offered any given result for the users. The process of explainability involved users so this process meant that the developers were refining their own interpretability while specifying definitive answers to the 'why' question for users. Here we see the development of ML algorithms where a number of different 'gut checks' were carried out by the developers to offer transparency to the users. This transparency was questionable at times because often the developers themselves were forced to reverse engineer the reason for the algorithm's 'solutions' through educated theoretical guesses. Different forms of 'gut checks', as they called them, involved pruning keywords, relative scoring, and suitability scoring.

Involving the users through user experience sessions was seen as successful by the developers because the developers received valuable feedback from their customers and at the same time the developers created customer emotional buy-in for the system:

> *"Thanks User Experience Lead (UX) for the additional review! The GLP Project Manager has been a big fan of our UX sessions, it brings her greater confidence in the tool. All this iterative work on the tool is really a nice experience for them. Thanks for championing an important culture of engagement and UX".*

During the user experience sessions, it became evident that along with developers continuing to tune the ML algorithm other changes would also be needed. The developers began to adopt methods of *gut checks*. Gut checks were the use of unsophisticated yet logical measures to gauge and test the results of the ML algorithm.

> *"1) Try using synonyms for the purpose of matching but with a lower weight. Like you are doing. 2) Don't return synonyms in the matching reasons but in a separate object of the response (something like matchingReasonsSynonyms). 3) Don't affect matching sentence for now. 4) See how this affects our tests cases".*

Noticeable from the developer's instructions a new practice was emerging of coupling tuning experiments along with a quick gut check. The weights for different talents were changed but the new results that were linked to this change were located under a separate term (i.e. matchingReasonsSynonyms) so as to not confuse the result with any of the other experiments being tuned. Users were different from the developers and had distinct needs. While they were not programmers and could not understand the code, they were however, knowledgeable about their own domain – HR. Machine learning algorithms clearly need to be understood by the developers building them but developers also need to balance a desire for understanding with the pragmatics of getting the job done. The users, however, needed and demanded clarity.

One manner of offering the user an explanation of why the ML algorithm produced a result was achieved by pruning keywords to sharpen the input. This was a specific pre-processing exercise. This made the results somewhat traceable. *Pruning keywords* involved removing and/or amending the filters designed into the ML algorithm in order to provide the user with more freedom to search with short phrases rather than lengthy description statements.

> *"We can probably prune it of weird words given it's a backend search, the user will only see 'matched keywords'*
>
> *- Training and certifications very important but not yet tuned/optimized*
>
> *- the question "What leader are you looking for" didn't make sense to him. Said we may need a better description*
>
> *- Tested both a job description version (that was their first instinct when looking at description), and then tested a shorter keywords based. He preferred the latter, and suggest we even say, "Key words" instead of "description"*
>
> *I am exploring ways to remove match_keywords that may not make much sense. For example, if I am looking for a 'doctor for ministry of health', there can be an irrelevant profile that have the word 'ministry' in it.*
>
> *I recall "Etc" was one of the guilty words".*

It was evident to the developers that pruning the frontend filters was relevant for users as this was the part of the program that users would interface with and understand. In this example the word 'ministry' proves to be unnecessary as it confuses the ML algorithm. The algorithm focuses on the word ministry, but this is the least important keyword input by users. The developers work to lower the weighting of this word so that the other words like doctor and health become a more prominent part of the search. The users were introduced to new filter and shown that they could remove such keywords in their search.

Developers were also keen to adjust the weights offered to each category of the filter. Doing so would offer more search manoeuvrability to the user and could reduce the need of explainability for pre-defined filtering. However, during the user experience sessions the developers noticed that influential users gave mixed approval on the type of scoring they wanted.

> *"It seems they [users] yet again confirmed that the range is strange (-5 to 10, with us 'ignoring negative numbers' and treating all positive numbers as StenScores). I agree with your initial worry that this just does not make sense, but as I shared with you after our call with (Legacy Developer), a) She doesn't really understand the data, b) we still don't really have clarity now. They want us to use "score", not StenScore. So, we need to clarify which score, how it's calculated, etc., BEFORE we waste time developing on the front end. If we 'assume' they will send a StenScore, which you suggested, then we could get in trouble if, as her email confirms, it turns out the score we are supposed to use is more complicated than that".*

Not only was approval mixed but the feedback offered by the users was confusing as well. This reflected the slender grasp the user had over the ML algorithm. Developers struggled with their own interpretation of the algorithm but bringing in the users so early into the project created a new level of confusion. The eventual decision of the developers was to seek relative scoring. Relative scoring was the adjusting of weights of the frontend filters to fine-tune the ML algorithm's search process such that the algorithm could then show the strength of each talent of any candidate relative to other talents.

> *"I still think its important to at least be able to copy what the database query will return: numbers from -5 to 10, with distribution seemingly clustered around 0, and with very very few scores above 5. Can we write a short function to simulate this? That way, no matter if they made a strange mathematical error in their logic, at least our front-end will be designed for their actual 'scores', not our 'assumptions' about what the scores should look like".*

Developers clearly had a different view on how the algorithm should perform, and this did not resemble what the users expected. Such differences forced the developers to interrogate the design and working of the ML algorithm yet more – explainability was proving harder than developer interpretability. To appease the user however, developers created workarounds.

The development of the ML algorithm sometimes led to failure in the search running as planned. In such situations developers had to become vigilant and carry out further 'gut checks'. The ML algorithm was a bit of a mystery to developers because they could not understand why the algorithm was not giving more weight to results that had multiple confirmed returns.

> *"I noticed that some of our ml tests for the matching algorithm are not passed anymore in my local version .. could it be the machine learning? What else have we changed? Could you take a quick look to if some sort of bug was introduced?"*

The developers attempted to remedy this through adding custom weighting to matches. Custom weights were a hardcoded set of weights added to the filter. The developers understood that human intervention was necessary when working with the ML algorithm. They then ran multiple tests to assess the suitability score of each talent. Suitability scoring was the gauging of the implementation of the ML algorithm enough to be able to create artificial bare minimum scores of suitability for all talents so that the user could have a threshold to work with, and even amend through relative scoring if needed.

> *"Had a look into it. There is only one profile search that fails. For the test that fails, our test description is "Food Security Undersecretary, Candidate needs to have international experience, network and knowledge of international best practices in food security and food technology. Previous experience in food safety research. Possibly doctorate in food science. Public health expert" So clearly we are expecting the algorithm to know that 'food' is the most important keyword here. however, if you look from the perspective of algorithm, tfidf value of 'best' is higher than that of 'food'. The guy with profile ID that we were expecting in this search is a true expert in food technology & should have appeared in the result but sadly 'food' does not have a high weight".*

Suitability scoring was beneficial because the candidates the developers expected to see emerge in the results of a search began to appear as needed:

> *"this test passes if you change that number to minimum suitability score but I guess we were expecting that people we cherry picked to be fit for a job should have suitability score higher than the bare minimum".*

The interesting realization of the developers was how writing algorithms, especially ML algorithms was a creative process that did not always make sense. The larger issues of algorithmic interpretability and explainability had been only slightly resolved. Indeed, the process of trying to achieve understanding of how the ML algorithm worked had forced the developers to accept that the algorithm was more elusive than ever.

> *"Building an algorithm is as much engineering or art as it is a science, it takes some tinkering, creativity, and making decisions about the model and data. Sometimes those decisions seem difficult to make, and we're only guided by these tests and 'tuning' to expected results, but that itself is not a bad thing. That's the nature of the work, and we could be getting pretty good as a team at building the right algorithms, tests, and user interactions to find the right balance out of all of this".*

6.6 Theory development

Machine learning algorithms, while they offer developers the opportunity to be creative, also pose issues of transparency and understanding to them. The developers in our narrative reveal how the process of building a system that would help ease the decision-making abilities of users forced the developers and users of the system alike to question how the machine learning algorithms unfolded in use. It is expected that users ask questions about the inner workings of a new system, but the machine learning algorithms also eluded the very developers that created them. We map the journey of developers that were building machine learning algorithms. These developers struggled to understand how the machine learning algorithms functioned, and why they offered any given output. In effect, the developers were striving for algorithmic transparency (see Table 8). We define *algorithmic transparency* to be how readily an algorithm reveals how it operates, including the criteria it depends on when it generates an output, why it operates in the manner that it does, and the reason and rationale for the relationship structures that emerge. It is about the possibility of tracing the workings of an algorithm from initialization to the generation of an output.

Given the developer's motivation and need for algorithmic transparency that emerged as the system under construction took more sophisticated shape the developers designed ways to interrogate the working of the machine learning algorithms. This led to the process of developer interpretability.

Table 8. Key concepts

Concept	Definition
Process of developer interpretability	The process of searching for algorithmic interpretability by developers to interpret a machine learning model, whereby offering multiple possible theoretical explanations for the results yielded by the algorithm.
Process of user explainability	The process of developers manufacturing a single explanation for how and *why* the ML algorithm produces any given result to satisfy the user.
Algorithmic transparency	Algorithmic transparency refers to how readily the algorithm reveals how it operates, including the criteria it depends on when it generates an output. It is about the possibility to trace the workings of an algorithm from initialization to the generation of an output.
Algorithmic opacity	Algorithmic opacity refers to the condition where the criteria used by an algorithm to generate any output are not visible and/or understandable by the user (and often even the creator of the algorithm).

6.6.1 Process of developer interpretability

Developer interpretability is the process of searching for algorithmic interpretability by developers to interpret a machine learning model, whereby offering multiple possible theoretical explanations for the results yielded by the algorithm. Developers of machine learning algorithms designed three different methods to unpack how and why the machine learning algorithm suggested any given results. These methods included tuning, filtering and matching. While tuning was a broad brush set of experiments that were constantly run, tweaked, and then re-run, filtering and matching were more localized methods of assessing the possibilities of the learning algorithm.

The process of developer interpretability reveals that the designers of learning algorithms build them with clear requirements and usual methods of software development, yet the nature of learning algorithms makes them elusive. The learning algorithm is not built to work to a target and instead is programmed to learn to work better on its own. The *learning to learn* is undoubtedly by design but the constant tuning, filtering and matching conducted by developers indicates how progressive the learning algorithm becomes in what it can achieve. Designers of learning algorithms appreciate the results offered by the algorithm but are uneasy about the opacity with which the results emerge.

Tuning offers real world experiment conditions where developers can peel away one criterion at a time to gauge which one in particular changed the nature of the algorithm running. The developers offer input to the algorithm and then look to the results. Doing these steps over and over again while changing the input ever so slightly allows developers the possibility to understand how the algorithm works, and why it offers different results. Tuning alone cannot provide the necessary insight that developers seek. Filtering is another method that is adopted to navigate the workings of the algorithm in a nuanced manner. Filters entail the use of weights given to different criteria being sought by the algorithm. A change in the weight of any criteria should and often does affect the results given by the algorithm. Using the change in results gives the developers the ability to narrow in on the subtle learning changes occurring in the algorithm as it repeats its work over time. Filters can be added at different levels of the algorithm making the work of algorithms more subtle and useful as well as offering better results.

Finally, the process of user interpretability adopts a matching technique to fine-tune the algorithm and better evaluate how the algorithm functions. Crafting what can be arbitrary scores for different categories being sought by the algorithm allows the developers to evaluate the results offered by the algorithm in relation to other results. Like filters, matching can be further fine-tuned to build scores for sub-parts of any search criteria thus allowing a relative form of evaluation of different criteria of the same candidate or product.

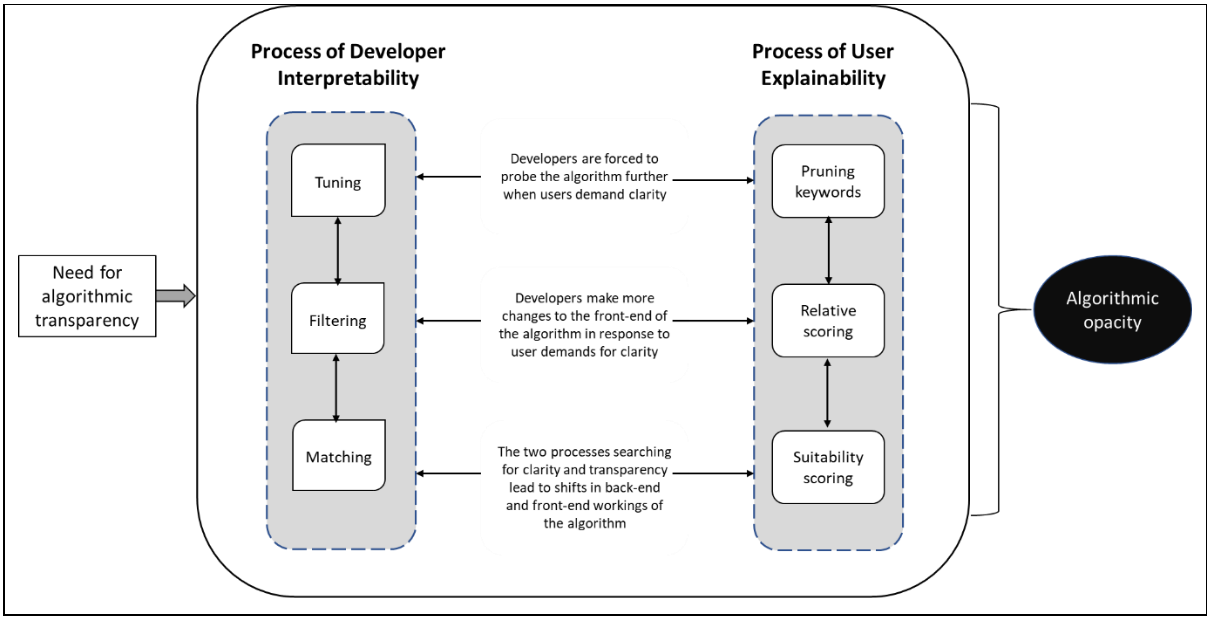

Figure 6.8. Need for algorithmic transparency leads to increased algorithmic opacity

These three techniques used together can help developers to better, though not fully, understand the learning algorithm they build (see Figure 6.8). What is noticeable from the process of interpretability is that developers are attempting to interpret the results and gauge the algorithm. We do not see the developers strive for one definitive theory of explanation, but instead they collectively offer different intelligent interpretations of why and how the algorithms works the way it does. This changes substantially once users of learning algorithms are included in the design or testing of the system. We then see a shift to user explainability.

6.6.2 Process of user explainability

We define the process of explainability as the process of developers manufacturing a single explanation for how and why the ML algorithm produces any given result to satisfy the user. This process reflects how the developers, who are in the course of building their own set of multiple, even competing theories to explain why the algorithm behaves the way it does, now must agree on the most logical explanation to give to the users. Organizational users of learning algorithms demand explanations from the developers of the system about how the results are achieved. This is a natural desire on the part of users because in their everyday use of the system they make small adjustments to the search query. This makes them eager to understand how any change they make in the search will affect the algorithm's results, and in turn their decision-making.

The process of user explainability depends on different techniques that users are offered by the developers and the algorithm to make small yet meaningful changes to the search query. The techniques include pruning keywords, relative scoring and suitability scoring. All three of the techniques are more grounded and pragmatic approaches to make adjustments to the front-end of learning algorithms. While the process of developer interpretability involved deep level changes to the back-end of the learning algorithm, the process of user explainability involves small shifts to the front-end only. This is a relevant detail because it shows how learning algorithms in a context of work need to be adapted by users of different levels of expertise. Most users are not developers and any change to learning algorithms can have unpredictable results, so developers begin to narrow the space of change possible by users. This narrowing ensures that only minor tweaks can be made to the search query, and related areas.

The three techniques of pruning keywords, relative scoring and suitability scoring together facilitate the user's engagement with the learning algorithm. The user can adopt any or all of these techniques to gain control over the outcome of the learning algorithm. Indirectly this process allows the user to understand how the algorithm works. Our study shows that the users experimented with the learning algorithm but returned to the developers for explanations of how and why the algorithm functioned. Clearly the three techniques offered to the users did not suffice to offer a clear explanation to them. The developers needed to step in to explain the work of the algorithm.

6.6.3 Algorithmic opacity

Our study shows that the process of developer interpretability unfolded alongside the process of user explainability. Together these two processes, though attempts to make better sense of the algorithm, led to greater algorithmic opacity rather than transparency. What is interesting is that the process of developer interpretability meant that developers, though struggling with deep understanding of the learning algorithm, were still able to offer different relevant theories of explanation. These explanations were in fact diverse interpretations made by expert developers who were able to handle multiple, even competing theories on the same algorithm without it affecting their work. It seems that the creators of learning algorithms take a more pragmatic approach to how such algorithms work and can balance their desire for understanding with the need to continue their work. Users, on the other hand, require clarity where single explanations are expected. Doubt or competing theories made the users less able to trust the system. Their belief being that if the experts who built the algorithm don't understand it then perhaps the algorithm isn't useful or built well.

The processes working together made the developers return to a position of questioning how the algorithm functioned. The slender understanding the developers had built of why the algorithm functioned the way it did was probed harder by the users thus making the developers feel a return to a mood of unease due to a lack of clarity. Further, the developers made it possible for more front-end changes to be made by users. This step was a simplistic manner of seeking understanding. What it entailed was tweaks to the search query so that the query could handle short phrases rather than lengthy descriptions. Such changes, though well-meaning began to remove the ability to interpret the algorithm further way from the actual working of the algorithm or how the algorithm learned. The developers had worked to use different methods to make the algorithm transparent and explicable but combining user needs along with developer expertise created countering tensions. The eventual outcome of both processes that had individually aspired for algorithmic transparency created algorithmic opacity instead. Algorithmic opacity refers to the condition where the criteria used by an algorithm to generate any output are not visible and/or understandable by the user (and often even the creator of the algorithm).

Learning algorithms are different. Developers and users struggle to make sense of why they operate in the manner that they do. Algorithmic transparency with learning algorithms will need time, and perhaps it will be other learning algorithms that will shed light on how they work.

6.7 Discussion and implications

Our study contributes a theoretical understanding of why learning algorithms in development and use prove difficult to parse. Making learning algorithms explainable is difficult because not only do they continue to change and adapt on their own, but in use the context, users and new relationships that are built into the algorithm make them yet more complex. Our study shows that the elusive nature of learning algorithms is very real and that the processes adopted to make them transparent push explainability further out of reach. This could imply that the processes chosen to gauge the algorithm are wrong and need to be changed, yet it is evident from this case that the methods and processes are both intuitively correct as well as sound software practice. Our theorization of algorithmic explainability shows how different processes work with and against each other to strive for transparency. Transparency, because the algorithm is constantly adapting and learning, is in effect unachievable.

Like all research this study has its limitations. Our case offers rich data and we ensured access to multiple source of data within the case study, but our analysis and contribution are based on a single case study. This makes our work analytically but not empirically generalizable (Lee and Baskerville 2003). Future research could look to a multi-case study to gauge how learning algorithms play out in multiple contexts and how that affects or improves the transparency that can be achieved. It would be equally interesting to conduct a longitudinal study of learning algorithms that spans a solid length of time (5-10 years of use) where both the code and its users and developers could be mapped to learn how algorithms evolve. There is hope that the future holds better learning algorithms as well as deeper understanding of how and why they work.

Our study has implications for scholarship on explainable algorithms and algorithmic decision-making.

6.7.1 Explainable algorithms

Explainability of algorithms (Arrieta et al. 2019) is an area of research that is related to algorithmic opacity (Kellogg et al., 2020) and algorithmic accountability (Diakopoulos, 2015). There is a growing need to be able to explain (understand and trace the workings of an algorithm) algorithms we use. This is based on our increased dependence on algorithms in nearly every aspect of our lives (Olhede and Wolfe, 2018). It is then somewhat worrying to study developers that build advanced and learning algorithms and find that those that build them are often confused by their own creations. Clearly, algorithmic opacity is not always intentional (Burrell, 2016) yet it still has implications.

Algorithms can be opaque because the user of them may not be experts in building or unpacking their inner workings (Paudyai and Wong, 2018). However, our study of expert software developers shows that machine learning involves computations and a capacity to learn that is not traceable enough to make it explainable. Even scholarship that acknowledges that machine learning is different does so on the basis of arguments such as scale and complexity (Burrell 2016). These are relevant issues, but this work shows that even in smaller scale projects the developers of learning algorithms can be mystified by their own creations. Scholarship has also shown that learning algorithms are complex because such algorithms are composed of different components that are built by different developers and organizations (Burrell 2016; Sandvig et al. 2014). Our work complements this because we show that even when developers build most if not all of the components themselves in the same organization the learning algorithm can remain opaque and unexplainable.

The social context of algorithmic adoption is also important (Matzner, 2019). There are numerous studies that show how algorithms affect work – largely adversely – in settings such as platform managed work. The work on the gig economy provides rich studies of Uber drivers being managed by recommendation algorithms (Möhlmann and Henfridsson, 2019) where information asymmetry between the algorithm and the drivers means the latter feel disempowered (Rosenblat and Stark, 2016) and dehumanized (Möhlmann and Henfridsson 2019). This body of work is rich in its perspective of the user working with algorithms but there are few studies which rely on the developers of learning algorithms and their context of working with them. Our study complements prior research because it focuses on developers as well as users of learning algorithms. Our work shows that each modification and tuning to the learning algorithms was done to grasp the meaning of how the algorithm functioned and why. This drive for explainability unfolded in a social setting where the combination of code and developers seeking their own understanding as well as offering 'solution' replies to their client users pushed the explanation further from the grasp of everyone involved. If the learning algorithm was inexplicable to the developers, it only became more so once the users began to ask questions.

6.7.2 Algorithmic decision-making

The drive for algorithmic explainability arises because non-developers using algorithms have increased. Algorithmic decision-making (Lindebaum et al., 2020) takes centre stage with predictive policing (Bennett Moses and Chan, 2018), health (Ahsen et al., 2019; Holzinger et al., 2017), platform work, high-frequency trading (MacKenzie, 2018), to name just a few. The invisibility and non-transparent manner of how decisions are made by algorithms is troubling in a world where we want justice, ethics and morality to be embedded in our everyday systems (Sandvig et al., 2016; Winfield and Jirotka, 2018). However, it is equally interesting to see the rise of learning algorithms and AI in areas such as healthcare being touted as a step forward from human experts. Recent comparative studies of breast cancer detection prove that algorithms are better than doctors at detecting cancer (McKinney et al., 2020). This may well be a step forward for humans but scholars caution against machines and algorithms that embed machinic rationality taking decisions (Lindebaum et al. 2020) without human oversight. Such decisions are not based on human sympathy or morality.

Studies on algorithmic decision-making make a distinction between discrimination and non-transparent decisions (Lepri et al., 2018). Non-transparent decisions are not necessarily opaque by design but studies on Uber (Rosenblat and Stark, 2016), Airbnb (Cui et al., 2020) and other gig economy companies have shown that discrimination is often designed into the algorithm on purpose (Sandvig et al. 2014). It is easier to perceive the algorithm as objective because it is technology. Our study of learning algorithms built to decide the best candidate for a given job shows how such discrimination can be both designed into the filters of the algorithm as well as negotiated by the user through a modification of a search query. Learning algorithms are mutable and amenable to change in the hands of the developers and users.

Using learning algorithms in organizations is beginning to shift job roles taking over decision-making responsibilities (Agrawal et al., 2019; Furman and Seamans, 2018). More decision-making jobs will be taken over by learning algorithms in the future, and while this can increase organizational efficiency, response time and overall performance, it can also decrease decision-making transparency and accountability (McAfee and Brynjolfsson 2016). Our study shows that though all the people involved with the learning algorithm were at times frustrated with how it worked and why it gave certain results they persevered with its use. There is a desire for efficiency that technology meets quite well but an unquestioning adoption of learning algorithms can trade off decision-making transparency and accountability for efficiency (Lindebaum et al. 2020).

6.8 Conclusion

This study examines the development and use of machine learning algorithms by an organization. Taking a process view we establish a grounded theorization of how algorithmic opacity becomes inevitable because of the structure and design of learning algorithms as well as their relational existence in human-centric contexts. We see two processes emerge that involve developers and then user who both strive towards algorithmic transparency. Our study contributes to extant work on explainable algorithms and decision-making algorithms. The processes reveal the different ways that developers and users employ to clarify the how and why learning algorithms work the way they do. It is somewhat paradoxical that the very measures taken by both stakeholders increase algorithmic opacity rather than transparency.

Chapter 7 – Negotiating complexity across the development of algorithmic and non-algorithmic personalization

7.1 Chapter preface

This chapter is based on a paper written by the thesis author. At the time of this writing, this work has been submitted to an Information Systems journal and may undergo changes to its current form before publication. Nonetheless, this work in its current form played an important role in answering the research questions set out in this thesis.

This work follows chapter 6 findings of pervasive inscrutability in emerging ICT-mediated personalization, which concluded with questions about generalizability. Within one case of algorithms, an HR platform, user desires for explainability or comfort in understanding why the platform made the decisions it did led to new layers and abstractions. Attempts to explain functionality were replaced with interpretations of its functionality. Even further, the ability to interpret how some features work proved challenged. When are algorithms easy to explain? When are they less explainable? When does inexplainability become interpretation, and when is an algorithm's function beyond interpretation? These questions inspired this chapter.

The first two papers explored a specific class of algorithms, built for HR managers. The AI company offered an opportunity to expand the scope of analysis. The company continued to build algorithms for other partners over the course of HR project. The company worked with a central bank to build an algorithm that defines sensitivity of documents, worked with a mid-sized insurance company to support underwriters in their assessment of risk, and worked with universities and industry to build a text analytics research platform. Interactive mapping methods proved fruitful for the HR project, so with questions of explainability in mind, the research was extended to include this mapping across three new projects. Carefully mapping each function, if a feature was classified as personalization it was kept aside for deeper analysis. In total, 34 personalization features were found across the four projects. Some were traditional applications of ICT-mediated personalization, like adaptable interfaces, while others utilized emerging ICTs like natural language processing and machine learning. This therefore became a holistic review of features across projects and over time. This was a good setting from which to ask questions about deeper generalizability. A framework of questions was built around perceptions of design complexity, shaped by notions of affordance and constraint. Stable patterns soon began to emerge. Complexity was operationalized into two dimensions: how difficult it is to build a set of features, versus how difficult it is to explain a set of features. Emerging ICTs often classified as AI, especially those utilizing social data, proved to consistently be hard to explain. But they were not always hard to set up. Some 'quick wins' could be found from the application of AI, but they introduce more inexplainability than more challenging applications of personalization that may not have used AI. Designers engage with them in different ways compared to more traditional ICTs, in part due to their occasional use of inscrutable logic, or their lack of explainability.

In the first paper pursuits of personalization were found to lead to the adoption of emerging ICTs thanks to new capabilities. In the second paper pervasive inscrutability with some of these systems was observed. In this third paper, the different ways traditional versus emerging ICTs are negotiated with are explored. However, across 34 cases issues of bias amplification have remained persistent as well. An important contribution of this thesis is not only advancing personalization research by unpacking the relationship between designers and these emerging ICTs, but also the raising of an alarm. Through the design process there are many moments where choices are made, human choices, that have significant implications for the overall process. For example, in this paper an early experimental algorithm was designed to create 'topics' out of a government official's social media accounts. A natural language processing algorithm was designed that would process Tweets into computer-readable formats, and would then 'cluster' similar words together based on how likely words are to be next to each other in a larger universe of English corpus/text. This is a highly risky experiment. The discourse of politicians was treated as if it was any other text, ignoring implications of power. Tweets serve more than just passive dialogue, but further political agendas (Bouvier and Machin, 2018; Farhall et al., 2019). Further inflaming this danger, the state of natural language processing of sarcasm and idioms is much more nascent than words where the meaning is direct. Expressions where the meaning is hidden, where some words say one thing but mean another, or where the polarity of emotional sentiment can flip to the opposite of what the words convey directly, require separate and still evolving approaches (Bagate and Suguna, 2020; Katz et al., 2004). Even though it was a highly sensitive context, there was no sign of awareness of these concerns from the developers.

If across 34 cases of personalization issues of inscrutability and biased decisions made by designers prove persistent, especially as emerging ICTs involving natural language processing and machine learning are adopted, extended commentary and investigation into developer bias is called for. Are developers appropriately positioned to make judgements about what algorithm to use? They often make assumptions on behalf of the end-user, should the end-user have more sway? Then again, there are complex ICT challenges that end-users cannot provide guidance for. These papers also reveal that cases involving a team of multiple developers means bringing competing and different interpretations and algorithmic familiarity. This can both mitigate overall bias, by allowing iterations to learn from diverse views, but can also mean biased decisions can be reinforced by group decision-making and this can have serious implications for the final product. How can we track blind faith? The first three papers do not explore if the personalized systems ultimately built are free of bias. The purpose was instead to understand relationships. Through a detailed mapping of the algorithms and their design, including over time, the persistence of these problems was nonetheless revealed. If anything, issues of blind bias are amplified because once an algorithm is set up and running autonomously, there is a certain amount of blind faith in whether or not it is optimized or mitigating bias. The algorithm can then reinforce bias autonomously. Because the adoption of inscrutable ICTs is on the rise, there is a growing call for the decisions made about their design to be accessible and made transparent. Research approaches such as the interactive mapping employed across these three papers gives some promise. We can go back in time and look back at the design decisions employed. Better still, questions of design complexity along explainability and interpretability could become an active frame during the design stage allowing for a tracking of higher-risk decisions to be captured at the moment of their design. This could shape future research: techniques for capturing risky decisions and strategies for mitigating them.

7.2 Introduction

This research explores the use of machine learning (ML) and natural language processing (NLP) in mediating personalization, the process of tailoring products and services to match individual preferences (Bragge et al., 2007; Montgomery and Smith, 2009; Riemer and Totz, 2003). These approaches are compared to more traditional applications of information communication technology (ICT)-mediated personalization. The use of AI to mediate personalization introduces dynamics that can change the nature of ICT-mediated personalization, by facilitating the development of a new type of agent that can learn how to mediate value for diverse users directly without the need for a human manager's mediation. Beyond being another tool, the findings in this paper shows that AI-mediated personalization creates webs of value whereby human and machine agency come together. The findings in this paper reveal that the process of configuring these systems involves complex social and material negotiations between designers and the black boxes they are increasingly engaging with.

Technology has played a critical role in the evolution of personalization. Mass production ushered in an era of machine-based standardization which defined the types of products we could consume (Zuboff and Maxmin, 2002). Following manufacturing innovations, more products could be produced via mass customization (Pine, 1993). A higher degree of interactivity has also been enabled by ICTs between customers and firms, leading to greater opportunities to learn about their needs and to find opportunities for further customization (Miceli et al., 2007). ICTs have been enabling managers to be more familiar with user needs and can help them facilitate service and product reconfigurations to continuously improve these systems (Bermell-Garcia et al., 2012; Needham, 2011).

Recent developments in technologies characterized as AI introduce newfound capabilities for personalization. Not only do these technologies allow for the unpacking of greater information from our data about our users than has ever been accessible (Mullainathan and Spiess, 2017), but they allow for vast new system-generated personalization opportunities because these systems are able to make inferences about customers on their own and are able to execute service functions autonomously. Increasingly, service experiences are being mediated by autonomous recommendation engines, natural language processing, and reinforced learning from behavioural data. They can also exist in complex webs of interaction across different algorithms, systems, and users. The number of digital agents, so to speak, between customers and employees is growing. However, with all of their power these technologies come with properties that introduce risks related to privacy (Aguirre et al., 2016), trust (Marino et al., 2020), algorithmic interpretability (Totaro and Ninno, 2014), algorithmic explainability (Barredo Arrieta et al., 2020) and algorithmic transparency (Waltl and Vogl, 2018). This research aims to deepen our understanding of AI-mediated personalization by looking at both the material features of these emerging technologies by opening up the black-box, as well as social features related to developer interpretations and negotiations with these systems. By using AI to mediate personalization, we are increasingly using agents that remain a black box even when we open them up.

Technology-mediated personalization does not come about spontaneously. It involves engineers interpreting the capabilities of their technologies, a material dimension, but also social dimensions like subjective user desires, organization goals, and the interpretations of service designers themselves (Leonardi and Barley, 2010). This research is particularly interested in the service designers. Service designers play an important role in the ability of an organization to successfully personalize services (Piccoli et al., 2017). This research closely follows 34 personalization features developed by an AI company between summer of 2018 and spring of 2020. Each feature was mapped from inception, to design, to implementation, utilizing full access to the codebase, design documents, as well as observations and interviews with developers at all stages of their development. Nearly half of the personalization features involved giving users more choice, usually with highly interactive interfaces. The other half were algorithms developed to make computations over behavioural data and output recommendations for personalization utilizing ML and NLP.

What emerged is a significant difference in adopted configuration and negotiation strategies between personalization features that were perceived as complex in either implementation or understanding. Unpacking this further, complexity in implementation, or how easy it is to develop, required different strategies than complexity in understanding, or how easy it is for developers to explain what exactly is happening. Similarly, the design of AI-mediated personalization involved different strategies from those that utilized complex interfaces but no AI. AI technologies embody *learning* from social contexts that make it difficult to fully understand why exactly an algorithm made the choice it made. And as this social learning becomes increasingly interwoven and interconnected, designer abilities to *understand* are instead shifting to making guesses or *interpretations* about the functions. That is, while some personalized AI systems are easy to implement, if they employ social learning they become something of a black box to designers. Despite being harder to understand, their impact on personalization is significant because these technologies are able to draw from social learning to make far better guesses about user needs than designers could have before these technologies became readily available (Kumar et al., 2019; Mullainathan and Spiess, 2017). Practical strategies emerge for configuring and negotiating with different types of personalization technologies.

7.3 A literature divided

Personalization is body of research that crosses the disciplines of computer science, economics and management (Adolphs and Winkelmann, 2010; Kwon et al., 2010). It is a field that has been on the rise, with the number of related research articles more than tripling each decade since 1990 (Sunikka and Bragge, 2012). Three influential literature reviews have found broad consistency over time with how personalization is defined (Adolphs and Winkelmann, 2010; Salonen and Karjaluoto, 2016; Sunikka and Bragge, 2012) Generally, personalization is defined as a process of tailoring products and services to match individual preferences. Synthesized learning about customers can help shape offers, recommendations, customizations, or multiple interaction touchpoints (Miceli et al., 2007; Vesanen and Raulas, 2006). Learning can also be used to change functionality, information content, and the distinctiveness of a system to increase personal relevance to users (Blom and Monk, 2003). These learnings can come from personal information about user preferences (Chellappa, Ramnath and Sin, Raymond, 2005), such as what information is directly shared by users, or information that is inferred based on their behaviours and profiles. Websites are particularly relevant because they can collect information about users from user habits like clicks (Arora et al., 2008; Ho, 2006). This research considers this learning about users as an imperative lens.

The literature has also been largely characterized as being split between user-centric and technology-centric research (Adolphs and Winkelmann, 2010; Salonen and Karjaluoto, 2016). User-centric studies have emphasized the impact personalization has on users themselves and their perceptions. Personalization has been generally seen as positive, linked to satisfaction (Devaraj et al., 2006; Ha et al., 2010; Herington and Weaven, 2009; Jiang et al., 2010; Liang et al., 2006; Piccoli et al., 2017), service adoption (Krishnaraju et al., 2016), can lead to trust (Aguirre et al., 2016; Komiak and Benbasat, 2006; Mukherjee and Nath, 2007), and loyalty (Che et al., 2015; Mukherjee and Nath, 2007). However, when used in certain ways, personalization can lead to feelings of intrusiveness which harms business performance (van Doorn and Hoekstra, 2013). Given personalization can use private information to generate richer inferences about users, privacy issues dominate much of the literature around trust

(Abu-Dalbouh, 2016; Aguirre et al., 2016; Awad and Krishnan, 2006; Jackson, 2018; Karwatzki et al., 2017; Li et al., 2018; Weinberger et al., 2018; Xiao et al., 2018).

The positive effects of personalization are quite conditional (Sunikka and Bragge, 2012), such as on cultural effects, timing (Bodoff and Ho, 2014), location (Abilash Reddy and Subramaniyaswamy, 2015), where in the buying journey a customer is (Lambrecht and Tucker, 2013), and personal disposition like motivation (Li and Liu, 2017). That is, personalization is deeply contextual (Salonen and Karjaluoto, 2016). This is posited as a challenge in the literature. Personalization therefore becomes more of a process of learning rather than a static concept (Adomavicius and Tuzhilin, 2005; Vesanen and Raulas, 2006).

The second major cluster of personalization research has been categorized as technology-centric (Salonen and Karjaluoto, 2016). These largely emphasize implementation techniques, strategies, and challenges associated with technologies in their design, adoption, breakdown, reconfiguration and more (Orlikowski and Scott, 2008). An understanding of design factors has been found to be essential for successful web personalization. Design when done right can increase trust (Li and Yeh, 2010) and loyalty (Chang and Chen, 2008). The design of personalization relies on previously collected customer data (Arora et al., 2008). This data can be inferred from the consumer's behavior or transactions (Montgomery and Smith, 2009), user search history (Yoganarasimhan, 2015), user profiles (Gajos et al., 2010), product views and clickstreams (Yang, 2010), customer life-cycle stage assessments (Ahn et al., 2010) and more. More recently, using computational strategies organizations are also trying to extract and process data to infer personality (Arazy, 2015; Capuano et al., 2015), implicit needs (Chang et al., 2009; Qiu et al., 2018) reputation and expertise (Martín-Vicente et al., 2012). Advanced data is needed to enable this deeper inference across social relations (Lee and Brusilovsky, 2017; Li et al., 2013), changes in interest over time (Li et al., 2014), and more.

A theme from this technology-centric literature is that personalization is difficult to implement as a business tool (Sunikka and Bragge, 2012) and that there is a difference between 'personalization done' and 'personalization done well' (Fan and Poole, 2006). Ultimately, preference finding is difficult (Chen et al., 2010). As highlighted in the user-centric literature as well, preferences were for too long viewed as static (Tuzhilin, 2009) while in reality user data is deeply contextual (Goldin et al., 2006).

There are promising opportunities emerging from new technologies when it comes to inferring the needs of customers. Big data has led to better personalization and customization (Anshari et al., 2019). Knowledge management, such as knowledge about users, has been found to be better facilitated by social software than traditional software (Von Krogh, 2012). Traditional rules-based expert systems are being disrupted by a deep-learning approach that puts data at the center of decisions (Mullainathan and Spiess, 2017). The growth in AI in particular has been noted. The high degree in personalization inherent within AI is considered to be a major factor behind it's growing popularity (Kumar et al., 2019). Thus far, however, the literature has treated these innovations as like any other. The research in this paper supports the argument that a more sensitive measurement of social as well as materially-rigid dimensions of these new approaches reveal dynamics that suggest a drastic altering of the role of the manager of personalized services, and this may have implications for our broader social networks.

7.4 Motivation - Bridging the divide to better understand AI-mediated personalization

Despite the recognized bifurcation in the literature, few personalization studies bring user-centric and technology-centric measurement together. Epistemological differences between the subjective logic of users on the one hand and the underlying reliability of computational logic on the other make bridging these perspectives challenging. This work draws upon similar debates taking place across organization, management, and information systems literatures and contends that separating the research into these streams has prevented a meaningful understanding of the ways personal value is generated when we rely on an intermediation between the complex social world and the highly standardized technological world. Social and material dimensions can be measured together, and in fact should be given their interdependence.

Technology-centric views of organizational management have been criticized for assuming endogeneity, homogeneity, predictability, and stability in how technology shapes humans, ignoring historical or cultural influences (Orlikowski, 2007). Technology is treated as an independent factor that if adopted, or not, has predictable influences on organizations. The problem is that evidence has shown that technology adoption is in fact deeply embedded within the complex social contexts across an organization. Technology with the same properties could have significantly different outputs in different settings due to social dimensions (Leonardi and Barley, 2010).

On the other side of the spectrum, user-centric views have been criticized for trivializing the material and predictable aspects of technology that do exist (Berg, 1997). Materiality, it turns out, matters a great deal. The specific functions of technology have traits that are necessarily closed from social influence (Arthur, 2009). For example, a simple logic-based computer program will run the same regardless of its place in time or geography. Similarly, they necessarily simplify data and experiences (Kallinikos et al., 2013). Critically for technology-mediated personalized services, social data that is used to inform or enable personalized experiences is highly dependent on the shapes of the data models that algorithms use to compute responses and dependent on the point of input where user experiences are converted into digital artefacts representing them. Ignoring these material features means losing sight of key factors that define the ways individual and social interaction can take place.

Humans can exercise agency with the way they interact with technology, but those choices are shaped by materials, structures, or processes that permit or limit certain actions (Leonardi and Vaast, 2017). Individuals make perceptions about what they can do with a technology, called an affordance (Leonardi, 2009), or what it cannot do, a constraint (Leonardi, 2011). Affordances and constraints become both a social as well as technical concepts. This view that technology is constitutive of our social life and its application into studies of organizations has been a major motivation behind this particular study.

Alongside this deeper understanding of personalization, this research aims to contribute to the success of a stream of research that bridges the gap, called sociomateriality (Orlikowski and Scott, 2008). “It is useful, in one sense, because it provides a way of transcending the two dominant (and antithetical) theoretical positions that organizational scholars have adopted in their study of technology: technological determinism and social constructivism.” (Leonardi and Vaast, 2017). Advances in sociomateriality have offered a more refined way to tackle research questions involving highly interdependent social and technology systems, such as the social tools becoming increasingly ubiquitous in the workplace.

Consider the literature from a related and growing class of ICTs, that of platforms. This literature tackles problems of highly distributed relations (Reuver et al., 2018) between users and social technologies that enable vast and scalable interactions. Following the interactions between human actors and the platforms that bring human actors together, though shaped by specific technical functionalities, has allowed for the discovery of a complex relationship of iterative design (Germonprez and Hovorka, 2013). An understanding of systems is not within grasp to even the designers, according to this literature, until they are built, interacted with, and then reconfigured (Germonprez et al., 2011). ICT-based platforms are ever-changing systems due to the living nature of the data and algorithms that configure them, and that makes researching them challenging. That is, the data and the algorithms themselves, the structuring role they have on user perceptions, are becoming important artefacts worthy of our investigations (Alaimo and Kallinikos, 2017). One cannot understand these systems as a snapshot in time for example, or in isolation like much of the older personalization literature contends (Tuzhilin, 2009; Vesanen and Raulas, 2006). These algorithms and features need to be understood as webs of reconfiguring human and machine innovations (Tilson, 2010). A longitudinal review of the evolution of innovations becomes valuable (Germonprez et al., 2011; Tilson, 2010). Seeing the gap in the personalization literature, a goal of this paper is to argue that the personalization literature has not yet benefited as explicitly from this sociomaterial paradigm as literatures like platforms.

This research is designed to extend our gaze into the inner workings of the design and implementation of personalization from the perspective of the designers of those systems. What emerges is a complex set of negotiations that take place not only across material and social domains, but across perceived notions of complexity. The rise of AI has further complicated this complexity because technologies are increasingly taking on roles involving social learning. “Machine learning provides a powerful tool to hear, more clearly than ever, what the data have to say.” (Leonardi and Vaast, 2017). Interestingly, the better they become at social learning and at delivering personalization, the less we understand exactly how they work.

This research contributes to the field in a few ways. We will compare a range of personalization features, some involving AI, and others not, to see if there are key negotiation strategies that emerge. Advances in sociomateriality have offered a more refined way to tackle research questions involving highly interdependent social and technology systems. This work extends this paradigm into the domain of personalization and deepens an understanding of different types of personalization. This has helped reveal a new relationship between managers and these new agents. Beyond simply reconfiguring these systems as they build them, designers of personalized AI are increasingly negotiating with them as if they have properties of agency.

7.5 Case and methodology

Between the fall of 2017 and the spring of 2020, an AI development company (the Company herein) was contracted to develop enterprise applications across four projects. The Company has engineers and project managers scattered across Toronto (Canada), London (UK), Dubai, (UAE), and Kathmandu, (Nepal). The team is primarily remote in operation, with the vast majority of team interactions taking place on their company chat, online video meetings involving screensharing, emails, project documents, their project management ticketing system, and the code base which is a network of repositories hosted on the Company’s GitLab service. The team specializes in building interactive web-browser based tools that support enterprise operations. This research was enabled by deep observation across this period, including participation as an unpaid information systems consultant in most key development meetings, user testing sessions, project sprint reviews, and more. This enabled the observation of 34 discrete personalization features across their ideation, development, and implementation stages. Thus, our case selection was justified on the basis of maximizing information utility (Flyvbjerg, 2006).

As summarized in Figure 7.1, this research involved two key phases. The first was deeply observational, allowing for a close observation of people, actors, systems and for generating more meaningful understandings (Becker, 1996). This was appropriate given the primary motivation of this research is to deepen an understanding of personalization through sociomaterial investigations into the process of their design. Generally the role within the projects was observational, enabling an observer-as-participant balance and minimizing direct disruption of interpretations (Gold, 1958). Given the rich data available the conditions were optimal for a period of deep observation unguided by particular theoretical dispositions. This grounded theory (Strauss and Corbin, 1998) approach aimed to inductively identify important dimensions through thick descriptions (Geertz, 1973) about the design of personalization, including developer understanding of the material properties of the technologies they develop and how these understandings lead to material reconfigurations.

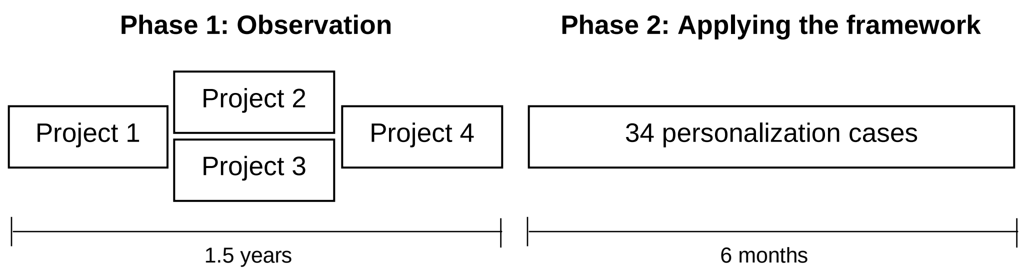

Figure 7.1. Overview of research phases

Our focus on a careful mapping of each technology as a logic function on the one hand and the perceptions of the designers of personalization on the other allow for the identification of the boundaries in agency between material and social actors. Algorithmic mapping involved classifying their logic structures, origins, purpose, design strategies, and challenges that may have emerged. This complimented observed interactions and discussions taking place between the designers. This phase aimed to support mid-range theory building (Eisenhardt, 1989).

Upon reaching theoretical saturation (Gasson, 2011) and having found thematic codes (Miles and Huberman, 1994) with internal consistency and external divergence (Marshall and Rossman, 2014; Mason, 2002), the second phase of the research involved the application of a framework that was built using uncovered dimensions. These 34 personalization features were then mapped from Projects 1 through 4.

7.6 Phase one: Observing dimensions of complexity

7.6.1 The projects

This observation stage occurred across two large projects and two smaller ones. The first was between August of 2018 and the spring of 2019, in which the Company built a series of interfaces and algorithms to support an internal leadership department in the selection of employees. The client was a central agency for a federal government that has as a mandate the adoption of new technologies to improve the country's leadership capabilities. The primary development focus was around the tool's employee search. HR managers were prompted by, “What employee are you looking” over a text-field with the pre-defined suggestions, “Job title, role, position, etc.”. This query is first dissected into root words and components in a process called “stemming”. These are then fitted into data objects that go through a matching algorithm. The matching algorithm is broken up into two independent models, one that is strong but not explainable, and another that has greater explainability. Weights are applied from a machine learning algorithm that is built on top of the process. This algorithm learns from clicks of users, specifically an array made out of 1s where recommendations were shortlisted by users, and 0s where recommendations were ignored.

The second project occurred between July and August 2019 in partnership with a Central Bank to explore two questions. First, could models be built to identify the likelihood that documents being emailed across and outside the Bank contain sensitive material? Second, what are the effects of mixing different trainers together? Can super-users, identified by their subject matter expertise, improve model performance? The original models were built by training a model using a dataset that included 50% documents that were classified as sensitive by experts, and 50% that were not. An explainable algorithm was added to create an environment where super-users could identify deeper opportunities for training. This facilitated an increase in model performance.

The third project occurred in the fall of 2019 in partnership with a mid-sized insurance company. The challenge was to build an interface that supports underwriters in their tracking and monitoring of target compliance and risk. Most of this tool involved interfaces that could facilitate knowledge storage and underwriter-to-underwriter engagement. However, a matching algorithm was used to compare key fields from an underwriters’ case notes to ‘similar cases’ from across the enterprise. This facilitated underwriters learning about investigation techniques and sources used by others across their organization. Additionally, search algorithms were used to facilitate deeper search of cases.

The fourth project is a product that has moved from ideation to pilots by fall of 2019 and was formally launched in the spring of 2020. This tool is under continuous development and improvement. It has been built in partnership with five governments, one NGO that specializes in citizen-centric service delivery, two universities, and two market research companies. This market research tool supports data classification, analysis, and reporting of client and employee engagement data. Surveys, verbatim comments, focus group transcripts and interview transcripts are ‘cleaned’ by users who are guided through prompts that help classify information. These prompts are user recommendation algorithms. Once cleaned and labelled, rules-based algorithms run common analytical models and return optimal models to users that point towards common insights that were traditionally calculated manually. Examples include regression analysis utilizing a series of random forest regression models and variations of hyper-parameters. These are then compiled into customizable reports. Along the way, when users confirm or decline recommendation prompts, training data is collected to support the refinement of the recommendation models. The four projects and their timelines are summarized in Figure 7.2.

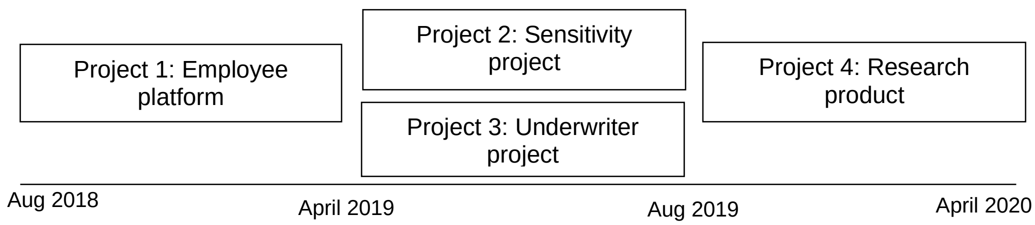

Figure 7.2. Project timeline

7.6.2 The technologies

The primary technology frameworks utilized by all four projects include a Python-based backend that facilitates calculation and database interaction from a frontend using APIs. The frontend is built using VueJS (JavaScript) to allow for highly dynamic and interactive engagement with users through a web-browser. These were installed on servers managed directly by the Company and were situated in the respective country of their clients.

7.6.3 Analytical lens: interactivity mapping

Unimpeded access was given for the purpose of this research. Observation included mapping out the code repositories using 'interaction' as a key dimension. This involved identifying boundaries between users in the real world and the data world. For example, when a user experiences a service what aspects of their experience are converted to code? What algorithms are run, and in what order? This research thus starts with the code. Inspired by interactivity mapping exercises from Human-Machine Interface literatures (Li and Jagadish, 2014), slowly, each and every function was mapped and user interactivity was used as a guide to support this. Along the way, developers were contacted to identify the origins and logic behind all algorithms and interactions. Development chats, project documentation, and project management tickets also revealed clues about origins, key challenges, and design logic. That is, this phase of the research utilized observation of both material features of the technologies, from the code-base first, as well as social features, including meetings, documents, chats, and interviews with designers about their design logics and strategies for overcoming key challenges. These data sources are summarized in Figure 7.3.

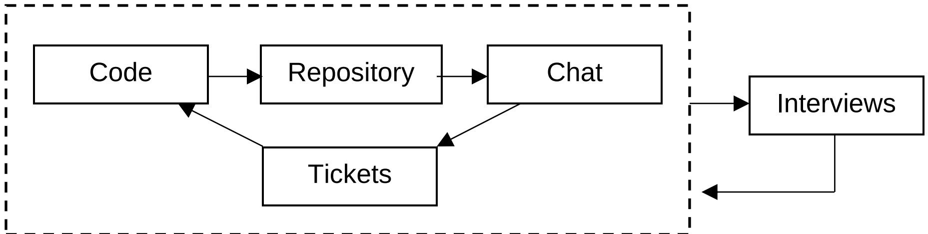

Figure 7.3. Project data sources

7.6.4 Example: Observing the iterative development of a topic analysis algorithm

It is helpful to consider the following illustration of how observation over the course of development led to meaningful revelations that later shaped the second phase of the research. As a recap, the fourth project involved the creation of a browser-based tool that lets market researchers upload data about customers, supports them in cleaning the data using prompts from a recommendation engine, then enables easy analysis. One of the types of analysis that is most challenging for these market researchers is the exercise of identifying meaningful insights from open-ended text data. In small populations and assuming the questions are short, it can be feasible to manually review each open-ended response from customers. But as the number or length of responses increase, this shifts from feasible to unfeasible. When asking a client of the Company how they traditionally manage open-ended text research when the number of responses reach over 500: “we go to the bar” replied one user, with their colleague nodding in agreement. This frustration is commonplace in the research industry. Meaningfully deep analysis of large amounts of open-ended text is challenging to do at scale (Boddy, 2016).

Topic analysis and topic segmentation algorithms (Reynar, 1998) were proposed as a solution. When market researchers do have a manageable amount of open-ended text data, one of the approaches they employ when answering the question “what are our customers saying” is to find a bundle of themes. These themes are supposed to represent individual discreet concepts. The Company aimed to support this market research exercise using an algorithm that automatically reads a collection of open-ended respondents and identifies meaningful clusters of keywords. These clusters of keywords should be sufficiently related to each other to imply they are related concepts to a larger theme. The tool does not name the theme, leaving the user to interpret the bundles of keywords and label the themes themselves. This particular algorithm took approximately one and a half years to mature.

May 2018 – A first pilot

An early version of the topic analysis algorithm was created using Python making use of packages called Spacy and NLTK. This pilot was generated in partnership with a government. An algorithm was produced that creates a list of topics from Tweets selected from the social media platform Twitter. It usually returns two to six bundles of keywords which it calls topics. It does this by taking every Tweet and converting elements like words into base and related structures. For example, a word like 'extremism' could be split into 'extreme' and 'extremist' and 'extrem-'. Each of these are then ran through the above packages and are compared to each other word and sub-word. Words are considered related to one another based upon globally trained word associations collected from vast databases of English text. Words that are commonly found near each other are given higher scores. Adding complexity, the bundles are also expected to be discrete from one another, maximizing word association strength with other keywords from the same topic but minimizing word association strength from the other bundles.

Consider the following example that occurred during early testing (Figure 7.4). Two thousand recent Tweets from a senior cabinet minister were selected. The algorithm returned six groups of keywords. Two groups in particular led to back and forth discussions by developers.

Topic 2 Keywords: iran qatar arab report world terrorism friend right important human

Topic 4 Keywords: qatar extremism saudi terrorism syria region arabia support president gulf

Figure 7.4. Pilot-generated topics

Developer R: "Topic 2 and 4 look similar. We can reduce the numbers of stories to 3, and see if they merge together?

Developer H: "They are not really similar to be honest, one is about Qatar and one about Iran which are distinct foreign affairs issues albeit they are connected"

Developer C: "It is true individual key tweets from one topic may actually fit better with other topics. We may want to build a UI element that lets users "drag" key tweets out of one topic and into another. This will help with our "training" later too.

Developer R: "I still need to think how to do the feedback learning from the algorithm, from the moving a tweet from one topic to the other. We need to study the algorithm itself in more depth as I am not sure yet how it will work."

The results led to different interpretations from three developers. One of the keywords, 'Qatar', was present in two topics. This made it hard to tell the difference between the two groups. Different negotiation activities were explored. One developer recommended tweaking the algorithm to return three bundles instead of six. A second developer disagreed and interpreted the topics as discrete from one another. A third developer had little interest in directly negotiating with the algorithm, and instead suggested giving the user the ability to move keywords around where the algorithm got it wrong. This led to the first exploration of setting up machine learning. While the underlying algorithms were understood, the fact the algorithms return results based on complex social data meant 'why' these bundles were chosen could not be objectively known.

October 2018 – Further doubt:

> **Developer C:** "I think we need to have a series of talks about the way the groups are generated. After talking to a handful of managers, there is some doubt about the strength of the initial groupings (in terms of interoperability at least, if not quality)."

The most conclusive finding from all the testing was that users simply could not trust the auto-generated groups just yet. They did not understand the groupings. There may indeed be high degree of association between these words according to the underlying computational logic of the algorithm, but they do not form groups that are interpretable to users. This led to many months of continuous reconfiguration of the systems and engagement with users.

***May 2019 – Ready for machine learning*:**

Half a year later, with the tool getting to a point where the automatically generated topics were working well, the developers were beginning to think about new possibilities with their technology. Their attention turned to questions about how machine learning could be used to deepen social learning.

> **Market research project manager G:** "Can we add our training modules to the tool? Like, common questions and approaches? We train our teams to code text in certain ways when we onboard them."

> **Developer C:** "Sure. We can let you input your most interesting themes and keywords ahead of time for instance, and make that available to team members when a project starts. But we have been talking about something. What if we could pick up the most popular keywords that your colleagues have used across their research. You said you do dozens of projects a year from this office alone. There could be a lot of good training there."

Developer P: "We can combine this with our project matching algorithm. We can give extra weight to previously used keywords that were also used in projects that get a higher similarity score."

The users were keen to explore facilitating learning for the algorithm from team practices, but as seen below, as long as they also had the ability to turn this learning off.

Market research project manager R: We would love this, but we would want to be able to turn it off. Sometimes certain research projects are too different from the others, so we wouldn't always want to interfere with the broader and more widely tested modules.

March 2020 – When good enough is delightful:

Users were beginning to appreciate the algorithm not necessarily for having no mistakes, but for having at least a few results that were powerful. Tolerance levels for errors grew as the users signalled the tool was better meeting their personal needs at least some of the time. Consider the following example (Figure 7.5). The topic analysis algorithm was run over around 300 customer comments about touchscreen features.

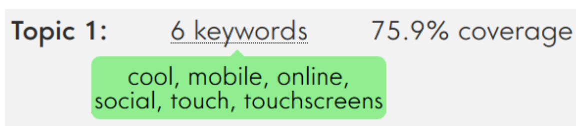

Figure 7.5. Application-generated topic

Market research project manager T: It's a dataset with customer feedback on our touchscreen features, so mentioning touch and touchscreens makes no sense. And I don't get what social and online mean. But, 'cool' is interesting. 'Cool' is related to the touchscreens?

The user more or less disregarded most of the keywords from this generated bundle, but interpreted some value in 'cool'. They then ran the tool's 'advanced search' on 'cool' (Figure 7.6). This advanced search is a word association algorithm built from similar Python packages as well as custom Company logic that returns similar words that have been found in the text.

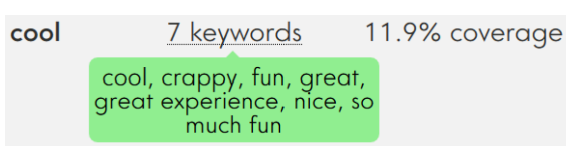

Figure 7.6. Advanced search on 'cool'

This search created a bundle that started to make sense. This advanced search on 'cool' returned a bundle of keywords that seemed to reasonably convey a sense of 'cool' or 'fun' over about 12% of the customer comments. There was still one more artefact the user opted to remove. When 'Crappy' this was removed, the coverage landed at 11.6%. The market researcher could confidently say at least 1 in 10 customers had a sense of fun or cool when using the touchscreen. This took interpreting an automatically generated topic and picking out the features that were interesting. Running an advanced search on this interesting feature revealed similar notions from other customers. Within a few clicks, the market researcher went from no insights to an insight about customers. Of note, a later feature emerged allowing users to 'ban' keywords that the tool was generating that did not meet their needs. This allowed for users to 're-run' the algorithm and get a new generated bundle of themes. Thus, personalization of the algorithm began to be managed directly by users.

Summarizing this observed example, across the development of this tool for market researchers several challenges emerged. The earliest versions were difficult to understand or interpret. Bundles of keywords may have been associated with each other in a mathematical sense thanks to the globally trained set of word associations, but it was not always clear to the users what these bundles convey. As a negotiation, developers gave users the ability to remove bad matches to clean up the user's workspace and improve the precision of the themes, combine multiple training models together, and to activate learning from other team members, as long as it can be turned off. That is, the final themes produced from the use of the tool will become a product of both the algorithm's original socially-trained suggestions from the topic analysis algorithm, as well as the interpretations and modifications of the user over their results.

The observation phase of this paper revealed countless examples of similar negotiations taking place between users and designers, some due to material features, and others due to social demands. As in the above example, tools that involved high degrees of social learning led to results that developers could not always understand. They understood the underlying logic of the algorithms, but the data fed into them was too dynamic. The engines created recommendations and the designers and users responded and reconfigured. Interestingly, developers did not always decide to understand an algorithm if they could not. Instead, they would add layers of interactivity, adaptability, and contextual training to accommodate the algorithm rather than directly change it.

7.7 Phase two: Testing a framework of personalization design and complexity

The first year and a half of research led to an unpacking of sociomaterial negotiations live. This enabled a longitudinal mapping of algorithms from their earliest conceptions, through early iterations, to mature versions deployed across organizational contexts. As illustrated in the topic analysis algorithm example, dimensions of design, perception, and complexity were consistently observed. As more personalization features occurred over the observation period, it was increasingly clear that these dimensions proved analytically valuable for cases of ICT-mediated personalization

Is this feature an example of personalization: At a basic level, personalization is any feature that matches to individual preferences. They provide personal relevance (Blom and Monk, 2003) by use of user data (Salonen and Karjaluoto, 2016). Once a feature has been identified that appears to be a personalization feature, a careful mapping of the code was conducted to reveal clues about design logic, interactivity with user data and other social learning, as well as key challenges that emerged. This was aided by a strong grasp of the code base. Two primary ways personalization was facilitated by the Company was to enable frontend customizations or capabilities for a user, where user information takes the form of clicks and requests. These are label as *user-initiated personalization features*. The other is by using algorithms that infer something meaningful for a user and to be displayed on a frontend for them to interact with. These are labeled as *system-initiated personalization features*. Once a personalization example has been mapped and given a cursory classification, the research turned to user stories, chats, interviews and more across each dimension of the framework.

Defining user needs through stories: A work process that significantly benefited the research journey is the fact the Company developers use a technical development ticketing approach whereby all technical features begin with *user stories*. These are descriptions of what a user would like to experience. This formalized exercise involved the use of a ticketing software called Jira, as well as the filling out of a standardized project document that asks the developers to define the purpose of the project explicitly by defining capabilities users wish to have. These 'stories' are analogous with the signifiers of personalization that inform personalization design as identified by the literature (Piccoli et al., 2017; Sonenshein, 2016). Clear communication is needed about what users want, about the structure and operation of the technology, and more (Haraty and Mcgrenere, 2016; Norman, 2013).

Initial design: With personalization cases identified and their user stories elaborated, developers proposed solutions in either the Company chats, technical development tickets on Jira, notes in the code through comments, or notes when testing and merging different code bases together. Perceptions about technology capabilities shape adoption (Leonardi, 2013), so it is interesting to see the perceived capabilities certain technologies have to different developers. Underling how socially-embedded these processes are, some of the developers had a better grasp of the capabilities of frontend technologies. These were well suited for solving personalization features that requested more options and customizability for users. Other developers had a better grasp of the types of data science tools broadly available, and how they can be incorporated into possible designs. These were well suited for designing systems-initiated personalization for example. The team found synergy thanks to the strengths of different developers and their understanding of what technology can do.

Understanding what was built: There was almost always a difference between the initial design and what was built. Technical and social challenges commonly emerged in the development of the tool that were not predicted. These led to the developers having to make sometimes slight, or sometimes significant changes to their original plans. They still achieve a solution to the original user story, but how it got achieved may have been easier or more complicated than planned. At this stage of the research, with personalization features and original plans identified, developers were asked to explain what actually happens. Often these requests for explanation came directly from users and were available in company meetings and chats.

Interpreting what was built: As we have observed, interpretation matters. Because different groups associated with technology development have their own goals, objectives, and social constraints, they ultimately interpret technology in different ways (Pinch and Bjiker, 1984). The actual functionality of the technology is the result of these different interpretations (Bjiker, 1995). In the first phase of research it was observed that not only was there a difference between what was designed and what was actually built, but also differences in which technologies were readily understandable and which ones were not. For example, the presence of a perceived black box was a common indication that there would be elements of the functioning of the technology that would simply never be known, or not worth pursuing. Not all features involved 'black boxes' as the developers called them, as they felt confident about the logic behind the functioning of features that follow the rules of the developers' frameworks. Sometimes projects would be challenging to build due to the amount of code required, but the developers retained confidence in how it worked and why it results in outcomes it does. Those systems that incorporated guessing based on previously compiled social learning, on the other hand, overlapped with what developers called black boxes. Those personalization exercises that used recommendation systems were built using training from global datasets, or directly from user clicks through machine learning. But the underlying key reasons for decisions made by the models were beyond the comprehension of developers. The mathematics may have been known, but developers could not predict outcomes by looking at inputs. Reconfiguration of these algorithms involved more guess work. Throughout many of the personalization features, additional algorithms were added on top of algorithms that cannot be interpreted, to create a layer of abstraction to guide users towards having a better sense of what was selected. The most common way this emerged was in the form of feature called "key reasons" which was added on top of, but not directly related to the primary recommendations. It can be revealing to observe how developers embrace the strengths of algorithms that involve deep social learning while simultaneously lacking a complete understanding of the complexity behind their decision-making.

Characterizing the key negotiations: The final step of the framework after mapping design, understanding and interpretation was to classify the types of negotiations that were utilized. To start, consider the challenge of configuring technology. The challenge for these developers was building a solution with tools they know, but figuring out how to configure everything right. These challenges test developers' unique understandings of technology capabilities, which is a social gaze upon the real and material world. The primary task is to try and pull those diverse developer assumptions about material features together into a way that solves a complex problem (Suchman, 2007). Thus, overcoming these challenges involved *human-to-technology negotiations*.

Other problems were not about technology complexity, but about not being able to guess what users want. These problems usually involved engaging with users to map their needs and requests to help balance or prioritize diverse needs. These can be classified as *human-to-human negotiations*, because the process of building this is less about the technology, and more about understanding what needs need to be met.

As a final category, there were cases where the developers had to negotiate with a system that was making recommendations based on deeply social data. This data could not be recalled and interpreted by humans, because it is live and interactive, and thus it requires special kind of social technologies like machine learning that can enable calculations involving diverse behavioural data on a vast scale and speed. This highly embedded and complex environment requires navigating through *human-to-sociotechnical* challenges, both the technical features of selected algorithms and unknowable social data and social interaction. It could be argued all negotiations are human-to-sociotechnical, or even sociotechnical-to-sociotechnical, but this research has found value in differentiating between those whose primary focus is material, social, or deeply both.

In summary, the second phase involved a deep review of the 34 personalization cases using dimensions from a framework on personalization design and complexity.

Deep investigation of each feature that can be classified as 'personalization':

a. Are they algorithms or interfaces? (systems-initiated or user-initiated)
b. If algorithms, what kind?
c. Why was it chosen as a feature? (Signifier/User story)
d. Why was this design choice selected? (Developer plans / response to the signal)
e. Once built, could developers explain how it functioned?
f. When explaining function, how often was interpretation needed (versus explanation)?
g. Is there a black box, as self-identified by developers?
h. Overall, when building, what was the main focus of 'negotiation'?

Reframing the above framework of questions into the perspective of the developer, we asked them to define personalization as a user story (the users want to do X), describe its design (it will be able to do X), understanding it once built (it does X), or interpreting it if unclear (I think it does X), and key negotiation strategies (can we make it do X?). This framework appears beneficial when done through qualitative methods such as interviewing developers and reviewing documents and archives. However, our case was made even stronger by the fact we were able to build a comprehensive understanding of the technologies as material artefacts, as well as the development ecosystem in the first phase of research. That is, this research was able to recognize the different dimensions of complexity in part due to familiarity with both social and material aspects of personalization design.

7.8 Findings

The 34 personalization features were split into 14 algorithms that represented system-initiated personalization, including six matching-algorithms, two machine learning applications, one computer vision algorithm, two natural language processes algorithms, and two prediction algorithms. They were also split into 20 frontend features that enable user-initiated personalization, including eleven adaptable user interfaces, four social networking features, three file management features, and two project management features.

Table 9 provides an overview of the example we reviewed from the first phase of the research. An automated topic analysis tool was developed as part of the market research platform and was deemed a feature of personalization because its system generates results based on the unique data a user has uploaded. Designers built a pilot that generated topic bundles that did not make sense to users, and developers could not fully explain how the word association made the choices it did. They understood the logic, but the exact data and interaction was fleeting and unknowable. This led to complex technical iterations over a year until the tool was adaptive to user needs.

Table 9. Mapping the 'automated topic analysis' feature

Case	Automated topic analysis
Project	Market research platform
Technology	Python: LDA, Spacy, NLTK
AI	Yes
Personalization	[System-initiated]
User Story	I want the tool to suggest topics that exist within text data.
Design	We will use natural language processing and clustering approaches.
Understanding	Using packages the tool stems and lemmatizes open-ended text to create bundles of highly associated keywords
Interpretation	What do we do with suggestions that just don't make sense?
Technical complexity	Difficult to implement

Ease of understanding	Difficult to understand
Training	Globally trained word associations + local trained tuning
Black box	Yes
Negotiation	**Human to sociotechnical** - The LDA interprets correlated clusters of words, but not interpretable to users. Discussion about letting users adapt and improve results.

Our example also revealed that when building a machine learning engine for algorithms like the above, users began requesting control over when that machine learning activates or not. This resulted in a separate personalization case, one that is user-initiated (Table 10). In this case, users can select if they want their projects to be influenced by their past training, their organization's data, or their social network's data. The design was simple to implement. Based on the user's choice, recommendation engines will rely on different vectors of previously trained information. Adding user experiences to globally-produced vectors is easy, as multiple training sets could be easily added together by concatenation, which is the joining of vectors end-to-end. This allowed for further control. All in all, the process took a day to set up. However, the inner workings of the algorithm remained opaque. The focus of negotiation remained largely around how to weight different training sets together. Should they all be treated equally? This was resolved through back and forth testing with the algorithm until an optimal configuration was set. This was therefore classified as a largely sociotechnical negotiation because it involved making interpretative tweaks to a system built from complex social data. Because data could be complex mixes of different and ever-changing sources, there is no meaningful way to reverse engineer the computations. Learning is built on top of itself in nested ways that get lost as new layers enter. The tool seemed to work well enough, as recommendations improved over time. It also delighted users because it gave them a choice. One thing in common with these two features is they both involved a negotiation with an algorithm that utilizes complex social data, albeit different data. These two examples show that while both are hard to understand, sometimes complex social-data algorithms are actually easy to implement. There is often, however, an acceptance of the unknown. Further still, sometimes there is a need for explainability and abstractions are created on top of already opaque systems, but other times there is no effort to enhance explainability at all.

Table 10. Mapping 'giving users control over machine learning'

Case	Giving users control over their machine learning
Project	Market research platform
Technology	Machine learning - combining training
AI	Yes
Personalization	[User-initiated]
User Story	I want to be able to control when I am training individual, organizational, or social network datasets

Design	We will give users a toggle that will give users control over their benchmarking
Understanding	We combine datasets of selected training models by concatenation of training vectors
Interpretation	Unable to explain the inner workings of the model like before. No explainability given.
Technical complexity	Easy to implement
Ease of understanding	Difficult to understand
Training	Trained by user + Trained by organization + Trained by network
Black box	Yes
Negotiation	**Human to sociotechnical** - How do we decide the strength of the various sources of training?

Consider the following mapping of a personalization case from the Central Bank innovation experiment (Table 11). The team was tasked with building a model that can detect sensitivity inside Bank documents. If an email, for example, was flagged as 'sensitive' by a system, users would have to run the document through a process to ensure it passes organizational security policy requirements. An original model was built by taking a large sample of emails that had previously been labeled as sensitive, versus a large sample that had not. This model was not explainable, only giving a response of "sensitive" or "not sensitive". The additional challenge was for the tool to incorporate learning from super-users like specially-trained security experts, to improve the algorithm at a faster rate than general employees. To facilitate this, trainers repeatedly asked for explainable features, such as 'key reasons' for why matches were made. The Company built a series of independent algorithms that interact with each other to facilitate an environment where trainers can get hints about possible reasons. This helped the trainer better interact with the algorithms which helped encourage more training. This is yet another example of sociotechnical negotiation because the models necessarily incorporate complex mathematical calculations over text data that we could not feasibly do manually, and results cannot be reverse-engineered due to the algorithms selected. The process of giving some users more power than others involved negotiating with human users and making judgement calls about how different algorithms can interact.

Table 11. Mapping the 'sensitivity score'

Case	Sensitivity score
Project	Central bank innovation experiment
Technology	Matching algorithm
AI	Yes
Personalization	[System-initiated] Impute the expertise of sensitivity experts to improve the algorithm
User Story	Can we train a sensitivity algorithm from all our past sensitivity scores? Can we improve it by further training using experts?
Design	Experiment between matching algorithms built using general versus expert employee coding. Used combination of interpretable and non-interpretable algorithms to aid the training.

Understanding	Expert-trained data proved more precise but sample sizes were more difficult to collect
Interpretation	A need for knowing why a document was scored the way it was required a custom architecture.
Technical complexity	Difficult to implement
Ease of understanding	Difficult to understand
Training	Training by general users + training by subject matter experts
Black box	Yes
Negotiation	**Human to sociotechnical features** - How to find the right balance between general-trained and expert-trained. And using trained data to 'inform' tester, but confirmed by tester.

Not all personalization negotiations involve deeply embedded social data. Long before AI and machine learning, personalization was made possible by giving users choice using interfaces and functions facilitated by rules-based calculations. Many of these calculations do not require computation from the backend, using Python, and can be run directly on the web browser's engine using VueJS (JavaScript). Some of these are neither difficult to understand nor difficult to implement. These are quick wins for personalization but can often be seen as token forms of personalization. However, some interface features can be complex and challenging to build. Consider the example below (Table 12) from the market research platform. One of the main purposes of this platform is to accommodate diverse types of customer experience data smoothly and to be accessible for non-technically trained users. As such, the system needed to allow users to upload what ever data they have and be guided through standardization checks. It was decided that the best method for facilitating this complicated standardization of diverse and often unstructured or semi-structured data was to go through each column of data and make conditional rules-based checks, like 'if' and 'else'. The code would look for certain artefacts that signal likely data types are found, like numbers formatted like dates signalling time data, or text content signalling open-ended data, or numbers ranging from 1 to 5 signalling a survey scale. Through an interface, a user is guided through a step-by-step checklist to verify data types and clean out unwanted artefacts. This was tremendously challenging to develop and took many iterations and extensive testing with users. However, when developers are asked if they could explain the code once it was built, they were firmly confident they could. The complexity was from the many rules that had to be detected and manually written, and in ensuring it is smooth and easy for the user to use. That is, this did not incorporate any machine learning or technology perceivable as black box. Developers had strong awareness of their technology stacks, be they backend or frontend, and especially after having completed the journey, could re-trace every line of code and every function in these features, and could describe the strict rules the functions follow. They could look at an input and confidently and accurately predict the code's output. This is unlike what they called black boxes, which utilize computations that cannot be recreated or unpacked. That is, these rules-based functions are unlike the algorithms that receive an input and compute an output that needs to be interpreted. This example underlines the differences between complexity of implementation versus complexity of understanding. The primary challenges facing the developers was therefore the complexity of designing these rules-based systems on top of each other in a way that could facilitate the task. This was therefore primarily negotiation with material features.

Table 12. Mapping the 'dataset classification' feature

Case	Dataset classification
Project	Market research platform
Technology	File management
AI	Maybe
Personalization	[System-initiated] helps the user standardize their data in a meaningful way.
User Story	I want the tool to make guesses about the type of data I have just uploaded
Design	We will read columns of data and run a series of logic checks
Understanding	It will check for columns with only unique values (ids), those with values that fall within common survey scale ranges while accounting for missing variable labels
Interpretation	Why did it make a bad guess? Because the various computational logic checks do not catch all cases.
Technical complexity	Difficult to implement
Ease of understanding	Easy to understand
Training	None
Black box	No
Negotiation	**Human to technical features** - Add new logic checks. Eventually consider adding a matching algorithm.

The central challenge and focus of negotiation may not always involve complexity in understanding. It may also may not even be technical in nature. Consider the following examples (Table 13), one involving a social profile feature, and another an example of adaptable UI. Users wanted some degree of enhanced choice and customization. Fortunately for developers, many of these requests can be implemented easily. The challenge is in knowing the needs of the users. What exact customizations do they want? How do we know if only some users want it? How many users should want it before we implement it? These are negotiations that are human-to-human in nature, and the technology in these cases do play more of a backstage role.

Table 13. Mapping 'organizational management' and 'UI' features

Case	Organizational management	Remember UI settings

Project	Market research platform	Research platform
Technology	Social profile	Adaptable UI
AI	No	No
Personalization	[User-initiated]	[User-initiated]
User Story	I want to be able to send invites to other members of my organization and manage them in one interface.	I want the system to remember my UI settings, like my preferred language, sort methods, icon sizes.
Design	We will allow accounts to be created via invite, and will create an admin for organizations that can manage their users.	We will store user-initiated personalization on the UI and recall them when they return to the application.
Understanding	When a user invites members, they will automatically be the admins, and the invitees will join under their organization.	The system loads preferences from the database that are saved asynchronously as users change preferences.
Interpretation	No interpretation documented	No interpretation documented
Technical complexity	Easy to implement	Easy to implement
Ease of understanding	Easy to understand	Easy to understand
Training	None	None
Black box	No	No
Negotiation	**Human to human** - What kinds of controls do admins want over their users?	**Human to human** - What UI can be made customizable and why? Issues of accessibility characterized these discussions.

The framework when applied to individual cases of personalization proved valuable. Different dimensions of personalization and key areas of challenge and negotiation could be explored across the life cycle of the development of the features. It allows for finding features that are going to yield quicker results, even if token. It also shows that using advanced social technologies unlock powerful recommendations that can improve personalization, even if these sometimes involve opaque algorithms that are difficult to understand.

The above exercise was applied to each of the 34 personalization cases. This enabled a macro look at some of the common patterns taking place across the Company's varied personalization features. Figure 7.7 summarizes some of the dimensions identified from the framework and exercises above. This helps visualize some of the common patterns. Nodes (circles or squares) represent a personalization feature. Squares represent those features that developers declared use black boxes, while the circles do not. As we learned, some personalization features are user-initiated and others are system-initiated. These have been labelled to aid visual representation.

Complexity plays a central role to this representation, technical complexity (how easy or difficult it is to implement) is along the Y axis, and ease of understanding why an input leads to an output is along the X axis. This creates four distinct quadrants. Those personalization features that are easy to implement and easy to understand are in the bottom left, and those that are difficult to implement and understand are on the top right. The final dimension that is reflected in the visual representation are the types of key challenges and negotiations. Orange nodes reflect personalization features that were largely categorized as involving human-to-human negotiations, Blue nodes reflect personalization features involved human-to-technical negotiations, and green nodes reflect personalization features involving sociotechnical negotiations.

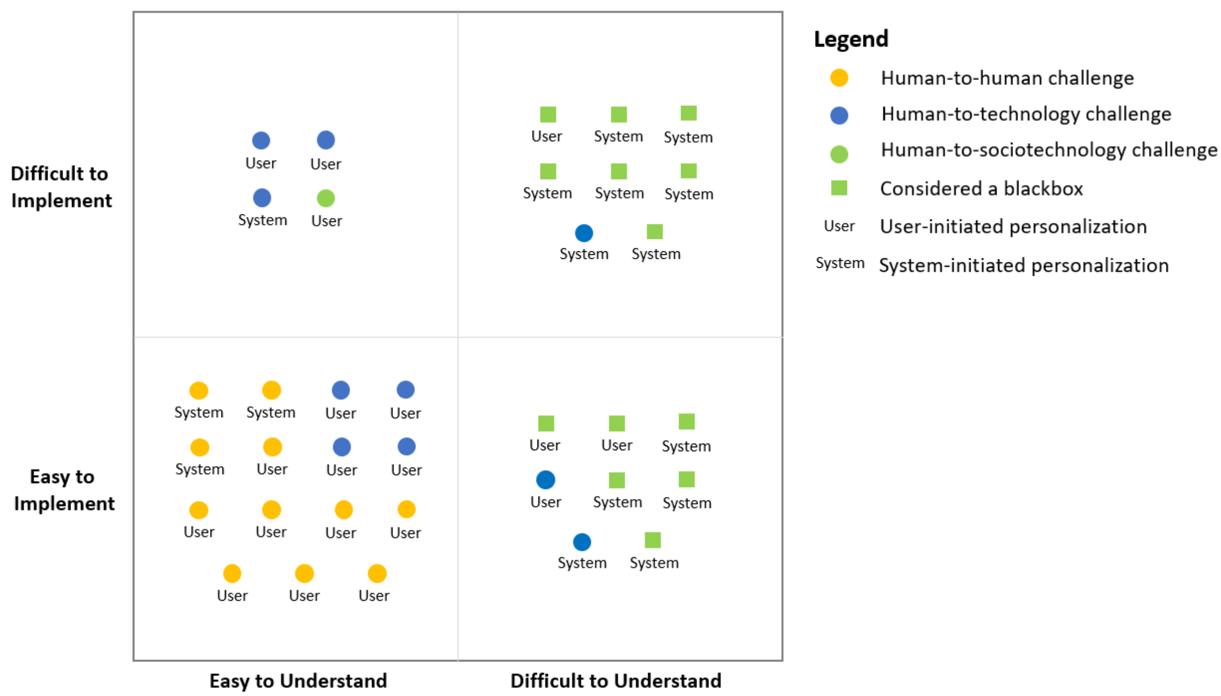

Figure 7.7. Visualizing personalization design complexity

Personalization features whose challenges were mostly the negotiations between the developer perceptions about user needs (human-to-human negotiations) were largely easy to implement and easy for developers to understand once built. Time and effort was spent on understanding users. Personalization features whose core challenges were technical in nature (human-to-technology negotiations) were largely easy to understand once built (7 out of 10). The examples that were difficult to understand once built involved complex encryption and advanced mathematical analysis but did not involve social data. For the most part, features characterized by human-to-technical negotiations involved user-initiated personalization (8/10). Sociotechnical negotiations were almost always involving features that were difficult to understand (12/13). The only exception was not involving social data. This was the example of allowing users to bring multiple methodologies into a tool. It was considered a human-to-sociotechnical negotiation instead of just human-to-human because there was complex discussion about how to meld dimensions of organizational practice like routines and flexibility into a single interface. Therefore not all personalization features that are deeply sociotechnical involve social algorithms, such as functions that become highly layered, dense, and interactive in order to accommodate diverse and complex user needs. Another observation is that not all features involving the use of sociotechnical negotiations are necessarily challenging to implement. There were many examples of quick-wins where developers were able to get a natural language processing algorithm up in a single day. While 'understanding' the computation behind these escaped developers when it came to explaining specific inputs and outputs, negotiations were not necessarily long before it delivered value for users. One last important observation is that the most challenging personalization problems, involving both difficulty in implementation and difficulty in understanding, required skills in navigating sociotechnical negotiations. As these tools continue to be expanded into every-day work, there will be more opportunity to develop these development skills.

"The most challenging personalization problems will increasingly require sociotechnical negotiations."

A few other trends of practical value were observed across the cases. The Company was able to successfully configure many types of personalization to solve varied problems. Search and recommendations were well suited by matching and natural language processing approaches. Data classification, on the other hand, does not necessarily need these advanced computational methods. Simple rules-based 'if' and 'else' logic when built into an accessible interface can let users go through structured 'checks' of data. However, just because the tools do not use advanced methods like social computing, does not make them easy. The development of a dynamic data classification system without the use of AI was nonetheless was a herculean effort involving careful testing with users. Adaptive UI can easily characterize half of the personalization features occupying the attention of the Company's developers. These do not need social computing and can be options or preferences that users can interact with at their own request or need. Sometimes these are also easy to implement. We also observed that adaptive UI can serve a supportive function for machine learning too. User-initiated personalization can be simple to set up, but by tracking and learning from user's actions we can begin creating new social data to add deeper intelligence to features at a later date, such as when the data gets rich enough to support new recommendation engines. All of this underlines that personalization is multifaceted, and investigations that are overly focused on technical aspects, or on the perceptions of users, can miss the bigger picture.

Other interesting themes that may be valuable for future investigations were around explainability versus non-explainability, and that sometimes explainability is an abstract concept disconnected from the actual functioning of an algorithm. For example, in several examples with the Company, two independent algorithms were created for one feature: one to drive accurate recommendations, and another to give a user a sense of understanding of why it 'may' have made the choice it made, or to give users the choice to improve, disregard, manipulate and enhance the results an algorithm has given them. It is also worth exploring how the more layers of explainability that are added to an algorithmic process, the more opaque it can actually become because both the difficulty in implementation and the difficulty in understanding can intensify.

7.9 Conclusion

Personalization research, in recognizing the ever-changing nature of its social dimensions, has begun to focus more on processes, and increasingly knowledge strategies (McMahon et al., 2004; Scheepers et al., 2004; Sudhindra et al., 2017). Personalization seen as a solution (Kim and Slotegraaf, 2016). A 'learning by doing' approach is supported where knowledge can both be personalized and codified as design progresses. (Bermell-Garcia et al., 2012). Personalization has become a way to think about services and those who use them, rather than a top-down set of policy prescriptions from older expert-dominated systems (Needham, 2011). It is about starting with the user first. Key elements in the successful design of these services therefore include good communication about key needs. This research project involved carefully observing this process across one and a half years of technical development of an AI company. 34 cases of personalization were identified and mapped across dimensions of developer perceptions of user needs, design, and implementation. In particular, dimensions of complexity in implementation and complexity in understanding proved valuable for identifying different negotiation strategies for solving personalization challenges. Having compared AI-mediated personalization (systems-initiated) and adaptable interfaces (enabling user-initiated personalization), key differences emerge. A multidimensional view of personalization proved sensible given personalization is being increasingly implemented in diverse ways across organizations, especially in light of the great power offered by emerging social tools. These tools, however, will require unique skillsets.

From the literature on technology and organizations there has been much debate about whether or not technology is merely a prop in our human-centered experiences, or at the other extreme that technology is deterministically independent. Sociomateriality offers a more nuanced way to understand the implementation of technologies like those that support personalization. Interestingly, when it comes to personalization features that have negotiations categorized as human-to-human in their nature, technology very much does play the role of a prop in the backstage of our attention. But there are problems where the material features of technology take center-stage. As we increasingly adopt tools that incorporate computation over complex social data, we will increasingly see technology play a leading role alongside the human actors. Our 'understanding' of their functionality in terms of making complex recommendations will increasingly give way to 'interpreting' it. This will lead to advances in the strategies we employ to negotiate with the technologies, users, and developers towards delivering more personalized services.

Chapter 8: Conclusion

This thesis is about a journey to build a deeper understanding of AI-mediated personalization. Specifically, two core questions inspired the papers that make up chapters 5, 6, and 7.

1. In what ways do technologies, especially AI, and personalization influence each other?
2. Are these personalization efforts utilizing social technologies leading to new organizational actors? If so, in what ways?

8.1 Bringing the research together

Chapter 2 summarized how personalization has been shaped by management logics that focus on standardization. New approaches, especially technologies, have unlocked the ability to deliver deeper personalized experiences for customers and citizens. In chapter 3 it was underscored that to effectively understand technologies and organizations, we need to utilize measurement strategies that are sensitive to users as well as sensitive to seemingly deterministic properties of ICTs. This led to the adoption of interactive mapping methods, starting code-first but also shaped by sociomaterial notions of *designer perceptions* about what *technologies can or cannot afford*. Using these approaches, an intimate opening-up of the design process revealed important themes for personalization. The first paper in chapter 5 followed a government who sought to use personalization to drive improved organizational capacity, specifically by changing the way in which new employees can be scouted and recruited. This was used to understand the first research question. In what ways are personalization and emerging ICTs influencing each other? The research in this paper traced the adoption of increasingly sophisticated ICTs in the pursuit of mediating user requests for a personalized experience on the HR platform. The use of natural language processing and machine learning could facilitate this experience better than traditional technologies. That is, the pursuit of personalization appeared to drive the adoption of algorithms that utilized social data, and these algorithms were being deployed in a way that had increasing autonomy.

However, as the black box of AI design was opened up, two important revelations persisted. This became the focus of the paper in chapter 6. First, despite being able to carefully retrace every algorithm and every design decision, certain digital artefacts remained inscrutable to even the most experienced designers. This was because of the adoption of dynamic and ever-changing data, as well as the use of algorithms that do not inherently return 'reasons' for their output. Nonetheless, users kept pressing upon the designers the question, 'why has it made the decision that it made'. Over time, new interfaces were added that could give the end-user greater ability to 'guess', but this implied that a deep and full understanding of the 'reasons' for some of the underlying processes escaped all parties. Thus, pushes for greater trust in the tool 'feeling right' for end-users led to processes that are often inscrutable, algorithms on top of algorithms. Explanation was replaced with interpretation, and at times even interpretations of the 'reasons' escaped the designers. The second recurring revelation is that in the design of these inscrutable algorithms there are many times in which designer decisions significantly influence the way the algorithms mediate specific experiences. Opportunities for designer-bias to be overlooked or even amplified by algorithms increased as more layers of inscrutability are added. As these systems gain more unchecked autonomy, these risks can become alarming.

This led to the third paper in chapter 7. How generalizable are these observations? Instead of investigating a single case of algorithms, like the HR system, the work in this chapter allowed for a cross-project look at 34 different types of personalization algorithms. Interactive mapping was used to trace issues of complexity across each of these features, half of which were traditional applications of ICTs and the other half of which utilized emerging capabilities like natural language processing and machine learning. Complexity was operationalized as, 'difficulty to build' and 'difficulty to explain why it made the decisions it made'. A system or algorithm that is difficult to build does not always mean that its behaviour or decisions are difficult to explain. Equally, a system or algorithm that is simple to build does not always mean that its behaviour or decisions are simple to explain. Some applications of personalization are complex in that different interfaces should be given only under certain circumstances for instance. Building these interfaces involves complex conditional logic. However, after being built, even if it took many engineer hours, these systems were easy to retrace because no decision deviated from tightly coupled and retraceable logic. Other algorithms were often easy to set up, including AI. Sometimes overnight. But when asking developers to explain why it results in outcomes that it does, answers were elusive. Across the 34 cases, emerging ICTs like AI tended to be harder to explain but not always hard to implement. Negotiation strategies used for building and tuning emerging ICTs was also different than traditional ICTs, where dynamic social data and inscrutable algorithms resulted in designers making guesses and interpretations about why it performed the way it did. Not only were there practical observations about how to handle personalization that is mediated by emerging ICTs, but issues of designer bias proved consistent across the 34 types of personalization. Bias issues were more pronounced when using emerging ICTs compared to traditional ICTs because the former were given greater autonomy to make decisions despite their inscrutability.

This algorithmic autonomy was possible because these algorithms utilize learning over social data. Hard-coded logic that is tightly coupled on conditional 'if' and 'then', which has no discretion, can only do so much for personalized experiences, and end up becoming challenging 'to get built'. Whereas emerging algorithms using natural language processing or machine learning are often elegant and easy to implement, and can make inferences based on dynamic data 'out of box'. This inferential ability means designers can take a step back and give decision-making authority over to these systems. However, this seems to imply a link between emerging ICT inscrutability and autonomy. To teach machines to be able to make guesses about users means to lose the ability to directly re-trace why it made decisions, because to infer decisions about users means to infer over personalized data that is ever-changing and individuated, and often requires turning to algorithms that while powerful, do not reveal their inner logic despite being seemingly deterministic tools shaped by coding and mathematics.

Research Question 1. In what ways do technologies, especially AI, and personalization influence each other?

Speaking to the first research question, findings in this thesis reveal that desires to adopt deeper personalization is leading to the adoption of emerging technologies like AI. A government that set as a national goal the improvement of their leadership desired a personalized platform that could help their HR managers find talented employees that meet HR manager needs at any given time. To build such a system meant having ICTs sophisticated enough to make inferences about people. Thus, the desire for personalization led to the adoption of emergent ICTs tools like natural language processing, because these could allow for employee profiles to be converted to a computer-readable format, then a matching algorithm could allow HR managers to receive a list of recommended employees based on a search query that they typed. Machine learning could then learn from HR manager interactions with these recommendations. Moving beyond this single case, across 34 total cases of personalization built by the AI company, attempts to solve the most complex user problems involved turning to other sophisticated algorithms like prediction, clustering, feature selection, and more. These could allow for autonomous decisions to be made about users and their needs in a live and self-service way, where the human designers and traditional human service delivery agents could take a passive role or have their role disintermediated. To teach machines to make decisions about users means to adopt machines that utilize social learning.

Research Question 2. Are these personalization efforts utilizing social technologies leading to new organizational actors? In what ways?

The adoption of personalization that is facilitated by emerging technologies like natural language processing and machine learning is resulting in a transfer of decision-making autonomy from individuals to ICTs. From an Actor-Network Theory perspective, ICTs are becoming actors of significance in terms of the way they mediate services, because they are directly enabling/distorting service options for users, make contextually-relevant recommendations, and learn from users for future interactions. It has been instrumental to consider technologies as more than just tools in the backdrop of human interactions. Technologies have properties that set them apart form other institutional artefacts because they are necessarily built from a logic of deterministically coupled logic chains. However, measuring why technologies get adopted or not has proven challenging when using user-centric or technology-centric methodologies. Sociomateriality calls for measurement approaches that bring both users and their subjective social constructions together with technologies and their objective forms. Specifically, tracing human perceptions of what objects can or cannot do helps bridge both worlds. Using an interactive mapping approach built from these perspectives, emerging technologies when used to drive personalization are emerging as important actors.

Actor-Network Theory suggests we need to be sensitive to all actors in a value-network, be they human or ICT. We can differentiate actors in this network based on the nature of their influence on the value-exchange and intermediation of experience. For example, humans have discretion and subjective desires and therefore are important actors in a value chain because they can change, interpret or reconfigure the chain. However, other times humans are constrained to minimal discretion in a certain exchange, leaving much of the experience to be mediated by technology. This thesis continues this dialogue. By intimately unpacking the design process of personalization artefacts that are mediated by emerging technologies versus traditional technologies, we are witnessing the rise of a new actor in our value-networks that can make decisions without a human involved. This does not happen in a straight forward fashion or without flaw. This research reveals that not only are these actors of consequence in many of our value-networks, but the process of building them reveals opportunities for designers to overlook aspects of bias and other risks like privacy. This becomes amplified if the algorithms utilized are inscrutable. The findings in this thesis also point to a relationship between calls for more explainability and interactivity by users sometimes leading, paradoxically, to less explainability and to the adoption of inscrutability because these systems contain outputs or decisions that are hard to explain and have inner workings that are difficult, if not impossible to understand. As autonomous ICT actors take on more responsibility, there is a growing need to be mindful and conscientious of their design. Where inscrutability remains persistent, giving a roadmap of decisions can increase transparency or at least allow for risky decisions to be traced.

8.2 Contributions

The primary aim of this thesis is theory development. What began as a descriptive process to uncover the relationship between personalization and emerging ICTs led to a mid-range theory that proved stable across 34 personalization cases built across 4 projects. The pursuit of personalization leads to the adoption of emerging ICTs, as social learning capabilities prove more effective than traditional applications of ICT. Specifically, they can make autonomous inferences about users. This autonomous inference is sought after by designers of personalization. This work also showed that the drive to adopt emerging ICT-mediated personalization is leading to a new and consequential actor in the workplace. As autonomous actors are developed to make inferences about individuals, value-chains are disintermediating human actors. Yet, despite this, very-human decisions made during the design stage introduce risks such as privacy and bias amplification that can lead to autonomously reinforced risk. This is amplified by the fact these actors make their decisions over increasingly dynamic data and are often unable to reveal hints about why they made the decisions they do. Thus, inscrutable and autonomous actors are among us. Pursuits of personalization will likely lead to more of these rather than less.

A secondary aim of this thesis includes advancing the personalization literature. This literature has been found to be largely divided between user-centric and technology-centric methodologies. Drawing from advances in understanding the interactive relationship between subjective humans and functionally enclosed ICTs that come from organization and management literatures, sociomateriality offered a way to bridge the divide in personalization. By looking at code-first, and then turning to designer perceptions about why decisions were made and what exactly the algorithms do, an intimate story unfolded. This thesis has the practical contribution of advancing the personalization literature by better understanding the adoption of emerging ICTs compared to traditional ICTs, and this led to insights about quick wins, negotiation strategies, risk awareness and risk mitigation.

A second practical contribution of this work was the advancement of interactive mapping as a method for unpacking the design process. Following the code from a user-perspective and triangulating it with designer insights proved rewarding. Li and Jagadish (2014) use interactive mapping to trace and visualize complex architectures, but combined it with experimental design. This thesis shows that this code-first approach can also be combined with mixed-methods including turning to developer documentation as a secondary source and interviewing key developers as a primary source. This approach has a number of advantages. It helps researchers orientate 'where to begin' when retracing complex code. It can help interrogate decisions made throughout, including over time. This can expose gaps in decision-making and understanding. This combined with questions of design complexity to effectively map out all personalization algorithms built by a company over the course of 4 projects. Approaches like this may be more relevant than ever as a need for awareness about risk grows as these non-human consequential autonomous actors become even more ubiquitous.

8.3 Limitations

For two papers the single-case view was restricted to a single family of features and algorithms, as complex as they are. Caution was needed when generalizing from these experiences. To mitigate this, the third paper aimed for theory-testing, specifically around the generalizability of some of the findings about persistent inscrutability, and bias amplification. As seen in Table 1 from chapter 4, the developers had a wide range of skills and experience. But even if skills were diverse, organizational cultures as well as other idiosyncrasies are hard to control for. A next stage of this research should be extended to a different company, perhaps a large enterprise instead of a small shop. This would help further generalize the findings.

Another set of challenges is even more important to highlight, challenges that are related to a theme that emerged throughout this thesis. That of bias amplification. It is acknowledged that the physical affordances of an object, or what it can enable or constrain, is both determined by physical properties but also by how these properties are observed by subjective humans. Throughout this work, designers and end-users were found to make judgements in a hurry or without full awareness of 'why' something was happening. Other times, inscrutable systems were welcomed, but awareness of their inscrutability varied. Sometimes there was evidence of this awareness during the design, other times awareness did not emerge until interrogated by the thesis author during the interactive mapping exercises. This suggests that bias remained an ever-present factor. As highlighted in chapter 4, strategies were employed to minimize author amplified bias. The author distanced themselves from the design of algorithms, allowing for an interrogation from a less embedded perspective. Two additional researchers played a critical role in this interrogation, bringing non-programmer perspectives into the analysis.

This research did not aim to objectively measure whether the AI company was building optimal systems. This research does underscore the pervasiveness of bias however. Future research could benefit from extending these findings about complexity and risk into design thinking literatures, with a deeper scrutiny of these decisions. Another logical extension would be to connect this work to emerging industry, regulatory and academic calls for understanding algorithmic transparency.

8.4 Discussion: Future work

This research intended to explore the boundaries of a new actor. This machine actor appears increasingly capable of mechanized knowing, and often in ways we cannot comprehend or explain. Despite not comprehending their logic with respect to specific decisions, we are increasingly affording them space in transforming our workplace and to make inferences for us. So long as these mechanical learning tools are used to understand people, personalization will be a practical tool for organizations and a field of study. As we better understand the boundaries between humans, socially-trained machines, and our collective interactions, we will be able to continue to expand mechanized learning. Personalization will increasingly influence the nature of our work.

Behavioural value reinvestment and conscientious design

Behavioural data poses great risks for the public if improperly used, but not all use of behavioural data is predatorial. We will conclude with an extended discussion about a future research space that can build from this thesis: conscientious design, which can inform systems design thinking for example. Building from the findings of this thesis, conscientious design involves awareness of algorithmic explainability by understanding if algorithms are inscrutable or not. This can be facilitated by interactive mapping. If there are aspects of an algorithm that are not understandable, steps can be taken to review their interpretability. Designers can also promote algorithmic transparency through a range of strategies, to mitigate negative impacts of inscrutability. Also, designers should be mindful to the ways algorithms or human design decisions amplify issues of bias.

Understanding users allows designers of services to modify features, improve results, and delight customers. This is deeper individuation, and AI-mediated personalization can be powerful for this. As seen in Figure 8.1, users generate data which can be rendered into analytics that can provide meaningful insights for service improvements. Data can include customer satisfaction surveys, feedback, complaints, clicks, time spent in a page, and more. Along the way, extra data may be collected than needed. This exhaust should be treated as such, data not needed for the service improvement or service personalization. This is data that may have privacy or security implications. However, it is not as simple as deleting unused data. It is challenging to predict which data will be valuable for service improvement and which will not. This calls for diligence and conscientiousness on the part of the service designers. A properly managed system of behavioural data for the purpose of service improvement can enable a *behavioural value reinvestment cycle* (Zuboff, 2019). This aligns with earlier discussions about human-machine reconfiguration. This discussion alludes to the need for conscientious human-machine reconfiguration when in the pursuit of AI-mediated personalization.

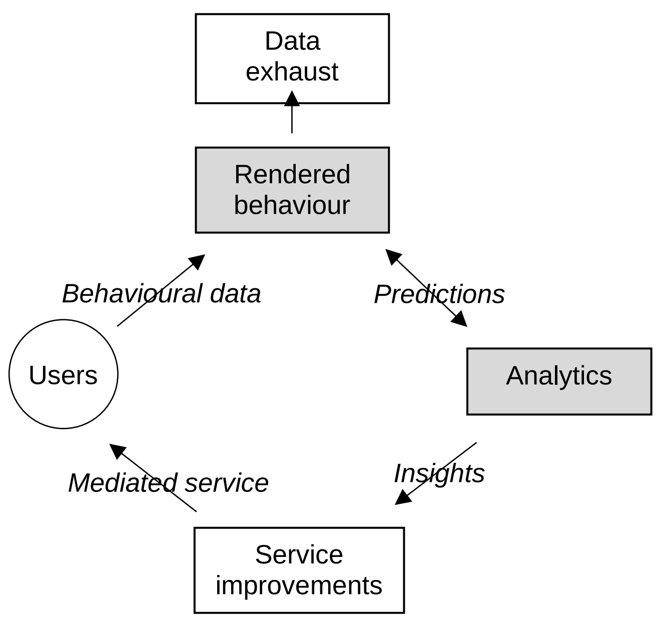

Figure 8.1. Based on Zuboff's behavioural value reinvestment cycle

It was this data exhaust that titans like Google identified as being an invaluable source of *surplus behavioural data.* This has incentivized the collection of more than just clicks when using a service, but all other sites visited, passwords, financial information, location data, phone data, email content, and more. This is not limited to the largest of digital firms. In 2017 an investigation into third party applications found of the top 1000 in China, 822 were found launching other applications for the purpose of collecting data (Xu et al., 2017). Conscientious service designers can conceptualize and even map the degree to which the data they are using cycles back to service improvement, and how to manage surplus data which becomes exhaust. This helps tackle issues of privacy. This is not being done enough.

Chapter 7 extended interactive mapping across 34 cases of personalization. This approach helped map algorithms by explainability and interpretability, as well as interrogate why they were built and negotiation strategies when tuning them. Like being mindful of data exhaust, approaches like this can help organizations be mindful of design implications and algorithmic complexity. The Canadian Government calls this *algorithmic impact* and as of the writing of this thesis, self-assessing this impact is a requirement for internal venders that utilize algorithms (Government of Canada, 2020).

Managers should seek to improve the quality of services to their customers, clients, and citizens, and therefore ought to pursue the advantages of technologies in promoting personalized service experiences, but must also be aware of the data it generates and its risks, as well as the bias they are introducing as well. We have summarized some of the risks throughout this thesis. Some negotiation strategies have emerged from the data in Chapters 5, 6 and 7. But with the technology so young, the affordances that unlocked them so recent, and a build-up of social data at a rate previously inconceivable, issues of bias, explainability and privacy will be emergent and dynamic.

We are already seeing evidence of conscientiousness on the part of the public sector (Chapter 5) and the developers are well aware of the lack of explainability of some systems (Chapter 6). But one thing is worth underlining. Even if ethics issues will drive a demand from users, citizens and policy-makers for greater explainability, the use of social data suggests we will not remove the black box because diverse, asynchronous, and subjective data makes many algorithms contextually-dependent and more opaque. If anything, we are slowly accepting the implications, both good and bad, of these new technologies and how they will change the workplace. Some aspects of this we will understand, and some aspects we will not. Strategies are emerging for increasing our ability to interpret the mechanization of learning. In the process, we are recognizing an actor that will become a ubiquitous part of our workplace. This thesis is a first step towards a longer-term research vision. How can these approaches be combined to optimize the advantages of learning from non-explainable technologies, while mitigating their challenges? What techniques can we use to remain aware, or approximate 'why' an AI made decisions about a person that it did? What implications will this have for an increase in trust and adoption for businesses and governments of more advanced personalization techniques?

Social constructivist researchers have offered deeper observation of social actors, processes and images, including dynamics of conflict, consensus, and choices. But all of this offers no specific judgement on meaning or purpose. It has been argued that when social constructivists opened up the black box of social experience, they found it empty of specific meaning (Winner, L. 1993). This thesis contributes to this dialogue by showing the process of 'opening up' the black box of new decision-making actors, including machine learning. While these technological actors have distinct contributions to conflict, consensus and choice compared to human actors, we nonetheless find an equally 'meaningless' box in that we increasingly cannot understand or provide judgement around 'how', or 'why' it does what it does. Chapter 6 shows an increasingly inexplainability regarding decision-making reasoning, and Chapters 5, 6, and 7 show how increasingly dynamic the data being used by machines is becoming. Chapter 7 shows that some of the interfaces we adopt rely on more rigid and less fluid data, like hard-coding logic, and shows how other algorithms are making use of complex and ever-changing social data and are sometimes considered black boxes. But what is interesting is that we are inviting into our organizational decision-making a myriad of inter-connected technologies that vary in terms of social embeddedness, inexplainability, complexity in design, and autonomy. These new social algorithms used for personalization are important because they do influence, distort, transform, and reflect our knowledge about users, our mediation of personal user experiences, and more.

What is more however, this research shows awareness of risk needs to go beyond the algorithms themselves. Humans are making decisions throughout the process, sometimes deliberated and other times innocuous, but ever significant. Why one algorithm and not another? Why one parameter and not another? Why one interface or value or decimal place or data type but not another? What implications would changing these have? If we are increasingly adopting inscrutable and autonomous systems into our value-chains, simply 'looking at the code' is no longer enough to understand how and why they behave the way they do. We may be accepting ICT actors that may never reveal fully their logic. This explains repeated and growing calls for transparency. If we cannot say why they made the decisions they did, we at least should be able to retrace relevant design decisions that were made.

References

42nd Parliament of Canada (2019) Standing Committee on Access to Information, Privacy and Ethics. Available at: https://www.ourcommons.ca/DocumentViewer/en/42-1/ETHI/meeting-141/evidence.

Abdelnour S, Hasselbladh H and Kallinikos J (2017) Agency and Institutions in Organization Studies. *Organization Studies*: 017084061770800. DOI: 10.1177/0170840617708007.

Abdul A, Vermeulen J, Wang D, et al. (2018) Trends and trajectories for explainable, accountable and intelligible systems: An HCI research agenda. *Conference on Human Factors in Computing Systems - Proceedings* 2018-April: 1–18. DOI: 10.1145/3173574.3174156.

Aberbach JD and Christensen T (2005) Citizens and consumers: An NPM dilemma. *Public Management Review* 7(2): 225–245. DOI: 10.1080/14719030500091319.

Abilash Reddy C and Subramaniyaswamy V (2015) An enhanced travel package recommendation system based on location dependent social data. *Indian Journal of Science and Technology* 8(16). DOI: 10.17485/ijst/2015/v8i16/63571.

Abu-Dalbouh H (2016) Implementing End-User Privacy through Human Computer Interaction for Improving Quality of Personalized Web. *Computer and Information Science* 9(1): 75. DOI: 10.5539/cis.v9n1p75.

Adadi A and Berrada M (2018) Peeking Inside the Black-Box: A Survey on Explainable Artificial Intelligence (XAI). *IEEE Access* 6. IEEE: 52138–52160. DOI: 10.1109/ACCESS.2018.2870052.

Adolphs C and Winkelmann A (2010) Personalization Research in E-Commerce – A State of the Art Review (2000-2008). *Journal of Electronic Commerce Research* 11(4): 326–341. Available at: http://www.jecr.org/sites/default/files/11_4_p05.pdf.

Adomavicius G and Tuzhilin A (2005) Personalization technologies: A process-oriented perspective. *Communications of the* ACM 48(10): 83–90. DOI: 10.1145/1089107.1089109.

Agrawal A, Gans JS and Goldfarb A (2019) Artificial intelligence: The ambiguous labor market impact of automating prediction. *Journal of Economic Perspectives* 33(2): 31–50. DOI: 10.1257/jep.33.2.31.

Aguirre E, Roggeveen AL, Grewal D, et al. (2016) The personalization-privacy paradox: implications for new media. *Journal of Consumer Marketing* 33(2): 98–110. DOI: 10.1108/JCM-06-2015-1458.

Ahn HJ, Kang H and Lee J (2010) Selecting a small number of products for effective user profiling in collaborative filtering. *Expert Systems with Applications* 37(4). Elsevier Ltd: 3055–3062. DOI: 10.1016/j.eswa.2009.09.025.

Ahsen ME, Ayvaci MUS and Raghunathan S (2019) When algorithmic predictions use human-generated data: A bias-aware classification algorithm for breast cancer diagnosis. *Information Systems Research* 30(1): 97–116. DOI: 10.1287/isre.2018.0789.

Alaimo C and Kallinikos J (2017) Computing the everyday: Social media as data platforms. *Information Society* 33(4): 175–191. DOI: 10.1080/01972243.2017.1318327.

Alaimo C, Kallinikos J and Valderrama E (2020) Platforms as service ecosystems: Lessons from social media. *Journal of Information Technology* 35(1): 25–48. DOI: 10.1177/0268396219881462.

Alford J (2008) The Limits to Traditional Public Administration , or Rescuing Public Value from Misrepresentation. 67(3): 357–366. DOI: 10.1111/j.1467-8500.2008.00593.x.

Alford J (2016) Co-Production, Interdependence and Publicness: Extending public service-dominant logic. *Public Management Review* 18(5). Routledge: 673–691. DOI: 10.1080/14719037.2015.1111659.

Amershi S, Cakmak M, Knox WB, et al. (2014) Power to the People: The Role of Humans in Interactive Machine Learning. *AI Magazine* 35(4): 105. DOI: 10.1609/aimag.v35i4.2513.

Andrews L (2019) Public administration, public leadership and the construction of public value in the age of the algorithm and 'big data'. *Public Administration* 97(2): 296–310. DOI: 10.1111/padm.12534.

Anshari M, Almunawar MN, Lim SA, et al. (2019) Customer relationship management and big data enabled: Personalization & customization of services. *Applied Computing and Informatics* 15(2). King Saud University: 94–101. DOI: 10.1016/j.aci.2018.05.004.

Arazy O (2015) Personalityzation: UI Personalization, Theoretical Grounding in HCI and Design Research. 7(2): 43–69.

Arnold M (2003) On the phenomenology of technology: The 'Janus-faces' of mobile phones. *Information and Organization* 13(4): 231–256. DOI: 10.1016/S1471-7727(03)00013-7.

Arora N, Dreze X, Ghose A, et al. (2008) Putting one-to-one marketing to work: Personalization, customization, and choice. *Marketing Letters* 19(3–4): 305–321. DOI: 10.1007/s11002-008-9056-z.

Arthur BW (2009) *The Nature Oftechnology, What It Is and How It Evolves*. London: Allen Lane.

Attewell P and Rule J (1984) Computing and Organizations: What We Know and What We Don't

Know. Available at: http://aisel.aisnet.org/icis1984%0Ahttp://aisel.aisnet.org/icis1984/17.

Awad NF and Krishnan MS (2006) The personalization privacy paradox: An empirical evaluation of information transparency and the willingness to be profiled online for personalization. *MIS Quarterly: Management Information Systems* 30(1): 13–28. DOI: 10.2307/25148715.

Bagate RA and Suguna R (2020) Different Approaches in Sarcasm Detection: A Survey. *Lecture Notes on Data Engineering and Communications Technologies* 38: 425–433. DOI: 10.1007/978-3-030-34080-3_48.

Bain R (1937) Technology and state government. *American Sociological Review* 2(6): 860–874.

Barley SR (1986) Technology as an occasion for structuring: Evidence from observations of CT scanners and the social order of radiology departments. *Administrative Science Quarterly* 31(1): 78–108. DOI: 10.2307/2392767.

Barocas S and Selbst AD (2014) Big Data ' S Disparate Impact. *California Law Review* 104: 671–732.

Barredo Arrieta A, Díaz-Rodríguez N, Del Ser J, et al. (2020) Explainable Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. *Information Fusion* 58: 82–115. DOI: 10.1016/j.inffus.2019.12.012.

Barrett M, Oborn E, Orlikowski WJ, et al. (2012) Reconfiguring boundary relations: Robotic innovations in pharmacy work. *Organization Science* 23(5): 1448–1466. DOI: 10.1287/orsc.1100.0639.

Baumgartner FR and Jones BD (1991) Agenda Dynamics and Policy Subsystems. *The Journal of Politics* 53(4): 1044–1074.

Becker H (1996) The epistemology of qualitative research. In: Jessor R, Colby A, and Shweder R (eds) *Essays on Ethnography and Human Development*. Chicago University Press, pp. 53–71.

Bennett Moses L and Chan J (2018) Algorithmic prediction in policing: assumptions, evaluation, and accountability. *Policing and Society* 28(7). Taylor & Francis: 806–822. DOI: 10.1080/10439463.2016.1253695.

Bentley T and Wilsdon J (2003) *The Adaptative State. Strategies for Personalising the Public Realm*. London: Demos.

Berg M (1997) Of forms, containers, and the electronic medical record: Some tools for a sociology of the formal. *Science Technology and Human Values* 22(4): 403–433. DOI: 10.1177/016224399702200401.

Bermell-Garcia P, Verhagen WJC, Astwood S, et al. (2012) A framework for management of Knowledge-Based Engineering applications as software services: Enabling personalization and codification. *Advanced Engineering Informatics* 26(2). Elsevier Ltd: 219–230. DOI: 10.1016/j.aei.2012.01.006.

Bernstein F, Modaresi S and Sauré D (2019) A dynamic clustering approach to data-driven assortment personalization. *Management Science* 65(5): 2095–2115. DOI: 10.1287/mnsc.2018.3031.

Biran O and Cotton C (2017) Explanation and Justification in Machine Learning: A Survey. *IJCAI Workshop on Explainable Artificial Intelligence* (XAI): 8–13. DOI: 10.1108/13563281111156853.

Bjiker WE (1995) *Of Bicycles, Bakelites and Bulbs: Toward a Theory of Sociotechnical Change*. Cambridge: MIT Press.

Blom JO and Monk AF (2003) Theory of personalization of appearance: Why users personalize their PCs and mobile phones. *Human-Computer Interaction* 18(3): 193–228. DOI: 10.1207/S15327051HCI1803_1.

Bloom N, Garicano L, Sadun R, et al. (2014) The distinct effects of information technology and communication technology on firm Organization. *Management Science* 60(12): 2859–2885. DOI: 10.1287/mnsc.2014.2013.

Bloomfield B and Vurdubakis T (2001) *The Vision Thing: Construction of Time and Technology in Management Advice. Critical Consulting: Perspectives on the Management Advice Industry*. Oxford: Blackwell.

Boddy CR (2016) Sample size for qualitative research. *Qualitative Market Research* 19(4): 426–432. DOI: 10.1108/QMR-06-2016-0053.

Bodoff D and Ho SY (2014) The effects of Time and number of Personalized items on Users' Amount of Sampling. In: ECIS, 2014.

Bouvier G and Machin D (2018) Critical discourse analysis and the challenges and opportunities of social media. *Review of Communication* 18(3): 178–192. DOI: 10.1080/15358593.2018.1479881.

Bovaird T and Loeffler E (2012) From Engagement to Co-production : The Contribution of Users and Communities to Outcomes and Public Value. DOI: 10.1007/s11266-012-9309-6.

Bowker G and Star SL (2002) *Sorting Things out: Classification and Its Consequences*. MIT Press.

Bozeman B and Sarewitz D (2011) Public Value Mapping and Science Policy Evaluation. (February): 1–

23. DOI: 10.1007/s11024-011-9161-7.

Bragge J, Relander S, Sunikka A, et al. (2007) Enriching literature reviews with computer-assisted research mining. Case: Profiling group support systems research. *Proceedings of the Annual Hawaii International Conference on System Sciences*: 1–10. DOI: 10.1109/HICSS.2007.209.

Brandsen T and Pestoff V (2006) Co-production, the third sector and the delivery of public services. *Public Management Review*. DOI: 10.1080/14719030601022874.

Brynjolfsson E and McAfee A (2016) *The Second Machine Age: Work, Progress, and Prosperity in a Time of Brilliant Technologies*. Norton & Company.

Bryson JM, Crosby BC and Bloomberg L (2014) Public Value Governance: Moving beyond Traditional Public Administration and the New Public Management. *Public Administration Review* 74: 445–456. DOI: 10.1111/puar.12238.Public.

Burrell J (2016) How the machine 'thinks': Understanding opacity in machine learning algorithms. *Big Data and Society* 3(1): 1–12. DOI: 10.1177/2053951715622512.

Capuano N, D'Aniello G, Gaeta A, et al. (2015) A personality based adaptive approach for information systems. *Computers in Human Behavior* 44: 156–165. DOI: 10.1016/j.chb.2014.10.058.

Chang CC, Chen PL, Chiu FR, et al. (2009) Application of neural networks and Kano's method to content recommendation in web personalization. *Expert Systems with Applications* 36(3 PART 1). Elsevier Ltd: 5310–5316. DOI: 10.1016/j.eswa.2008.06.067.

Chang HH and Chen SW (2008) The impact of customer interface quality, satisfaction and switching costs on e-loyalty: Internet experience as a moderator. *Computers in Human Behavior* 24(6): 2927–2944. DOI: 10.1016/j.chb.2008.04.014.

Chapman J (2003) Public value - The missing ingredient in reform? In: Bentley T and Wilsdon J (eds) *The Adaptative State. Strategies for Personalising the Public Realm*. Demos.

Che T, Peng Z, Lim KH, et al. (2015) Antecedents of consumers' intention to revisit an online group-buying website: A transaction cost perspective. *Information and Management* 52(5). Elsevier B.V.: 588–598. DOI: 10.1016/j.im.2015.04.004.

Chellappa, Ramnath K and Sin, Raymond G (2005) Personalization versus Privacy: An empirical examination of the online consumer's dilemma. *Information Technology and Management* 6: 181–202.

Chen DN, Hu PJH, Kuo YR, et al. (2010) A Web-based personalized recommendation system for mobile phone selection: Design, implementation, and evaluation. *Expert Systems with Applications* 37(12). Elsevier Ltd: 8201–8210. DOI: 10.1016/j.eswa.2010.05.066.

Ciborra CU (1996) *Teams, Markets and Systems*. Cambridge University Press.

Ciborra CUCU and Hanseth O (1998) From tool to Gestell: Agendas for managing the information infrastructure. *Technology* 11(4): 305–327.

Cohen M, March JG and Olsen JP (1972) Garbage Can Model of Organizational Choice. *Administrative Science Quarterly* 17(1): 1–2. DOI: 10.1002/9781118785317.weom110102.

Cordella A and Bonina CM (2012) A public value perspective for ICT enabled public sector reforms: A theoretical reflection. *Government Information Quarterly* 29(4). Elsevier B.V.: 512–520. DOI: 10.1016/j.giq.2012.03.004.

Cordella A and Iannacci F (2010) Information systems in the public sector: The e-Government enactment framework. *Journal of Strategic Information Systems* 19(1). Elsevier B.V.: 52–66. DOI: 10.1016/j.jsis.2010.01.001.

Cordella A and Paletti A (2019) Government as a platform, orchestration, and public value creation: The Italian case. *Government Information Quarterly* 36(4). Elsevier: 101409. DOI: 10.1016/j.giq.2019.101409.

Cordella A and Tempini N (2015) E-government and organizational change : Reappraising the role of ICT and bureaucracy in public service delivery. *Government Information Quarterly* 32(3). Elsevier Inc.: 279–286. DOI: 10.1016/j.giq.2015.03.005.

Crawford K (2017) The Trouble With Bias. In: *Keynote at Neural Information Processing Systems*, 2017.

Cui R, Li J and Zhang DJ (2020) Reducing discrimination with reviews in the sharing economy: Evidence from field experiments on airbnb. *Management Science* 66(3): 1071–1094. DOI: 10.1287/mnsc.2018.3273.

Daft RL and Lengel RH (1986) ORGANIZATIONAL INFORMATION REQUIREMENTS, MEDIA RICHNESS AND STRUCTURAL DESIGN. *Management Science* 32(5): 554–571.

Danner-Schröder A and Geiger D (2016) Unravelling the motor of patterning work: Toward an understanding of the microlevel dynamics of standardization and flexibility. *Organization Science* 27(3): 633–658. DOI: 10.1287/orsc.2016.1055.

Davis SM (1987) *Future Perfect*. Addison-Wesley.

de Laat PB (2018) Algorithmic Decision-Making Based on Machine Learning from Big Data: Can Transparency Restore Accountability? *Philosophy and Technology* 31(4). Philosophy & Technology: 525–541. DOI: 10.1007/s13347-017-0293-z.

Demasio A (1994) *Descartes' Error: Emotion, Reason and the Human Brain*. Random House.

Devaraj S, Fan M and Kohli R (2006) Examination of online channel preference: Using the structure-conduct-outcome framework. *Decision Support Systems* 42(2): 1089–1103. DOI: 10.1016/j.dss.2005.09.004.

Devitt SK (2018) Trustworthiness of autonomous systems. In: Abbass HA, Scholz J, and Reid DJ (eds) *Foundations of Trusted Autonomy*, pp. 1–12. DOI: 10.1007/978-3-319-64816-3_1.

Dewett T and Jones GR (2001) *The Role of Information Technology in the Organization: A Review, Model, and Assessment*. DOI: 10.1016/S0149-2063(01)00094-0.

Diakopoulos N (2015) Algorithmic Accountability: Journalistic investigation of computational power structures. *Digital Journalism* 3(3). Routledge: 398–415. DOI: 10.1080/21670811.2014.976411.

Douglas L (2017) AI is not just learning our biases; it is amplifying them. *Medium*, December.

Eisenhardt KM (1989) Building Theories from Case Study Research. *Academy of Management Review* 14(4): 532–550. DOI: 10.5465/AMR.1989.4308385.

Fan H and Poole MS (2006) What is personalization? perspectives on the design and implementation of personalization in information systems. *Journal of Organizational Computing and Electronic Commerce* 16(3–4): 179–202. DOI: 10.1207/s15327744joce1603&4_2.

Faraj S, Pachidi S and Sayegh K (2018) Working and organizing in the age of the learning algorithm. *Information and Organization* 28(1). Elsevier: 62–70. DOI: 10.1016/j.infoandorg.2018.02.005.

Farhall K, Gibbons A, Wright S, et al. (2019) Political Elites ' Use of Fake News Discourse Across Communications Platforms The University of Melbourne , Australia The University of Texas at Austin , USA. *International Journal of Communication* 13: 4353–4375. Available at: https://ijoc.org/index.php/ijoc/article/view/10677.

Flyvbjerg B (2006) Five Misunderstandings About Case-Study Research. *Qualitative Inquiry* 12(2): 219–245. DOI: 10.1177/1077800405284363.

Ford M (2015) *The Rise of the Robots: Technology and the Threat of Mass Unemployment*. Simon and Schuster.

Fountain JE (2001) Enacting Technology. In: *Building the Virtual State: Information Technology and Institutional Change*. Brookings Institution Press.

Fuertes V and Lindsay C (2016) Personalization and Street-Level Practice in Activation: the Case of the Uk'S Work Programme. *Public Administration* 94(2): 526–541. DOI: 10.1111/padm.12234.

Furman J and Seamans R (2018) *AI an the economy*. Cambridge.

Gains BYF and Stoker G (2009) Delivering ' Public Value ': Implications for Accountability and Legitimacy. 62(3): 438–455. DOI: 10.1093/pa/gsp007.

Gajos KZ, Weld DS and Wobbrock JO (2010) Automatically generating personalized user interfaces with Supple. *Artificial Intelligence* 174(12–13). Elsevier B.V.: 910–950. DOI: 10.1016/j.artint.2010.05.005.

Galbraith JR (1974) Organization Design: An Information Processing View. *Interfaces* 4(3): 28–36. DOI: 10.1287/inte.4.3.28.

Gasser U and Almeida VAF (2017) A Layered Model for AI Governance. *IEEE Internet Computing* 21(6): 58–62. DOI: 10.1109/MIC.2017.4180835.

Gasson S (2011) Rigor in Grounded Theory Research. *The Handbook of Information Systems Research*: 79–102. DOI: 10.4018/978-1-59140-144-5.ch006.

Geertz C (1973) Thick Description: Toward an Interpretive Theory of Culture. In: Geertz C. (ed.) *In The Interpretation of Cultures: Selected Essays*. New York: Basic Books, pp. 3–30.

George AL and Bennett A (2005) *Case Studies and Theory Development in the Social Sciences*. MIT Press.

Gera DL (2003) *Ancient Greek Ideas on Speech, Language, and Civilization*. Oxford Scholarship.

Germonprez M and Hovorka DS (2013) Member engagement within digitally enabled social network communities: New methodological considerations. *Information Systems Journal* 23(6): 525–549. DOI: 10.1111/isj.12021.

Germonprez M, Hovorka D and Gal U (2011) Historical Synopsis of the Sami / United Nations Relationship A brief Saami history Saami Today. *Journal of the Association for Information Systems* 12(10): 662–683. Available at: http://www.thearctic.is/PDF/Synopsis of Sami-UN Relations PDF.pdf.

Gerring J (2004) What is a case study and what is it good for? *American political science review*.

Available at: http://journals.cambridge.org/abstract_S0003055404001182 (accessed 24 April 2017).

Gillespie T (2017) Algorithmically recognizable: Santorum's Google problem, and Google's Santorum problem. *Information Communication and Society* 20(1). Taylor & Francis: 63–80. DOI: 10.1080/1369118X.2016.1199721.

Glaser BG and Strauss AL (1973) *The Discovery of Grounded Theory : Strategies for Qualitative Research*. Aldine Pub.

Gobry P-E (2011) Eric Schmidt to World Leaders at EG8: Don't Regulate Us, or Else. *Business Insider*, 24 May. Available at: http://www.businessinsider.com/google-ceo-larry-page-wants-a-place-for-experiments-2013-5.

Goffman E (1958) Characteristics of total institutions. In: *Symposium on Preventive and Social Psychiatry*. US Government Printing Office, pp. 43–84.

Goffman E (1959) *The Presentation of Self in Everyday Life*. NY: Anchor Books.

Gold RL (1958) Roles in Sociological Field Observations. *Social Forces* 36(3): 217–23. DOI: 10.2307/2573808.

Goldin D, Smolka, A. S and Wegner P (1998) *Interactive Computation - The New Paradigm*. Springer.

Goldin D, Smolka, A. S and Wegner P (eds) (2006) *Interactive Computation: A New Paradigm*. Springer-Verlag Berlin.

Goodfellow I, Bengio Y and Courville A (2016) *Deep Learning*. 1st ed. MIT Press.

Government of Canada (2020) Algorithmic Impact Assessment Tool. Available at: https://www.canada.ca/en/government/system/digital-government/digital-government-innovations/responsible-use-ai/algorithmic-impact-assessment.html.

Grabowski M and Roberts K (1996) Organizations, technology and structuring. In: *Handbook of Organization Studies*. London: SAGE Publications, Inc., pp. 409–423.

Grimsley M and Meehan A (2007) e-Government information systems : Evaluation-led design for public value and client trust. (November 2006): 134–148. DOI: 10.1057/palgrave.ejis.3000674.

Guidotti R, Monreale A, Ruggieri S, et al. (2018) A survey of methods for explaining black box models. *ACM Computing Surveys* 51(5). DOI: 10.1145/3236009.

Gunning D and Aha DW (2019) DARPA's explainable artificial intelligence program. *AI Magazine* 40(2):

44–58. DOI: 10.1609/aimag.v40i2.2850.

Ha HY, Muthaly SK and Akamavi RK (2010) *Alternative Explanations of Online Repurchasing Behavioral Intentions: A Comparison Study of Korean and UK Young Customers*. DOI: 10.1108/03090561011032757.

Hacking I (1999) *The Social Construction of What?* Harvard University Press.

Hallinan B, Brubaker JR and Fiesler C (2020) Unexpected expectations: Public reaction to the Facebook emotional contagion study. *New Media and Society* 22(6): 1076–1094. DOI: 10.1177/1461444819876944.

Haraty M and Mcgrenere J (2016) Designing for Advanced Personalization in Personal Task Management.

Hendricks LA, Burns K, Saenko K, et al. (2018) Women Also Snowboard: Overcoming Bias in Captioning Models. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 11207 LNCS: 793–811. DOI: 10.1007/978-3-030-01219-9_47.

Herington C and Weaven S (2009) E-retailing by banks: e-service quality and its importance to customer satisfaction. *European Journal of Marketing* 43(9/10): 1220–1231.

Ho SY (2006) The Attraction of Internet Personalization to Web Users. *Electronic Markets* 16(1): 41–50. DOI: 10.1080/10196780500491162.

Holzinger A, Biemann C, Pattichis CS, et al. (2017) What do we need to build explainable AI systems for the medical domain? (Ml): 1–28. Available at: http://arxiv.org/abs/1712.09923.

Huber GP (1990) A Theory of the Effects of Advanced Information Technologies on Organizational Design, Intelligence, and Decision Making. *Academy of Management Review* 15(1): 47–71. DOI: 10.5465/amr.1990.4308227.

Huynh K, Benarivo J, Xuan CA, et al. (2021) Improving Human-Centric Software Defect Evaluation, Reporting, and Fixing. In: *2021 IEEE 45th Annual Computers, Software, and Applications Conference* (COMPSAC), 2021. DOI: 10.1109/COMPSAC51774.2021.00064.

Institute for Citizen-Centred Service (2012) *Citizens First* 6.

Institute for Citizen-Centred Service (2018) *Citizens First* 8. Toronto, Canada.

Isen AM and Patrick R (1983) The effect of positive feelings on risk taking: When the chips are down. *Organizational Behavior and Human Performance* 21(2): 194–202. DOI:

10.1056/NEJM198302033080530.

Jackson JR (2018) A perfict fit: Personalization versus privacy. *Journal of Organizational Psychology* 18(4): 33.

Jiang Y, Shang J and Liu Y (2010) Maximizing customer satisfaction through an online recommendation system: A novel associative classification model. *Decision Support Systems* 48(3). Elsevier B.V.: 470–479. DOI: 10.1016/j.dss.2009.06.006.

Jick TD (1979) Mixing Qualitative and Quantitative Methods : Triangulation in Action. *Administrative Science Quarterly* 24(4): 602–611.

Kallinikos J (2005) The order of technology : complexity and control in a connected world. *Information and Organization* 15(3): 185–202.

Kallinikos J (2006) *The Consequences of Information*. Edward Elgar Publishing Ltd.

Kallinikos J (2009) On the Computational Rendition of Reality : Artefacts and Human Agency. 16(2): 183–202. DOI: 10.1177/1350508408100474.

Kallinikos J and Tempini N (2014) Patient data as medical facts: social media practices as a patient data as a foundation for medical knowledge creation. *Information Systems Research Publication* 25(4): 817–833. DOI: 10.1287/isre.2014.0544.

Kallinikos J, Hasselbladh H and Marton A (2013) Governing social practice. *Theory and Society* 42(4): 395–421. DOI: 10.1007/s11186-013-9195-y.

Karahanna E, Xu SX, Xu Y, et al. (2018) The needs-affordances-features perspective for the use of social media. *MIS Quarterly: Management Information Systems* 42(3): 737–756. DOI: 10.25300/MISQ/2018/11492.

Karat J, Karat C-M and Ukelson J (2000) Affordances, Motifation, and the Design of User Interfaces. *Communications of the* ACM 43(8): 49–51.

Karwatzki S, Dytynko O, Trenz M, et al. (2017) Beyond the Personalization–Privacy Paradox: Privacy Valuation, Transparency Features, and Service Personalization. *Journal of Management Information Systems* 34(2). Routledge: 369–400. DOI: 10.1080/07421222.2017.1334467.

Katz AN, Blasko DG and Kazmerski VA (2004) Saying what you don't mean: Social influences on sarcastic language processing. *Current Directions in Psychological Science* 13(5): 186–189. DOI: 10.1111/j.0963-7214.2004.00304.x.

Kellogg KC, Valentine MA and Christin A (2020) Algorithms at work: The new contested terrain of control. *Academy of Management Annals* 14(1): 366–410. DOI: 10.5465/annals.2018.0174.

Kelly G, Mulgan G and Muers S (2002) Creating Public Value: An analytical framework for public service reform. *Strategy Unit, Cabinet Office, UK Government*: 1–35. Available at: http://webarchive.nationalarchives.gov.uk/20100416132449/http:/www.cabinetoffice.gov.uk/media/cabinetoffice/strategy/assets/public_value2.pdf.

Khalifa AS (2004) Customer value: a review of recent literature and an integrative configuration. *Management Decision* 42(5): 645–666.

Kim Y and Slotegraaf RJ (2016) Brand-embedded interaction: a dynamic and personalized interaction for co-creation. *Marketing Letters* 27(1): 183–193. DOI: 10.1007/s11002-015-9361-2.

Kiritchenko S and Mohammad S (2018) Examining Gender and Race Bias in Two Hundred Sentiment Analysis Systems.: 43–53. DOI: 10.18653/v1/s18-2005.

Knuth DE (1996) *Selected Papers on Computer Science*. Cambridge University Press.

Kobsa A (2000) User modeling as a key factor in system personalization. In: *CHI2000*, The Hague, 2000.

Kokolakis S (2017) Privacy attitudes and privacy behaviour: A review of current research on the privacy paradox phenomenon. *Computers and Security* 64(2013): 122–134. DOI: 10.1016/j.cose.2015.07.002.

Komiak SYX and Benbasat I (2006) The effects of personalization and familiarity on trust and adoption of recommendation agents. *MIS Quarterly* 30(4): 1.

Kramer ADI, Guillory JE and Hancock JT (2014) Experimental evidence of massive-scale emotional contagion through social networks. *Proceedings of the National Academy of Sciences of the United States of America* 111(24): 8788–8790. DOI: 10.1073/pnas.1320040111.

Krishnaraju V, Mathew SK and Sugumaran V (2016) Web personalization for user acceptance of technology: An empirical investigation of E-government services. *Information Systems Frontiers* 18(3). Information Systems Frontiers: 579–595. DOI: 10.1007/s10796-015-9550-9.

Kulesza T, Stumpf S, Burnett M, et al. (2013) Too much, too little, or just right? Ways explanations impact end users' mental models. *Proceedings of IEEE Symposium on Visual Languages and Human-Centric Computing, VL/HCC*: 3–10. DOI: 10.1109/VLHCC.2013.6645235.

Kumar V, Rajan B, Venkatesan R, et al. (2019) Understanding the role of artificial intelligence in personalized engagement marketing. *California Management Review* 61(4): 135–155. DOI:

10.1177/0008125619859317.

Kwon K, Cho J and Park Y (2010) How to best characterize the personalization construct for e-services. *Expert Systems with Applications* 37(3). Elsevier Ltd: 2232–2240. DOI: 10.1016/j.eswa.2009.07.050.

Lambrecht A and Tucker C (2013) When does retargeting work? Information specificity in online advertising. *Journal of Marketing Research* 50(5): 561–576. DOI: 10.1509/jmr.11.0503.

Langely A and Denis J-L (2006) Neglected dimensions: Towards a situated view. In: Lines R, Stensaker I, and Langley A (eds) *Handbook of Organizational Change and Learning*. Bergen: Fagbokforlaget, pp. 136–161.

Latham R and Sassen S (2005) *Digital Formations: IT and New Architectures in the Global Realm*. Princeton: Princeton University Press.

Latour B (1987) *Science in Action: How to Follow Scientists and Engineers through Society*. Cambridge: Harvard university press.

Latour B (2005) *Reassembling the Social: An Introduction to Actor- Network-Theory*. Oxford: Oxford University Press.

Latour B (2017) Technology Is Society Made Durable. *Design: Critical and Primary Sources*: 103–132. DOI: 10.5040/9781474282901.0013.

Leavy S (2018) Gender bias in artificial intelligence: The need for diversity and gender theory in machine learning. *Proceedings - International Conference on Software Engineering*: 14–16. DOI: 10.1145/3195570.3195580.

Lee DH and Brusilovsky P (2017) Improving personalized recommendations using community membership information. *Information Processing and Management* 53(5). Elsevier Ltd: 1201–1214. DOI: 10.1016/j.ipm.2017.05.005.

Leonardi PM (2009) Why do people reject new technologies and stymie organizational changes of which they are in favor? Exploring misalignments between social interactions and materiality. *Human Communication Research* 35(3): 407–441. DOI: 10.1111/j.1468-2958.2009.01357.x.

Leonardi PM (2011) When Flexible Rouintes Meet Flexible Technologies. *MIS Quarterly* 35(1): 147–167. Available at: papers3://publication/uuid/CA0A1006-57F7-4207-A810-D72599FB77D6.

Leonardi PM (2013) When does technology use enable network change in organizations? A comparative study of feature use and shared affordances. *MIS Quarterly: Management*

Information Systems 37(3): 749–776. DOI: 10.25300/misq/2013/37.3.04.

Leonardi PM (2017) The social media revolution: Sharing and learning in the age of leaky knowledge. *Information and Organization* 27(1). Elsevier Ltd: 47–59. DOI: 10.1016/j.infoandorg.2017.01.004.

Leonardi PM and Barley SR (2010) What's Under Construction Here? Social Action, Materiality, and Power in Constructivist Studies of Technology and Organizing. *The Academy of Management Annals* 4(1): 1–51. DOI: 10.1080/19416521003654160.

Leonardi PM and Vaast E (2017) Social Media and Their Affordances for Organizing : A Review and Agenda for Research A REVIEW AND AGENDA FOR RESEARCH Paul M . Leonardi Technology Management Program University of California , Santa Barbara Emmanuelle Vaast Desautels Faculty of Management. *Academy of Management Annals*.

Lepri B, Oliver N, Letouzé E, et al. (2018) Fair, Transparent, and Accountable Algorithmic Decision-making Processes: The Premise, the Proposed Solutions, and the Open Challenges. *Philosophy and Technology* 31(4): 611–627. DOI: 10.1007/s13347-017-0279-x.

Leslie D (2019) Understanding artificial intelligence ethics and safety systems in the public sector. *The Alan Turing Institute*.

Li C and Liu J (2017) A name alone is not enough: A reexamination of web-based personalization effect. *Computers in Human Behavior* 72. Elsevier Ltd: 132–139. DOI: 10.1016/j.chb.2017.02.039.

Li F and Jagadish H V. (2014) Constructing an interactive natural language interface for relational databases. *Proceedings of the VLDB Endowment* 8(1): 73–84. DOI: 10.14778/2735461.2735468.

Li L, Zheng L, Yang F, et al. (2014) Modeling and broadening temporal user interest in personalized news recommendation. *Expert Systems with Applications* 41(7). Elsevier Ltd: 3168–3177. DOI: 10.1016/j.eswa.2013.11.020.

Li Y, Xu L and Liu B (2018) Evolutionary game analysis on e-commerce personalization and privacy protection. *Wuhan University Journal of Natural Sciences* 23(1): 17–24. DOI: 10.1007/s11859-018-1289-y.

Li YM and Yeh YS (2010) Increasing trust in mobile commerce through design aesthetics. *Computers in Human Behavior* 26(4). Elsevier Ltd: 673–684. DOI: 10.1016/j.chb.2010.01.004.

Li YM, Wu C Te and Lai CY (2013) A social recommender mechanism for e-commerce: Combining similarity, trust, and relationship. *Decision Support Systems* 55(3). Elsevier B.V.: 740–752. DOI: 10.1016/j.dss.2013.02.009.

Liang TP, Lai HJ and Ku YIC (2006) Personalized content recommendation and user satisfaction: Theoretical synthesis and empirical findings. *Journal of Management Information Systems* 23(3): 45–70. DOI: 10.2753/MIS0742-1222230303.

Liem CCS, Langer M, Demetriou A, et al. (2018) Psychology Meets Machine Learning: Interdisciplinary Perspectives on Algorithmic Job Candidate Screening. In: Escalante HJ, Escalera S, Guyon I, et al. (eds) *Explainable and Interpretable Models in Computer Vision and Machine Learning*. Springer International Publishing, pp. 197–253.

Lindebaum D, Vesa M and den Hond F (2020) Insights from the Machine Stops to Better Understand Rational Assumptions in Algorithmic Decision-Making and Its Implications for Organizations. *Academy of Management Review* 45(1): 247–263.

Lipton ZC (2018) The mythos of model interpretability. *Communications of the* ACM 61(10): 35–43. DOI: 10.1145/3233231.

Littlejohn R and Dippmann J (2011) *Riding the Wind with Liezi: New Perspectives on the Daoist Classic*. SUNY Press.

Luhmann N (2005) *Risk : A Sociological Theory*. New Brunswick N.J.: Aldine Transaction. Available at: http://www.worldcat.org/title/risk-a-sociological-theory/oclc/476713602 (accessed 22 April 2017).

MacKenzie D (2018) ‘Making’, ‘taking’ and the material political economy of algorithmic trading. *Economy and Society* 47(4): 501–523. DOI: 10.1080/03085147.2018.1528076.

Marakas GM, Johnson RD and Palmer JW (2000) Theoretical model of differential social attributions toward computing technology: when the metaphor becomes the model. *International Journal of Human Computer Studies* 52(4): 719–750. DOI: 10.1006/ijhc.1999.0348.

Marino DL, Grandio J, Wickramasinghe CS, et al. (2020) AI Augmentation for Trustworthy AI: Augmented Robot Teleoperation.: 155–161. DOI: 10.1109/hsi49210.2020.9142659.

Markus ML and Robey D (1988) Information Technology and Organizational Change: Causal Structure in Theory and Research. *Management Science* 34(5): 583–598.

Marshall C and Rossman G (2014) *Designing Qualitative Research*. Sage publications.

Martín-Vicente MI, Gil-Solla A, Ramos-Cabrer M, et al. (2012) Semantic inference of user’s reputation and expertise to improve collaborative recommendations. *Expert Systems with Applications* 39(9): 8248–8258. DOI: 10.1016/j.eswa.2012.01.162.

Masli A, Richardson VJ, Watson MW, et al. (2016) Senior executives' IT management responsibilities: Serious IT-related deficiencies and CEO/CFO turnover. *MIS Quarterly: Management Information Systems* 40(3): 687–708. DOI: 10.25300/MISQ/2016/40.3.08.

Mason J (2002) *Qualitative Researching*. Sage.

Maspero G (2009) *Manual of Egyptian Archaeology: A Guide to the Studies of Antiquities in Egypt*. BoD.

Mathur P (2017) *Technological Forms and Ecological Communication: A Theoretical Heuristic*. Lexington Books.

Matzner T (2019) The Human Is Dead – Long Live the Algorithm! Human-Algorithmic Ensembles and Liberal Subjectivity. *Theory, Culture and Society* 36(2): 123–144. DOI: 10.1177/0263276418818877.

McAfee A and Brynjolfsson E (2018) *Machine, Platform, Crowd*. WW Norton & Co.

McCorduck P (2004) *Machines Who Think*. Natick, MA: A. K. Peters, Ltd.

McKinney SM, Sieniek M, Godbole V, et al. (2020) International evaluation of an AI system for breast cancer screening. *Nature* 577(7788). Springer US: 89–94. DOI: 10.1038/s41586-019-1799-6.

McMahon C, Lowe A and Culley S (2004) Knowledge management in engineering design: Personalization and codification. *Journal of Engineering Design* 15(4): 307–325. DOI: 10.1080/09544820410001697154.

Mercado JE, Rupp MA, Chen JYC, et al. (2016) Intelligent Agent Transparency in Human-Agent Teaming for Multi-UxV Management. *Human Factors* 58(3): 401–415. DOI: 10.1177/0018720815621206.

Merriam-Webster.com (2020) 'Agent'. Available at: https://www.merriam-webster.com/dictionary/agent.

Merriam-Webster.com (2021) 'Artificial Intelligence'. Available at: https://www.merriam-webster.com/dictionary/artificial intelligence.

Meynhardt T (2015) Public value: Turning a Conceptual Framework Into a Scorecard. *Public value and public administration*: 147–169.

Miceli G, Ricotta F and Costabile M (2007) Customizing customization: A conceptual framework for interactive personalization. *Journal of Interactive Marketing* 21(2). Elsevier: 6–25. DOI: 10.1002/dir.20076.

Miles M and Huberman A (1994) *Qualitative Data Analysis: A Sourcebook*. Available at: http://www.jstor.org/stable/pdf/1163741.pdf (accessed 24 April 2017).

Miller T (2019) Explanation in artificial intelligence: Insights from the social sciences. *Artificial Intelligence* 267. Elsevier B.V.: 1–38. DOI: 10.1016/j.artint.2018.07.007.

Mitchell TM (1997) Does machine learning really work? *AI Magazine* 18(3): 71–83. Available at: http://www.aaai.org/ojs/index.php/aimagazine/article/viewArticle/1303.

Mittal B and Lassar WM (1996) The Role of Personalization in Service Encounters. *Journal of Retailing* 72(1): 95–109.

Mobasher B, Cooley R and Srivastava J (2000) Automatic Personalization Based on Web Usage Mining Architecture for Usage-based Web Personalization Mining Usage Data for Web Personalization. *Communications of the ACM* 43(8): 1–20. DOI: 10.1145/345124.345169.

Möhlmann M and Henfridsson O (2019) What people hate about being managed by algorithms, according to a study of Uber drivers. *Harvard Business Review*: 2–8.

Monroe W, Green S and Manning CD (2014) Word segmentation of informal Arabic with domain adaptation. *52nd Annual Meeting of the Association for Computational Linguistics, ACL 2014 - Proceedings of the Conference* 2: 206–211. DOI: 10.3115/v1/p14-2034.

Montgomery AL and Smith MD (2009) Prospects for Personalization on the Internet. *Journal of Interactive Marketing* 23(2): 130–137. DOI: 10.1016/j.intmar.2009.02.001.

Moore J (2016) Here Are 10 of the Oldest IT Systems in the Federal Government. *NextGov*, May. Available at: https://www.nextgov.com/cio-briefing/2016/05/10-oldest-it-systems-federal-government/128599/.

Moore MH (1995) *Creating Public Value : Strategic Management in Government*. Harvard University Press.

Moore MH (2003) *The Public Value Scorecard: A Rejoinder and an Alternative to 'Strategic Performance Measurement and Management in Non-Profit Organizations' by Robert Kaplan*.

Moore MH (2012) Public Value Accounting : Establishing a Philosophical Basis A commissioned background paper about Public Value Accounting as it relates to the creation of public value .

Mukherjee A and Nath P (2007) *Role of Electronic Trust in Online Retailing: A Re-Examination of the Commitment-Trust Theory*. DOI: 10.1108/03090560710773390.

Mullainathan S and Spiess J (2017) Machine learning: An applied econometric approach. *Journal of Economic Perspectives* 31(2): 87–106. DOI: 10.1257/jep.31.2.87.

Needham C (2011) Personalization: From story-line to practice. *Social Policy and Administration* 45(1): 54–68. DOI: 10.1111/j.1467-9515.2010.00753.x.

Negnevitsky M (2005) *Artificial Intelligence: A Guide to Intelligent Systems*. Pearson.

Nevitte N (2002) *Value Change and Governance in Canada*. University of Toronto Press.

Nicolini D (2013) *Practice Theory, Work, and Organization: An Introduction*. Oxford: Oxford University Press.

Norman D (2013) *The Design of Everyday Things: Revised and Expanded Edition*. New York: Basic Books.

O'Flynn J (2005) A Public Value Framework for Contractual Governance Janine O ' Flynn PUBLIC, Issue 07, ESADEs Institute of Public Management, Barcelona. 19/12/2005. In: *PUBLIC*, Barcelona, 2005, pp. 5–8. ESADEs Institute of Public Management.

O'Flynn J (2007) From New Public Management to Public Value : Paradigmatic Change and Managerial Implications. *Australian Journal of Public Administration* 66(3): 353–366. DOI: 10.1111/j.1467-8500.2007.00545.x.

OECD (2019) OECD Principles on AI. Available at: http://www.oecd.org/going-digital/ai/principles/.

Olhede SC and Wolfe PJ (2018) The growing ubiquity of algorithms in society: Implications, impacts and innovations. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* 376(2128). DOI: 10.1098/rsta.2017.0364.

Orlikowski W and Scott S V (2015) What happens when evaluation goes online ?: exploring apparatuses of valuation in the travel sector. DOI: 10.1287/orsc.2013.0877.

Orlikowski WJ (2000) Using technology and constituting structures. *Organization Science* 11(4): 404–428.

Orlikowski WJ (2007) Sociomaterial practices: Exploring technology at work. *Organization Studies* 28(9): 1435–1448. DOI: 10.1177/0170840607081138.

Orlikowski WJ and Barley SR (2001) Technology and institutions: What can research on information technology and research on organizations learn from each other? *MIS Quarterly: Management*

Information Systems 25(2): 145–165. DOI: 10.2307/3250927.

Orlikowski WJ and Baroudi JJ (1991) Studying Information Technology in Organizations: Research Approaches and Assumptions. *Information systems Research* 2(1): 1–28.

Orlikowski WJ and Iacono S (2001) Desperately seeking the 'IT'in IT research: A call to theorizing the IT artifact. *Information Systems Research* 12(2): 121–134. DOI: 10.1287/isre.12.2.121.9700.

Orlikowski WJ and Scott S V. (2008) Sociomateriality: Challenging the Separation of Technology, Work and Organization. *The Academy of Management Annals* 2(1): 433–474. DOI: 10.1080/19416520802211644.

Osborne D and Gaebler T (1992) *Reinventing Government: How the Entrepreneurial Spirit Is Transforming the Public Sector*. Mass: Addison: Reading.

Osborne S (2010) Delivering Public Services: Time for a new theory? *Public Management Review* 12(1): 1–10. DOI: 10.1080/14719030903495232.

Osborne SP (2010) Delivering Public Services: Time for a new theory? *Public Management Review* 12(1): 1–10. DOI: 10.1080/14719030903495232.

Osborne SP and Strokosch K (2013) It takes two to tango? Understanding the co-production of public services by integrating the services management and public administration perspectives. *British Journal of Management* 24(S3). DOI: 10.1111/1467-8551.12010.

Oved MC (2019) How Waterfront Toronto's Stephen Diamond went up against Sidewalk Labs and won. *The Toronto Star*, 31 October. Available at: https://www.thestar.com/news/gta/2019/10/31/how-waterfront-torontos-stephen-diamond-went-up-against-sidewalk-labs-and-won.html.

Parasuraman A, Zeithaml VA and Berry LL (1988) Servqual: A Multiple-Item Scale For Measuring Consumer Perceptions of Service Quality. *Journal of Retailing* 64(1): 12.

Pasquale F (2017) Toward a Fourth Law of Robotics: Preserving Attribution, Responsibility, and Explainability in an Algorithmic Society.

Paudyai P and Wong BLW (2018) Algorithmic opacity: Making algorithmic processes transparent through abstraction hierarchy. *Proceedings of the Human Factors and Ergonomics Society* 1: 192–196. DOI: 10.1177/1541931218621046.

Perez JC (2011) Page: Google+ will have big effect on other Google products. *ComputerWorld*, October. Available at: https://www.computerworld.com/article/2498930/page--google--

will-have-big-effect-on-other-google-products.html.

Pettey C (2016) Treating information as an asset. *Smarter with Gartner*, 17 February. Available at: www.gartner.com/smarterwithgartner/treating-information-as-an-asset.

Pfister H and Böhm G (2008) Emotional functions in decision-making. 3(1): 5–17.

Piccoli G, Lui TW and Grün B (2017) The impact of IT-enabled customer service systems on service personalization, customer service perceptions, and hotel performance. *Tourism Management* 59. Elsevier Ltd: 349–362. DOI: 10.1016/j.tourman.2016.08.015.

Pickering A (1993) The Mangle of Practice: Agency and Emergence in the Sociology of Science. *American Journal of Sociology* 99(3): 559–589. DOI: 10.1086/230316.

Pinch TJ and Bjiker WE (1984) The social construction of facts and artefacts: How the sociology of science and the sociology of technology might benefit each other. *Social Studies of Science* 14(3): 399–441.

Pine JI (1993) *Mass Customization*. Boston: Harvard Business School.

Poole D, Mackworth A and Goebel R (1998) *Computational Intelligence: A Logical Approach*. Oxford: Oxford University Press.

Puranam P, Y.R. S, He VF, et al. (2018) *Algorithmic Induction through Machine Learning: Opportunities for Management and Organization Research*.

Qiu H, Yun L, Guo G, et al. (2018) BPRH: Bayesian Personalized Ranking for Heterogeneous Implicit Feedback. *Information Sciences* 453: 80–98.

Raj M and Seamans R (2019) Primer on artificial intelligence and robotics. *Journal of Organization Design* 8(1). Journal of Organization Design. DOI: 10.1186/s41469-019-0050-0.

Rakower L (2011) Blurred Line: Zooming in on Google Street View and the Global Right to Privacy. *Brooklyn Journal of International Law* 37(1): 8.

Ramos J (2003) Using TF-IDF to Determine Word Relevance in Document Queries. In: *Proceedings of the first instructional conference on machine learning*, Piscataway, NJ, 2003, pp. 133–142.

Reichheld FF (2003) One Number You Need to Grow. *Harvard Business Review*.

Reuver M De, Sørensen C and Basole RC (2018) The digital platform : a research agenda. *Journal of Information Technology* 33(2). Palgrave Macmillan UK: 124–135. DOI: 10.1057/s41265-016-0033-3.

Reynar JC (1998) Topic segmentation: Algorithms and applications. *IRCS Technical Reports Series*: 66.

Riemer K and Totz C (2003) The Many Faces of Personalization. *The Customer Centric Enterprise*: 35–50. DOI: 10.1007/978-3-642-55460-5_3.

Rosenblat A and Stark L (2016) Algorithmic labor and information asymmetries: A case study of Uber's drivers. *International Journal of Communication* 10: 3758–3784. DOI: 10.2139/ssrn.2686227.

Salonen V and Karjaluoto H (2016) Web personalization: The state of the art and future avenues for research and practice. *Telematics and Informatics* 33(4): 1088–1104. DOI: 10.1016/j.tele.2016.03.004.

Samek W, Wiegand T and Müller K-R (2017) Explainable Artificial Intelligence: Understanding, Visualizing and Interpreting Deep Learning Models. Available at: http://arxiv.org/abs/1708.08296.

Sandvig C, Hamilton K, Karahalios K, et al. (2014) Auditing Algorithms -- Sandvig -- ICA 2014 Data and Discrimination Preconference. "*Data and Discrimination: Converting Critical Concerns into Productive Inquiry," a preconference at the 64th Annual Meeting of the International Communication Association. May 22, 2014; Seattle*, WA, USA.: 1–23. DOI: 10.1109/DEXA.2009.55.

Sandvig C, Hamilton K, Karahalios K, et al. (2016) When the algorithm itself is a racist: Diagnosing ethical harm in the basic Components of Software. *International Journal of Communication* 10(June): 4972–4990.

Sauder M and Espeland WN (2009) The Discipline of Rankings : Tight Coupling and Organizational Change. 74: 63–82.

Scheepers R, Venkitachalam K and Gibbs MR (2004) Knowledge strategy in organizations: Refining the model of Hansen, Nohria and Tierney. *Journal of Strategic Information Systems* 13(3): 201–222. DOI: 10.1016/j.jsis.2004.08.003.

Schwartz P (1989) The Computer in German and American Constitutional Law: Towards an American Right of Informational Self-Determination. *The American Journal of Comparative Law* 37(4): 675. DOI: 10.2307/840221.

Schwartz PM and Solove DJ (2011) The PII problem: Privacy and a new concept of personally identifiable information. *New York University Law Review* 86(6): 1814–1894.

Searle JR (1995) *The Construction of Social Reality*. NY: Free Press.

Searle JR (2008) Language and social ontology. *Theory and Society* 37(5). Springer Netherlands: 443–459. DOI: 10.1007/s11186-008-9068-y.

Seneler CO, Basoglu N and Daim T (2009) Interface feature prioritization for web services: Case of online flight reservations. *Computers in Human Behavior* 25(4). Elsevier Ltd: 862–877. DOI: 10.1016/j.chb.2008.12.028.

Sharma R, Mithas S and Kankanhalli A (2014) Transforming decision-making processes: A research agenda for understanding the impact of business analytics on organisations. *European Journal of Information Systems* 23(4): 433–441. DOI: 10.1057/ejis.2014.17.

Shen A and Ball AD (2009) Is personalization of services always a good thing? Exploring the role of technology-mediated personalization (TMP) in service relationships. *Journal of Services Marketing* 23(2): 80–92. DOI: 10.1108/08876040910946341.

Shrestha YR, Ben-Menahem SM and von Krogh G (2019) Organizational Decision-Making Structures in the Age of Artificial Intelligence. *California Management Review*: 1–18. DOI: 10.1177/0008125619862257.

Simmons R, Birchall J and Prout A (2007) Hearing Voices? User Involvement and Public Service Cultures. *Consumer Policy Review* 17(5): 2–8.

Simon HA (1973) Applying Information Technology to Organization Design Herbert A. Simon. *Public Administration Review* 33(3): 268–278.

Sismondo S (1993) Some Social Constructions. *Social Studies of Science* 23(3). SAGE Publications: 515–553. DOI: 10.1177/0306312793023003004.

Smith A (1776) *An Inquiry Into the Nature and Causes of the Wealth of Nations*. London: Strahan and Cadell.

Sonenshein S (2016) Routines and creativity: From dualism to duality. *Organization Science* 27(3): 739–758. DOI: 10.1287/ORSC.2016.1044.

Spohrer J, Maglio PP, Bailey J, et al. (2007) Steps toward a science of service systems. *Computer* 40(1): 71–77. DOI: 10.1109/MC.2007.33.

Stake RE (1995) *The Art of Case Study Research*. Sage Publications.

Stohl C, Stohl M and Leonardi PM (2016) Managing opacity: Information visibility and the paradox of transparency in the digital age. *International Journal of Communication* 10(1): 123–137.

Stoker G (2006) Public Value Management: A New Narrative for Networked Governance?'. *The American Review of Public Administration* 36(1): 41. DOI: 10.1177/0275074005282583.

Strauss A and Corbin J (1998) *Basics of Qualitative Research: Procedures and Techniques for Developing Grounded Theory*. Thousand Oaks.

Stumpf S, Rajaram V, Li L, et al. (2007) Toward Harnessing User Feedback for Machine Learning. In: *Proceedings of the 12th International Conferenceon Intelligent User Interface* (ed. A for C Machinery), New York, 2007, pp. 82–91. DOI: dx.doi.org/10.1145/1216295.121631.

Suchman LA (2007) *Human–Machine Reconfigurations: Plans and Situated Actions*. Cambridge: Cambridge University Press.

Sudhindra S, Ganesh LS and Arshinder K (2017) Knowledge transfer: An information theory perspective. *Knowledge Management Research and Practice* 15(3). Palgrave Macmillan UK: 400–412. DOI: 10.1057/s41275-017-0060-z.

Sun T, Gaut A, Tang S, et al. (2020) Mitigating gender bias in natural language processing: Literature review. ACL 2019 - 57*th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference*: 1630–1640. DOI: 10.18653/v1/p19-1159.

Sunikka A and Bragge J (2012) Applying text-mining to personalization and customization research literature - Who, what and where? *Expert Systems with Applications* 39(11): 10049–10058. DOI: 10.1016/j.eswa.2012.02.042.

Surprenant C and Solomon M (1987) Predictability and personalization in the service encounter. *Journal of Marketing* 51: 86–96.

Swaminanthan J (2001) Enabling Customization Using Standardized Operations. *California Management Review* 43(3): 125–135.

Sweeney JC, Soutar GN, Whiteley A, et al. (1996) Generating Consumption Value Items: A Parallel Interviewing Process Approach. In: *Association for Consumer Research: Second Asia Pacific Conference*, 1996, pp. 108–115.

Taddeo M and Floridi L (2018) How AI can be a force for good. *Science* 361(6404): 751–753.

Taylor T (2014) No discretion required? Caseworker autonomy and the rules of welfare reform. *Sociological Inquiry* 84(3): 412–434. DOI: 10.1111/soin.12038.

Thaler RH and Sunstein CR (2008) *Nudge: Improving Decisions about Health, Wealth, and Happiness*. Yale University Press.

Thomaz AL and Breazeal C (2008) Teachable Robots: Understanding Human Teaching Behavior to Build More Effective Robot Learners. *Artificial Intelligence* 172(6): 716–737.

Tilson D (2010) Digital Infrastructures : The Missing IS Research Agenda.: 1–12. DOI: 10.1287/isre.1100.0318.

Totaro P and Ninno D (2014) The Concept of Algorithm as an Interpretative Key of Modern Rationality. *Theory, Culture & Society* 31(4): 29–49. DOI: 10.1177/0263276413510051.

Tseng M and Jiao J (2001) Mass Customization. In: Salvendy G (ed.) *Handbook of Industrial Engineering*. 3rd ed. New York: Wiley.

Tsoukas H and Chia R (2002) On organizational becoming : Rethinking organizational change.

Tuzhilin A (2009) Personalization: the state of the art and future directions. In: Adomavicius G and Gupta A (eds) *Business Computing: Handbooks in Information Systems*. 3rd ed. Bingley: Emerald, pp. 3–43.

UAE Cabinet (2016) UAE *Model for Government Leadership*. Available at: https://uaecabinet.ae/en/details/news/uae-model-for-government-leadership.

Valverde M and Flynn A (2018) MORE BUZZWORDS THAN ANSWERS' - TO SIDEWALK LABS IN TORONTO. *Landscape Architecture Frontiers* 6(2): 115.

van Doorn J and Hoekstra JC (2013) Customization of online advertising: The role of intrusiveness. *Marketing Letters* 24(4): 339–351. DOI: 10.1007/s11002-012-9222-1.

Vargo S and Lusch R (2004) Evolving to a New Dominant Logic for Marketing. *Journal of Marketing* 68(January): 1–17. DOI: 10.1509/jmkg.68.1.1.24036.

Vargo S, Maglio P and Archpru Akaka M (2008) On value and value co-creation: A service systems and service logic perspective. *European management journal* 26: 145–152.

Venkatesh V, Thong JYL, Chant, Frank KY, et al. (2016) Managing Citizens ' Uncertainty in E-Government Services : The Mediating and Moderating Roles of Transparency and Trust. *Information Systems Research* 27(1)(September 2018): 87–111.

Vesanen J and Raulas M (2006) Building bridges for personalization: a process model for marketing. *Journal of Interactive Marketing* 20(1). DOI: 10.1002/dir.

Villasenor J and Foggo V (2019) Algorithms and sentencing: What does due process require? *Brookings Institute*, March. Available at: https://www.brookings.edu/blog/techtank/2019/03/21/algorithms-and-sentencing-what-does-due-process-require/.

Von Krogh G (2012) How does social software change knowledge management? Toward a strategic research agenda. *Journal of Strategic Information Systems* 21(2). Elsevier B.V.: 154–164. DOI: 10.1016/j.jsis.2012.04.003.

Waltl B and Vogl R (2018) Explainable artificial intelligence - The new frontier in legal informatics. *Jusletter IT* (February).

Wang S-C (2003) Artificial Neural Network. In: *Interdisciplinary Computing in Java Programming. The Springer International Series in Engineering and Computer Science*. Boston, MA, pp. 81–100.

Wang Y and Kosinski M (2017) Deep Neural Networks Can Detect Sexual Orientation From Faces. *Journal of personality and social psychology*: 1–47. DOI: 10.17605/OSF.IO/HV28A.

Wei W and Guo H (2009) A bias free, quantum random number generator. *Optics InfoBase Conference Papers*: 2–4. DOI: 10.1364/fio.2009.fwl4.

Weinberger M, Bouhnik D and Gan R (2018) Place Determinants for the Personalization-Privacy Tradeoff among Students. *Proceedings of the 2018 InSITE Conference* 15: 907. DOI: 10.28945/4069.

Weller A (2019) Transparency: Motivations and Challenges. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 11700 LNCS(Section 2): 23–40. DOI: 10.1007/978-3-030-28954-6_2.

Wendt A (1999) *Social Theory of International Politics*. Cambridge University Press. DOI: 10.2307/20049552.

Williams BA, Brooks CF and Shmargad Y (2018) How Algorithms Discriminate Based on Data they Lack: Challenges, Solutions, and Policy. *Source Journal of Information Policy Journal of Information Policy* 8(8): 78–115. Available at: http://www.jstor.org/stable/10.5325/jinfopoli.8.2018.0078%0Ahttp://www.jstor.org/stable/10.5325/jinfopoli.8.2018.%0Ahttp://about.jstor.org/terms.

Winfield AFT and Jirotka M (2018) Ethical governance is essential to building trust in robotics and artificial intelligence systems. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* 376(2133). DOI: 10.1098/rsta.2018.0085.

Wirtz BW, Weyerer JC and Geyer C (2019) Artificial Intelligence and the Public Sector—Applications and Challenges. *International Journal of Public Administration* 42(7). Routledge: 596–615. DOI:

10.1080/01900692.2018.1498103.

Wisconsin SC of (2016) *State v. Eric L. Loomis*. Available at: https://wscca.wicourts.gov/caseDetails.do?caseNo=2015AP000157&cacheId=6769A660FF16C5ED9230BF0852100C7E&recordCount=1&offset=0.

Wu T (2019) Will artificial intelligence eat the law? The rise of hybrid social-ordering systems. *Columbia Law Review* 119(7): 2001–2028. DOI: 10.2139/ssrn.3492846.

Xiao L, Guo FP and Lu QB (2018) Mobile Personalized Service Recommender Model Based on Sentiment Analysis and Privacy Concern. *Mobile Information Systems* 2018(1). DOI: 10.1155/2018/8071251.

Xu M, Ma Y, Liu X, et al. (2017) AppHolmes: Detecting and characterizing app collusion among third-party android markets. *26th International World Wide Web Conference*, WWW 2017: 143–152. DOI: 10.1145/3038912.3052645.

Yang Y (2010) Web user behavioral profiling for user identification. *Decision Support Systems* 49(3): 261–271. DOI: 10.1016/j.dss.2010.03.001.

Yanofsky NS (2011) Towards a definition of an algorithm. *Journal of Logic and Computation* 21(2): 253–286. DOI: 10.1093/logcom/exq016.

Yin RK (2003) *Case Study Research : Design and Methods*.

Yoganarasimhan H (2015) Search Personalization Using Machine Learning. *SSRN Electronic Journal*. DOI: 10.2139/ssrn.2590020.

Zakaria NZ., Hamdan A., Yahaya J, et al. (2016) User Centric Software Quality Model For Sustainability: A Review. *Lecture Notes on Software Engineering* 4(3). DOI: 10.18178/lnse.2016.4.3.250.

Zammuto RF, Griffith TL, Majchrzak A, et al. (2007) Information technology and the changing fabric of organization. *Organization Science* 18(5): 749–762. DOI: 10.1287/orsc.1070.0307.

Zeithaml V (1988) Consumer Perceptions of Price, Quality, and Value: A Means-End Model and Synthesis of Evidence. *Journal of Marketing* 52(3): 2–22.

Zerilli J, Knott A, Maclaurin J, et al. (2019) Transparency in Algorithmic and Human Decision-Making: Is There a Double Standard? *Philosophy and Technology* 32(4). Philosophy & Technology: 661–683. DOI: 10.1007/s13347-018-0330-6.

Zhang J (2011) The Perils of Behavior-Based Personalization. *Marketing Science* 30(1): 170–186.

Zuboff S (2015) Big other: Surveillance capitalism and the prospects of an information civilization. *Journal of Information Technology* 30(1). Nature Publishing Group: 75–89. DOI: 10.1057/jit.2015.5.

Zuboff S (2019) *The Age of Surveillance Capitalism*. London: Public Affairs, Hachette Book Group.

Zuboff S and Maxmin J (2002) *The Support Economy: Why Corporations Are Failing Individuals and the Next Episode of Capitalism*. Penguin Books.