Hidden Intentions Behind Recommender Systems: Understanding Online Manipulation as Covert Influence

Master Thesis

Javier Mondragón Briseño

TU Delft

Hidden Intentions Behind Recommender Systems: Understanding Online Manipulation as Covert Influence

Master Thesis

by

Javier Mondragón Briseño

Student Name	Student Number
Mondragón	5837618

First Supervisor:	M. Klenk
Second Supervisor and Chair:	N.v.d. Wal
Project Duration:	03, 2024 - 07, 2024
Faculty:	Technology, Policy & Management, Delft

Cover:	René Magritte, *The Treachery of Images (This is Not a Pipe)*, 1929
Style:	TU Delft Report Style, with modifications by Daan Zwaneveld

TU Delft

Preface

This thesis explores online manipulation from recommendation systems, a subject that has fascinated me due to its interdisciplinary nature and the intersection of two topics that I feel passionate about: ethics and technology. The decision to get involved with this area was triggered by a snowball effect of arbitrary events. After beginning my masters studies at TU Delft, I applied for a part-time job position as AI Engineer at an AI start-up, after being connected via a student jobs head-hunter. A few days before joining the company in January of 2023, OpenAI released ChatGPT-3, their most advanced Large Language Model yet. Coincidentally, I began to work with Generative AI on a daily basis and got to experience first-hand the benefits and perils of blindly developing a technology that could have huge impacts for everyone. Later that year, I took a course on Ethics of AI which I found extremely interesting. I had never studied ethics or philosophy formally and I became hooked right away. After discussing with the course professor about doing my thesis on a related topic, he suggested Dr. Michael Klenk's research on online manipulation as a good starting point. Given that I am passionate about music, I wanted to study something related to music recommendation systems. Again, I had never heard of online manipulation before and I quickly decided to study more about its implications for music recommendation systems. Despite my recent interest in philosophy, I wanted to use a technical methodology since my experience as a software engineer would come in handy, which was the point where we decided to explore agent-based modeling as a way to understand online manipulation empirically. Dr. Natalie van der Wal joined as chair and second-supervisor and I began my journey to learn how online manipulation could affect user preferences. At some point I had to pivot to a book recommendation system because of data availability and personal interest.

I would like to express my deepest gratitude to Dr. Michael Klenk and Dr. Natalie van der Wal, whose invaluable guidance and support played a crucial role in the completion of this work, not only from an academic perspective but also from a personal one. Special thanks also go to my parents, sisters and friends for their support along these past two years.

Reflecting on this journey, I have learned how to conduct academic research, as well as the challenges of finding a subject that has not been studied and exploring an objective that contributes to the existing body of work. In particular, I got to learn about a new topic to me and its operationalization under a methodology to which I was exposed during my studies. This was significantly gratifying as it felt like a direct consequence of the knowledge that I acquired at TU Delft.

I hope this work contributes to the understanding of manipulative recommendation systems from an empirical point of view and provides a foundation for further studies in the field of online manipulation.

Javier Mondragón Briseño
Delft, July 2024

Summary

The emergence of social media and recommendation systems has profoundly transformed user interactions with digital content, bringing in both opportunities and ethical challenges. This thesis scrutinizes the online manipulation exerted by RS, which, while enhancing engagement and profitability, can guide user behavior subtly and without their awareness. This definition is based on covert influence, a particular account of manipulation.

A critical examination of existing literature reveals a significant gap: while the conceptual framework for online manipulation is well-discussed, empirical studies providing concrete evidence are scant. This research addresses this deficiency by employing an agent-based model to simulate interactions between users and recommendation systems, aiming to systematically analyze and quantify the effects of covert manipulation on user preferences.

This study contributes to the field by operationalizing the concept of manipulation within a controlled simulation of a book recommendation system, providing a clearer understanding of its mechanisms and effects. This approach not only offers insights into the ethical implications of RS but also aligns with current legislative movements, such as the European Union's Artificial Intelligence Act, aimed at regulating and mitigating harmful manipulative practices by intelligent systems. The findings are intended to guide the design and regulation of RS to ensure they serve the user's interests without compromising ethical standards.

The results show that book recommendation systems can modify user preferences by 9.79% when prioritizing items covertly, while awareness of the intentions and social influence can diminish the effect of the manipulative algorithm's intention to 4.01% and 3.64%. When compared to the 2.58% change in the case of a non-prioritized RS, the values provide a measurable estimation of the difference between manipulative and non-manipulative book RS for the change of user preferences after interacting with it for some time.

Contents

Preface i

Summary ii

Nomenclature vii

1 Introduction 1

2 Literature Background 3
2.1 Literature Review . . . 3
2.1.1 Gap in Literature . . . 5
2.1.1.1 Agent-Based Modeling for RS . . . 7
2.1.1.2 Manipulation in Book RS . . . 8
2.2 Manipulation of Online Book Users . . . 9
2.2.1 Covert Influence: a Change in Priorities . . . 9
2.2.2 Harm: a Change in User Preferences . . . 12
2.2.3 Social Influence . . . 13
2.2.4 Transparency . . . 13

3 Method 15
3.1 Model . . . 15
3.1.1 Overview . . . 15
3.1.2 Dataset . . . 16
3.1.2.1 Data Pre-processing . . . 18
3.1.3 User Personas . . . 20
3.1.4 Model Set-up . . . 21
3.1.4.1 Model Initialization . . . 21
3.1.4.2 Model Logic . . . 24
3.1.4.3 Type of Recommendations . . . 25
3.1.4.4 Model Outcome . . . 25
3.1.5 Verification and Validation . . . 26
3.1.5.1 Model Verification . . . 26
3.1.5.2 Model Validation . . . 27
3.2 Sensitivity Analysis . . . 29
3.3 Experimental Set-up . . . 35
3.3.1 Benchmark . . . 37
3.3.2 Covert . . . 37
3.3.3 Overt . . . 37
3.3.4 Overt With Social Influence . . . 37

4 Results 39
4.1 Experiments Results . . . 39

5 Discussion 45
5.1 Discussion of Results . . . 45

5.2 Strengths and Weaknesses . . . 47
5.3 Practical and Theoretical Implications . . . 48
5.4 Future Research . . . 48

6 Conclusion **50**

References **51**

A Theoretical Background **57**
A.1 What is manipulation? . . . 57
A.1.1 Accounts of Manipulation . . . 57
A.1.1.1 Manipulation as Bypassing Rationality . . . 57
A.1.1.2 Manipulation as Trickery . . . 58
A.1.1.3 Manipulation as Pressure . . . 58
A.1.2 Persuasion and Coercion . . . 59
A.1.2.1 Persuasion . . . 59
A.1.2.2 Coercion . . . 60
A.1.2.3 Relation with Manipulation . . . 60
A.1.3 Manipulation and Deception . . . 60
A.2 Covert Influence . . . 61
A.3 Harm . . . 61
A.3.1 Frustration of Self-Interest . . . 61
A.3.2 Undermining Autonomy . . . 62
A.4 Recommender Systems . . . 62
A.4.1 Types . . . 62
A.4.2 Explanations . . . 63

B Code Fragments **64**
B.1 model.py . . . 64
B.2 agents.py . . . 68

List of Figures

2.1 Documents per year resulting from searching “online manipulation” in Scopus 4
2.2 Graph of equation 2.1 for values when $\alpha = 2$ 11

3.1 Example of single step flow for an agent 16
3.2 Overview of model components and methods 17
3.3 Flow diagram of one simulation run step 18
3.4 Histogram of book genres for sensitivity analysis items dataframe 32
3.5 Sensitivity analysis results 34

4.1 Histogram of book genres for results items dataframe 40
4.2 Experiments results 42
4.3 Experiments results by reader persona 43
4.4 Distribution of average books consumed per user 44

A.1 Distinction between persuasion, manipulation and coercion as forms of influence 59

List of Tables

3.1 Description of Goodreads datasets columns 17
3.2 User agent model set-up 22
3.3 Item agent model set-up 23
3.4 Model initialization parameters 23
3.5 Model verification steps and the action(s) taken to address them 27
3.6 Model validation steps and the action(s) taken to address them 29
3.7 Sensitivity analysis parameters 30
3.8 Parameter initialization per experiment 36

4.1 General results 39
4.2 Top items consumed 41

Nomenclature

Abbreviations

Abbreviation	Definition
RS	Recommender System(s)
ABM	Agent-Based Model(ing)

1

Introduction

The rise of social media platforms in the digital era has transformed how we interact with digital content, profoundly impacting our society through phenomena such as globalization and the spread of misinformation. Central to this transformation is the deployment of recommendation systems (RS), which use data gathered from users to tailor content, thereby influencing user behavior. These algorithms, while designed to enhance user engagement and platform profitability through advertisements or subscriptions[1], have also raised significant moral concerns due to their potential to manipulate users (Genovesi, Kaesling, and Robbins, 2023). However, such manipulation can be understood in multiple ways, even when most people have a general grasp of its meaning. Therefore, the term has been researched from a philosophical perspective, as a way to properly define it and set it apart from other forms of influence, such as persuasion or coercion (Noggle, 2022). Moreover, since the domain of its application can be quite vast, there have been numerous attempts to select the conditions for which an action could be deemed manipulative (Noggle, 1996). Given that I am focusing on manipulation from digital platforms, the scope of this study is "online" manipulation (Klenk, 2022). Although the prefix mostly limits the type of system where manipulation arises, it can still be considered from different points of view. One of these views is that manipulation arises when someone influences their target in a covert way, which seems to be particularly appropriate for an algorithm that gives recommendations to users. Furthermore, online manipulation in this context is defined as *covert influence*: the use of RS to exploit user biases without overtly displaying their true objectives or without the users knowing that they are being manipulated, influencing user behavior subtly yet effectively in doing so (Susser, Roessler, and H. Nissenbaum, 2019). Although this account is one among many and leaves out some clear examples of manipulation (Klenk, 2022), the hidden component of covert influence has been suggested as a sufficient condition for manipulation (Jongepier and Klenk, 2022). Regarding RS in particular, this view is different from manipulation as an intrinsic component of the algorithm (Zhu et al., 2024), in the sense that there is an assumption that the designers or owners of a RS do have an explicit intention passed on to the system.

The current body of research, while extensive on the conceptual categorization of online manipulation, sometimes lacks empirical evidence to support its prevalence in real-life environments (Burr, Cristianini, and Ladyman, 2018). Moreover, such evidence can be collected as simulated data of interactions between users and RS (Carroll et al., 2023). Even though there are multiple challenges associated with modelling online manipulation using a simulation, there are notable examples from previous studies (e.g., Ross et al., 2019; Yesilada and Lewandowsky, 2022; Kopp, Korb, and B. I. Mills, 2018). Nevertheless, in most cases the authors use online manipulation as a term for evaluating the existence of some influence, falling short of distinguishing between

[1] How does Facebook make money?

different types of influence or what the effect of that particular account is (Yin et al., 2019). For example, Ross et al. (2019) study how likely are bots to influence public opinion, mentioning that there are growing worries about malicious actors that "might spread misinformation online to manipulate the public", yet they proceed by using the spiral of silence theory to model the evolution of opinions, without delving into the reasons why bots are being manipulative. One could argue that bots in social media are not manipulating their users because they are overt about the stance they make, even if they could still influence the users by amplifying the voice of the minorities. Therefore, this thesis addresses this gap by aiming to determine what a specific account of manipulation is causing on the preferences of the users of a RS. In order to quantify the effect, I will model a book RS with an ABM simulation, since ABM provides a thorough way to study the evolution of complex socio-technical systems by simulating the behavior of each agent (i.e. user) and measuring its change through time (Dam, Nikolic, and Lukszo, 2013). I believe this is a legitimate approach since manipulation has sometimes been deemed as a concept that is too vague to be analyzed systematically (Jongepier and Klenk, 2022), but the recent interest from researchers and policymakers to define and regulate "AI systems that deploy harmful manipulative subliminal techniques"[2] highlights the importance of better understanding it.

Furthermore, this thesis acknowledges but will not extensively delve into the precise definition of manipulation, the full range of its impacts, or the lack of data transparency from companies about their RS algorithms. These areas, while recognized, are beyond the scope of this singular investigation.

Therefore the main research question is:

How significantly does covert influence alter user preferences, compared to overt influence and no influence?

To address the central challenge of this question in the context of a RS, this thesis proposes the use of an agent-based model (ABM). This approach will enable an examination of how RS can use hidden intentions to influence user behaviour and will assess the implications for the preferences of a user in doing so. By focusing on online manipulation, this research aims to contribute to a clearer understanding of how RS can be designed and regulated to prevent problematic uses, aligning with the emerging regulations of intelligent systems such as the European Union's Artificial Intelligence Act[3]. The novelty of the study lies in how manipulation can be operationalized within a simulation paradigm for a specific dataset, allowing for an empirical and systematic method that could be easily extended to other cases.

[2] EU AI Act Briefing
[3] EU AI Act Website

2

Literature Background

This chapter contains the literature background relevant for the thesis. I start by doing a brief overview of the related research, including some discussion of how online manipulation and ABM have been studied. Then I proceed by expanding on the gap in the existing literature and the way in which I am addressing it. Lastly, I provide a theoretical background of the concepts that support the research proposal.

2.1. Literature Review

We are all familiar with the term *manipulation* in one way or another. For instance, we use the word for referring to using a tool (e.g. "manipulating a pair of scissors") or as a relationship advice (e.g. "he is manipulating you into thinking he's indispensable in your life"). Even though both case examples could refer to taking advantage of something for their own benefit, they point towards different instances of manipulation. Recently, with the rise of the digital era, the term has also been widely applied to the ways in which social media and digital platforms influence their users at the expense of the users' well-being[1]. This expression of manipulation is hereby referred to as *online manipulation*. However, besides the popularized usage of the term, online manipulation has been formally studied in the literature, as a subset of the traditional view of manipulation from philosophy (Noggle, 1996). For the present work, I will use the definition of online manipulation as covert influence provided by Susser, Roessler, and H. Nissenbaum (2019): "the use of information technology to covertly influence another person's decision-making, by targeting and exploiting decision-making vulnerabilities" (see appendix A.2). Given that I am focusing on RS, where the algorithm is the one in charge of providing the recommendations to the users without them knowing exactly how the recommendations were generated, I will restrict this definition to only consider the cases where information technology is used by a platform and not by another peer.

Even though this work is meant to fit the intersection between online manipulation, RS and ABM, the term "online manipulation" has only started to be more widely and systematically used in the last decade. Figure 2.1 shows the number of articles published per year that contain the term. For this reason, there has been work that studied this phenomenon without explicitly mentioning it. Therefore, part of this literature review contains articles that might not sound related to online manipulation while discussing it under a different wording.

Most existing recent work around manipulation has been done from a philosophical perspective: the use of conceptual analysis breaking down a concept into smaller parts that can be analyzed each, such as the conditions

[1] The Social Dilemma

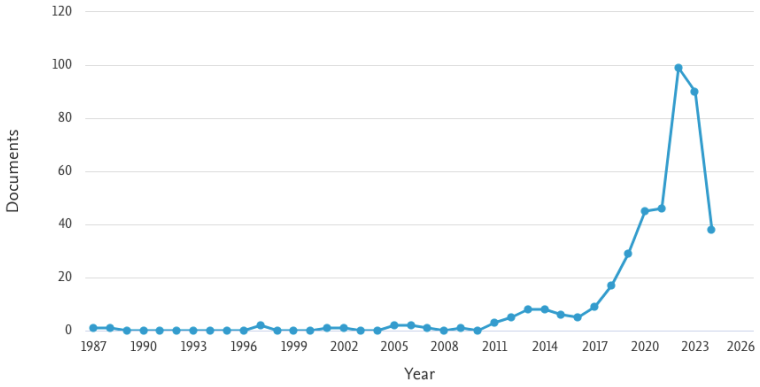


Figure 2.1: Documents per year resulting from searching "online manipulation" in Scopus

and consequences that would be required for the concept to become valid (Jongepier and Klenk, 2022). In particular, manipulation has mainly been studied through the exposure of examples where an action could or could not be deemed manipulative. These examples are sometimes stated as counterfactuals, providing a hypothetical scenario of a condition under different circumstances (Richens, Beard, and Thompson, 2022). Online manipulation, however, has recently been studied as an interdisciplinary field, borrowing examples and terminology from multiple areas of research, from psychology to computer science (Susser, Roessler, and H. Nissenbaum, 2019). These studies can provide the conceptual framework for the operationalization of covert influence as a way to model manipulation. For instance, Susser, Roessler, and H. F. Nissenbaum (2018) discuss the different views on manipulation and how they can be applied to multiple digital systems. They separate the work into two theoretical sections: one defining manipulation as a different form of influence than persuasion and coercion, by appealing to the subversion of the conscious decision-making of the target, and another defining the means by which manipulation can be executed. The latter is the more relevant for this thesis, since they conclude that an influence is manipulative when it is hidden; the target acts without knowledge of the ways in which they are being influenced. In particular, they focus on its presence within information technologies, specifically on how they can be worrisome through surveillance, digital platforms and mediation. Another example is the work by Carroll et al. (2023), where they shed light on how manipulation can be characterized and operationalized in the context of AI systems, using four axes based on existing literature:

1. Incentives: the potential reward for an AI system to act
2. Intent: the aim of the algorithm, regardless of the incentives
3. Covertness: the degree to which a user is unaware of how the AI system is attempting to influence them
4. Harm: the negative effect from the AI system on the user's state

Yet, their work focuses on manipulative behaviour in the absence of explicit human intentions. This is different from my definition of covert influence as I am approaching the problem from an intentional algorithmically-aided manipulation: embedding an intention into an algorithm so that the algorithm becomes the medium through which the target is influenced (Christiano, 2022).

To my knowledge, there has not been any research explicitly mentioning the measurement of online manipulation from RS as a change in user preferences using an ABM. Even though there have been some articles that model how users are influenced by digital platforms or elements within digital platforms with ABM, they do not dive into the philosophical implications of the term. This is a subtle but crucial aspect for studying manipulation, as mapping the contribution of a manipulative RS to an effect on the users that are consuming it depends

greatly on setting a solid theoretical base. Some of the most relevant studies for this thesis are discussed below.

Ross et al. (2019) create an ABM to determine the impact of biased content from a small proportion of bots on a social media platform to influence other users. They find that even a density of 2-4% of biased bots can be sufficient to have an effect on the opinion of the public in 60-70% of the cases. The type of influence that they measure is the spiral of silence: users who think their opinion is not aligned with their environment (i.e. the content they get) tend to avoid expressing their opinions. Even though this influence shares the hidden intention condition of the covert influence definition I am using, the main difference is related to the manipulator subject. In their work, the manipulators are bots trying to take advantage of algorithmic vulnerabilities to introduce a hidden agenda, whereas in this thesis the manipulators are the owners of the algorithms who knowingly attempt to steer their users' behaviour. The distinction is important because the implications for the responsibility of the platform are different: in the former, the platform should regulate an manipulative agent of their system (which I am not modeling), in the latter the platform is the manipulative agent. Furthermore, they do not expand on a definition of manipulation, but only mention it as the action that bots use to steer opinions.

In a similar way, Kopp, Korb, and B. I. Mills (2018) attempt to model opinion changes from users in social media in the presence of fake news, using game theory to simulate the decision-making process of the users. Additionally, they model such content as a deception from a few corrupted actors. Their results are similar to Ross et al. (2019): a small number of deceivers can have the effect of disrupting the equilibrium of the network by using fake news for exploiting opinion diffusion. Even though deception is closely related to manipulation (see section A.1.3), they measure it through their own Borden-Kopp model, which contains four information-theoretic models: degradation, corruption, denial and subversion. Some of these are closely related to covert influence from a conceptual point of view, as they require some concealment of information to be labeled as deceptive. Yet, by the word "withholding" they are specifically referring information theory, not to a philosophical discussion of deception. As with the work of Ross et al. (2019), they do not discuss manipulation from a philosophical perspective; they only mention it as a description of the information-theoretic model developed by Shannon.

Using game theory as well, Yin et al. (2019) acknowledge the limitation of focusing on the principle of the minority being subordinate to the majority in the context of opinion formation modelling. They create an ABM for understanding the opinion formation process through sociology and psychology, without necessarily depending on a hierarchical influence. Their findings show that unified initial opinion in a group has the effect of accelerating consensus reaching for users' opinions, while controversial topics have the effect of introducing great uncertainty into the opinion formation process. From the perspective of online manipulation, the way users are influenced to form an opinion without necessarily being aware thereof is closer in its process to the works above, given that users are the ones introducing the influence, not the platform itself. They do not even mention the term manipulation or any of its derivatives. Although the propagation of opinion formation could be seen as a kind of manipulation, they are not concerned with such discussion.

2.1.1. Gap in Literature

In general, there is a lack of empirical work around online manipulation (Klenk, 2020). In particular, when I mention the term "online manipulation" here, I am referring to the philosophical perspective of manipulation described above. Given the widespread use of the term for referring to a broad number of topics, some of the previous empirical works have touched upon its discussion superficially. Most of the work done has been conceptual, aiming to define the required conditions for an intelligent system's actions to be considered manipulative. Moreover, there is a predominantly normative focus in the research of AI ethics, based on the ethical considerations of the interactions between agents upon them, such as privacy, exploitation and manipulation (Benn and Lazar, 2022). Even when agreed that manipulation can be problematic, there are nuances that can tilt the moral appreciation towards on side or the other. For example, an individual analysis of users consuming items from a RS in an online marketplace might result in very small consequences for the user's behavior, yet this

could have wider implications when considering the collective: a cascading effect can effectively alter the community's opinion towards a product. It is hard to judge how serious is a manipulative action based on one single individual, since it would require understanding a) how much of the influence affected the decision change, and b) how better off would be the individual had they not been influenced in the first place (Benn and Lazar, 2022). However, many works describe hypothetical scenarios where an individual is the single target of a manipulative action (Spencer, 2019). Both of these questions have a complex and subjective component that limits the degree to which they can be answered via thought experiments or conceptual analysis. In order to gain a quantifiable measure of how effective is a manipulative RS for modifying user preferences, the situation could be modeled as a simulation advancing through time. By including multiple users with different preferences and consumption activity, the results can offer a measure of change derived from the online manipulation, which in turn can be used for objectively assessing the research question of this thesis.

As there has seldom been research done on an account of manipulation that could hold as a condition for any case (Klenk, 2022) and since discussing the definition of manipulation is out of scope for this thesis, it follows that selecting a specific account of manipulation would be best in order to quantify its impact on user preferences. Using Noggle's definition of manipulation as non-rational influence and trickery (Noggle, 2021), some researchers have selected covert influence as a requirement for a manipulative action, since the former bypasses conscious decision-making without the target noticing it and the latter makes the victim believe something other than what the manipulator intended (Susser, Roessler, and H. F. Nissenbaum, 2018). Even though this claim leaves out accounts of manipulation where the intentions or the means of influence of the manipulator are overt, some researchers have already discussed the application of this definition to real-world scenarios (Susser, Roessler, and H. Nissenbaum, 2019). For instance, targeted advertisements are becoming inherent features of online platforms, making it difficult for users to navigate without being completely aware of the manipulative mechanisms. Moreover, RS in particular can exploit the trust that users place on them to guide user preferences without them being aware of it (Bermúdez et al., 2023). This effect fits well with the definition of online manipulation as covert influence. However, it could be argued that users consuming the feed of a digital platform are certainly aware that they could be subjects to manipulation. For example, X (formerly Twitter) has a "For you" and a "Following" feed, where one is curated by a RS based on the user's activity while the other simply displays content by time of publication. This could open the discussion about the necessary conditions of manipulation, which results in a discussion about what is manipulation as a whole. However, since covert influence has been specifically suggested as a condition for manipulation, and since the black-box nature of intelligent systems makes them ideal candidates for hiding (willingly or unwillingly) their functionality, I will use this account to operationalize manipulation in the model.

Despite the significant efforts in understanding and discussing online manipulation, a precise gap remains evident in the direct measurement and operationalization of intentional manipulation effects on user preferences within RS, facilitated by the system's owners using ABM. This gap can be detailed as follows:

1. Lack of direct measurement in RS: to date, no research has explicitly focused on measuring how RS-based online manipulation directly influences user preferences using ABM. Previous works have either approached manipulation from a non-philosophical perspective (e.g. Ashton and Franklin, 2022; Grisse, 2023) or measured the impact of external influences like bots or deceptive content, which, though possibly manipulative, is performed by the participants of the system and not by the system itself (e.g. Kopp, Korb, and B. I. Mills, 2018; Ross et al., 2019).
2. Intentional manipulation by algorithm owners: even though intentional algorithmically-aided manipulation has been researched in past studies, they have been mostly focused on the ethical implications of having owners of digital platforms prioritize certain aspects of the algorithms (Christiano, 2022). They have not been aimed at understanding the actual effect of such changes in user preferences, but at designing a framework where its consequences could be mitigated (Yeung, 2017).
3. Operationalization of covert influence using ABM: there is also a need for a robust methodological frame-

work that can operationalize the concept of covert influence within RS, using ABM. Such a framework would not only help in quantifying the extent of manipulation but also in understanding the dynamics of how such influence is executed and managed within various digital platforms. There are some works that have addressed possible ways to do that so far (e.g. Susser, Roessler, and H. Nissenbaum, 2019; Carroll et al., 2023) but none that have actually implemented them.

Addressing these gaps is crucial for several reasons. Firstly, it would enhance the theoretical understanding of online manipulation, providing a clear and measurable framework to study its impacts. Secondly, it would help in developing more ethical algorithms by highlighting the manipulative potential of intentionally prioritizing certain items in current systems, thereby guiding the development of more transparent and user-centric RS. Lastly, exploring these gaps could lead to more informed regulatory and policy-making decisions, ensuring that digital environments foster trust and fairness rather than covert manipulation.

In terms of the actual implementation, the platforms that have been studied so far are social media platforms such as Twitter. In order to narrow down the research of this study to a specific field with real values, the model will be initialized using real open-source datasets from a book review platform. The problem of manipulation in book review platforms has been addressed in the past (Hu et al., 2012). However, they define manipulation as any vendor, publisher, writer or third-party that posts non-authentic reviews as if they were the customers. This phenomenon is manipulative since the users of a system are attempting to tamper with the algorithm while it assumes that any review is a veracious one. Even though it is different from my view of manipulation as covert influence, it highlights the existence of hidden intentions behind a RS. Moreover, given the widespread use of book recommendation platforms, there have been studies that survey the perception of users from such recommendations, with respect to their own preferences (Burbach et al., 2018). Since books can be catalogued by genre easily, they have been proven great options for building RS around them (e.g. Kibe, 2023; Mathew, Kuriakose, and Hegde, 2016).

2.1.1.1. Agent-Based Modeling for RS

One research method that has been used to capture interactions between users and digital platforms is ABM. ABM has been used to model groups of individuals where each agent has a set of rules for which to act upon. It is ideal for uncovering complex patterns that emerge from an initial configuration and a simulation through time (Gausen, Luk, and Guo, 2023). Regarding the main research question, studying the effects of online manipulation requires this dynamism characteristic to ABM. For instance, if a user receives a single recommendation from a RS, it is not clear what its intentions are nor how is the user being affected by it. Only by giving the system some time to develop is it able to guide user behavior. Moreover, there are challenges for these systems when they lack prior information from a user (Liao, Sundar, and B. Walther, 2022).

In order to build a RS for book recommendations using ABM, I am following the general model approach proposed by J. Zhang et al. (2020). They build ABMs to study the impact of different factors on the dynamics of RS's performance. In particular, they are focused on simulating the consumption strategies of how users explore and consume items in RS. They define the aspects that each class of agents should have and the processes taken at each timestep, which basically include updating the recommendation engine with the activity of the users based on their consumption of the items and the profiles of the nearest neighbors (i.e. users with the most similar profiles). Since one of the main contributions of this thesis is to understand how covert influence is affecting user preferences in a RS, their work provides the methodological design required for modelling how users consume books after being given a set of prioritized recommendations. Furthermore, their model takes into consideration the users' preferences in order to measure their change after the simulation is run. They use public sample datasets from Netflix and Yahoo! Music to exemplify their study. Their findings suggest that the relevance of item recommendations decreases as the number of users of a RS increases. This might sound counter-intuitive but they attribute such behaviour to the over-reliance of users on these systems for discovering new relevant items. As a final remark, even if they follow a very similar configuration as the one proposed in this thesis, they are concerned more with the algorithmic characteristics of the RS, disregarding any potential

interference due to a manipulative action.

Traditionally, the three most common types of RS are content-based, collaborative filtering and demographic filtering (Liao, Sundar, and B. Walther, 2022). Content-based filtering is focused on the similarities between users' preferences and items, while collaborative filtering tries to find users with similar characteristics and predicts items in common. For instance, J. Zhang et al. (2020) model a collaborative filtering type of RS because of the interactions amongst agents. Demographic filtering is less used and it focuses on considering a user's personal characteristics for the predictions. There are resources available already built for simulating RS and their interactions with users (Ie et al., 2019). However, they provide little flexibility for configuring the ways in which items are prioritized, which is the main objective of this research. Therefore, a separate model will need to be simulated.

2.1.1.2.Manipulation in Book RS

The RS that was chosen for this thesis is based on the GoodReads platform. This platform is a social cataloging website that allows its users to search for books in a large database. It has more than 125 million users and 3.5 billion books (as of 2022[2]). Its members can interact with these books by reading any information associated to the book (such as summary, author, year of publication, etc.) and by performing actions that update both the user's and the book's attributes (such as review, add to wish-list, get recommendations, etc.). Additionally, the website offers a social network component in the form of peer-to-peer interactions, where users can follow other users and get notifications about their platform activity. Even though the features of the platform extend beyond these functionalities, one of their core products that have been researched is its recommendation engine (Wan and McAuley, 2018). The system allows for users to get a list of book recommendations based on their history of book consumption and interactions with other users.

Regarding manipulation from book review platforms, there has not been much work done about it. Most of the research about problematic manipulation (even when the term is not used under the philosophical lens) has revolved around polarization and radicalization, using communication channels that involve exposure to news (Martin and Yurukoglu, 2017). Given the large-scale impact of small time windows, it is morally relevant to focus on these topics. On the other hand, books are information recipients that tend to have longer consumption periods. For instance, the average adult in the U.S. consumes around 12 books per year, but the median is 4 (Perrin, 2016). These numbers suggest that there is a non-negligible proportion of the population that reads many books, pushing the average much higher than the mean. Under these circumstances, it is harder for a manipulator to take advantage of the target readers. Additionally, books tend to have better reviewing mechanisms (either through other peers or through an editorial), which reduces the probability of unethical content being published. Needless to say, this is not a rule and there are many exceptions. Furthermore, I believe it is relevant to study manipulation from book recommendations because of two reasons: 1) there are many people who consume books on a high frequency basis (i.e. at least one per month), and 2) given the higher degree of trust that users have on books, manipulative actions can require less exposure to become effective. Regarding the former, the dataset that was chosen for this thesis is a clear example of a vast amount of people reading books on a continuous basis (see section 3). As for the latter, some studies have found that there is a high reliance on books as opposed to social media or media in general (Kousha, Thelwall, and Abdoli, 2017). For instance, in 2017, around 78% of U.S. adults thought libraries were a reliable source of information to learn new things[3], whereas 34% said they trusted the media in 2022[4]. If one has more trust on a specific source of information and that source ends up containing false or dubious claims, then one might be quickly mislead to believe in false information.

the moral justification of choosing book recommendations as an experimental environment, there has been research done on the actual RS of these type of platforms, with GoodReads in particular being one of the most

[2] Goodreads: A Platform for Readers and Authors
[3] Most Americans – especially Millennials – say libraries can help them find reliable, trustworthy information
[4] Americans' Trust In Media Remains Near Record Low

popular ones (Martin and Yurukoglu, 2017). The open access nature of their large database allows for analysis and simulation of their RS.

An important component for making predictions related to books is the book label or genre, since this represents the content of the item which can be used for any content-based RS. In particular, GoodReads uses a collaborative voting mechanism for determining book genre tags: each user can vote which genres does a book belong to. The actual genre is seen as the one which has the majority of votes from the users. Some researchers have used these tags to try to predict the personality trait of users who tagged books (Annalyn et al., 2020), while others have used them to test predictions as recommendation lists in controlled experiments (Liu, M. Xie, and Lakshmanan, 2014). However, the previous research that has touched upon manipulation has been mostly focused on the non-philosophical view of manipulation, referring to how users can manipulate the collaborative-filtering RS by flooding the platform with biased reviews (Wijnhoven and Bloemen, 2014). They have found that it is possible to make a book popular or unpopular through the ratings and interactions given. Yet, this form of manipulation has been addressed by both platform designers and academics, reducing the number of manipulative reviews considerably in the past years by applying constraints on the reviewing mechanisms of platforms (Ivanova and Scholz, 2017).

Some other type of "hidden influence" that has been researched from book RS is related to how information is displayed to the user. This is particularly important for this research since the consumption of items given a list of recommendations in the model varies with the sorting of the items. Thus, the display has an important impact on the results of the model. There are primarily two possibilities of action after a user requests a recommendation: 1) they consume one of the recommendations, and 2) they rate the consumed recommendation. The former refers to the probability that a user will consume an item, depending on the attributes of the item (e.g. the image displayed, the naming convention, the rating of the item, etc.) whereas the latter refers to the probability that a user will assign a certain rating or review to the item (e.g. number of starts from 1 to 5, number between 0 and 10, textual review, etc.). Both steps have been shown to be influenced by the ways in which items are shown. For instance, some researchers have used Discounted Cumulative Gain (DCG) to give an estimate of the relevance of particular items in a list of recommendations (Liu, M. Xie, and Lakshmanan, 2014). They assume that items that have a higher ranking (i.e. appear higher on a list) are more useful to the users. Other researchers have shown that giving additional information of an item to a user when they offer a review can nudge the user towards a specific sentiment (Cosley et al., 2003). For example, when a GoodReads user submits a review for a book, they can be influenced by the average rating given to the item if they view it at that moment. Moreover, if users are prone to consume items based on the ratings received, a platform could take advantage of this and decide for which items to show their average rating upon reviewing. Both of these studies show that users are susceptible to be influenced by visual queues, without necessarily being aware of it. Therefore, the manipulation can be exercised via the display action from the platforms.

2.2. Manipulation of Online Book Users

The following section condenses the theoretical background into the operationalization of the model. It discusses the relevant literature to map concepts to parameters and attempts to answer the question: how do RS manipulate covertly? It is divided into the operationalization of covert influence as an input to the model and the operationalization of harm as the output measure of the model. Additionally, the operationalized parameters of social influence and transparency are discussed. The theoretical background behind many of the concepts is addressed in the appendix A.

2.2.1. Covert Influence: a Change in Priorities

Let us summarize the covert influence account of online manipulation as the definition from Susser, Roessler, and H. Nissenbaum (2019): using technology to covertly influence how people make decisions. After stating the definition, the authors proceed to justify its applicability by using three conditions:

1. Constant digital surveillance from digital platforms
2. Real-time learning and updating of digital platforms' profiling
3. Technological transparency of digital platforms' interfaces

From the perspective of RS in particular, these three conditions hold, since the algorithms are constantly being updated to match the user's profile (Grisse, 2023) - conditions 1 and 2 - and there is a degree of ignorance of how these algorithms are mapping preferences to digital profiles (Hall, Johansson, and Strandberg, 2012) - condition 3.

As mentioned before, broadly speaking, RS are filtering and ranking tools for navigating vast amounts of information. When providing recommendations to a user, there is an implicit use of a prioritization scheme, since there needs to be a way for the RS to assign a numerical value to an item so that it can be compared with the rest of the items, in order to get the content sorted by such a value. This priority can take many forms, from similarity between the content to similarity between the two users' preferences. Despite the many methods that can be used, users have some expectations of which factors are affecting the recommendations (Adamopoulos and Tuzhilin, 2015). Such expectations are aligned with the user's view of their own preferences, thus users usually expect recommendations to be closely aligned to their own preferences (Kotkov et al., 2018), which in terms of RS means that the digital profile of the user is used as the primary factor for computing recommendations. Even though users can be benefited by some degree of unexpectedness from the algorithms (Bouneffouf, Bouzeghoub, and Gançarski, 2012), the novelty relies on small doses of randomized information, which is different than a dominating factor that is not what the user expects (Grisse, 2023). Moreover, when the attention of the user is drawn towards multiple different items where they can make a rational comparison, the degree of manipulative action is reduced. However, the way in which RS display information about their recommendations has a high impact on the user's consumption: when recommendations are shown as a list, the probability of consuming an item decreases exponentially with the items of the list (Carare, 2012). This behavior has been studied using a DCG approximation for measuring ranking accuracy (Liu, M. Xie, and Lakshmanan, 2014), yet the metric is specifically used for estimating the accuracy of information retrieval results, not taking into account the actual perception of a user. Hence, I chose the approach proposed by Carare (2012) and expanded by J. Zhang et al. (2020), where the probability of item x ranked in position i being consumed decreases exponentially based on the following equation:

$$P(x_i) = k \cdot \alpha^{-i} \tag{2.1}$$

Where α is a parameter that should be larger than 1 in order to simulate the exponential decay and k can be estimated as $k = \alpha - 1$ if we assume that the sum of the probabilities of all items should be 1 (J. Zhang et al., 2020), which makes sense as a recommendation list has mutually exclusive elements. Figure 2.2 shows a graph of the equation when $\alpha = 2$. The first element on the list would have a probability of 50%, the second 25% and so on. After the 10th element, the probabilities become very low. Even though this is a good approximation of reality, it is not the only one: other probability distribution functions have been used to estimate the attention given to each item. For instance, Sar Shalom et al. (2016) use the Boltzmann distribution to take into consideration an increase in probability for the first few items, followed by an exponential decrease. This is justified by the fact that users want to have some points of comparison before making a decision and not choose blindly the first option given. However, their study was made for video game players going into a virtual store, where the novelty of items played a very important role. My assumption for choosing the fist approach is that book readers care less about the newness of an item than about its attributes (G. Zhang and Sun, 2012), since there have been many more books available for a much longer period of time.

If the best results returned by a RS are being prioritized by a specific motive, then the user could be induced into making a decision that would not be on their best interest. This assumption lies at the core of the covert influence

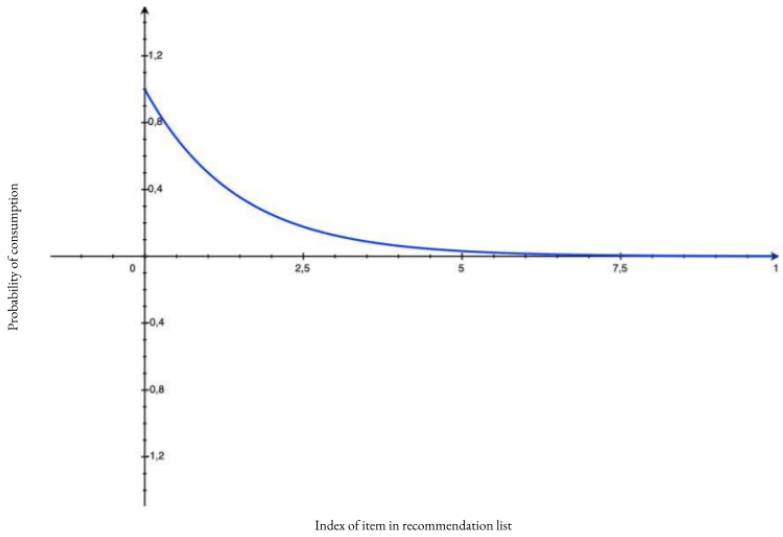


Figure 2.2: Graph of equation 2.1 for values when $\alpha = 2$

thesis described above. Therefore, two requirements can be used to form a framework for the operationalization of covert influence:

1. Prevalence of hidden priority: there is a factor(s) that is affecting most of the recommendations that a user is receiving and it is unknown to the user
2. Biased display of choices: the display of choices from the RS is biased towards the hidden priority from (1).

For the specific case of this research, users will get a prioritized list of recommendations based on different strategies. When there is no hidden influence, the priority is solely based on the user's preferences. If there is a change in priority that modifies the dominating factor for getting recommendations and displays them at the top of the list, then a user would be biased towards those items. Nevertheless, the action of consumption is a probabilistic one; it is not guaranteed that the top item would be consumed by the user, specially when there is some form of explanation of the intention behind the recommendation (Grisse, 2023).

As discussed before, a RS is a filtering and ranking method for navigating large amounts of information. A system that is not hiding any form of intention should perform its functions solely based on the RS algorithm (e.g. content-based). Assuming that the model that the algorithm has of the user is accurate with respect to the user's preference (see Burbach et al. (2018)), such recommendations would be aligned with the user's profile. When there is a hidden influence in place, there can be modifications to such ideal mechanism, where options are shown not in the best interest of the user but in a way that maximizes a particular metric for its designers. In other words, items are prioritized differently than if they would have a pure numerical similarity with the user. Even though this prioritization can be inferred by a black-box algorithm such as a neural network (Wei et al., 2021), the focus here is on the designer's intentions.

In terms of the actual model, this influence is modelled by adjusting the "priority" value of the items when initializing them. The strategies are detailed in section 3. When there is no priority, the books are sorted by the similarity between the items' vector and the user's, in descending order. When there is a random priority, some books will have an initial maximum value, which guarantees their insertion high in the recommendation list. If the user decides to consume them is a question that is closely related to nudging (see appendix A). Finally, when

the priority is based on a book genre, all books which have the specific genre as the maximum category - books have a vector with values per category - will be given top priority, resulting in a potential bias towards this type of books.

It is important to distinguish here between determining the existence of a quantifiable effect on users preferences and the magnitude of such effect. If we think about the former, then we risk falling into a circular reasoning: it will almost certainly be the case that users' preferences will change if the priority changes, given that the consumption of items is directly correlated to their position on the results. However, the model is built to measure the latter. Moreover, the existence of agent decisions taken from probability distributions will allow for an experiment that should be hard to estimate beforehand.

While this form of manipulation is explicitly built into the RS, there are other ways in which manipulation could manifest. For instance, many platforms deploy models that are constantly learning about user activity. There are subtle changes in the algorithm's learning motivations that could pass undetected by their designers while changing the behavior of the users (Carroll et al., 2023). Even though this is a clear example of manipulation as hidden influence, the factors affecting the RS are inherent to the technology being used, which makes them very hard to measure (Carroll et al., 2023). Establishing a relation between the RS output and the behavior change in the user becomes a hard task without explicitly asking the user about their experience. This form of manipulation is closely related to the reflective transparency definition (Andrada, Clowes, and Smart, 2023), hence it is left out of scope from the model as it would be hard to simulate without real users.

2.2.2. Harm: a Change in User Preferences

By looking at the theoretical framework from this research, harm is part of the outcome of a manipulative action (Jongepier and Klenk, 2022). Thus, harm as studied here is frustration of self-interest and undermining of autonomy; it is a consequence of the action. This could imply that manipulation is a "success concept", in the sense that only when the target does as the manipulator intended, one can speak of the target being manipulated (Wood, 2014). For instance, Richens, Beard, and Thompson (2022) define an action being harmful only if the person being affected would have been better off if the action had not occurred. There are two considerations that arise if we use this operationalization: 1) there is no objective definition of what "better off" means, and 2) it can lead to a beneficial change if read as its counterfactual: an action can benefit an individual if they are better off had the action occurred. Even though the authors proceed by stating the expected value of a harmful event from an individual's actions, it is based on the assumption that "counterfactual reasoning is necessary for harm aversion". In other words, the only way in which a user can avoid this type of harm is by reasoning about the hypothetical case of not being affected by the action. However, from the previous section we know that that is not possible, since the influence exerted by the manipulator is covert and hidden from the user; there is no way for the target to assess the scenario where the manipulator is not manipulating them if they do not know that there are some hidden intentions behind the manipulator. Nevertheless, we can still consider this definition if we assume that a) a change in user's preferences need not be a requirement for an action to be manipulative, and b) we are not concerned with the potential benefits of manipulation.

Additionally, the action itself could still be regarded as manipulative even when the attempt is unsuccessful since the disrespect of the user's autonomy is prevailed (Susser, Roessler, and H. Nissenbaum, 2019). From the operationalization of manipulation described above, it becomes clear then that having a hidden intention is itself an autonomy-undermining act which aims at frustrating the self-interest of the user by covertly guiding them towards an outcome, regardless of the choice from the user.

Let us now narrow down the operationalization of harm from general manipulation to RS in particular. If we understand harm as a disrespect to autonomy, the a RS can be harmful by undermining the autonomy of a user when it has a hidden agenda. This could be achieved by taking advantage of the decision-making vulnerabilities of the user and prioritizing items that the user might think they are on their best interest, resulting in a reduced freedom of choice for the user (Grisse, 2023). If this is the case, then a user might be prone to consume items

that are not aligned with their preferences thinking that they are. This would clearly represent a shift of decision-making capabilities as the user would not have the full rational capacity to decide if they should or should not consume an item based on the reasons why it was shown. Moreover, given that these algorithms are constantly updating the digital profiles that they form of their users, the consumption of an unrelated item would modify their digital profile in such a way that subsequent recommendations are shifted slightly towards the consumed item's characteristics (Hall, Johansson, and Strandberg, 2012), potentially changing the user's behavior without them noticing it. Therefore the operationalization of harm for this research is based on the following assumptions:

1. Disregard for successful consumption: the actual consumption of a prioritized item in a RS with a hidden priority is not a necessary condition for a manipulative action, only the exposure to the hidden strategy
2. Harm as preference change: the degree of harm caused by a manipulative RS can be measured as the change in the user's preferences after being exposed to prioritized items

With this assumptions in mind, we can proceed to build a model where each user's preferences are measured before and after they are being manipulated with multiple prioritizing strategies. Depending on the magnitude of this change, it can be quantified how harmful was the strategy for the user.

2.2.3. Social Influence

The use of large social media platforms has given rise to extensive digital social networks that exhibit complex behavior as the network grows larger. One of such effects is that of social influence, which is defined by Cercel and Trausan-Matu (2014) as the "power exerted by an individual a on an individual b, having the effect of change on the opinion of the individual b". This influence has multiple properties and can be measured in different ways, most of which are related to network metrics (Peng, G. Wang, and D. Xie, 2017). For instance, a platform like Twitter can be modelled as a directed graph where each node represents a user and each directed edge represents one user following another (Himelboim et al., 2017). The magnitude of such link could be measured by the interactions between followed and follower, using likes, retweets or comments as the data source. The capacity to study social networks like this has given rise to social influence analysis, where specific methods can be used for measuring how opinions are propagated in a network - ABM, for instance (Yin et al., 2019).

Social influence analysis is a broad topic on its own and it is out of scope for this thesis to explore its application to the present model. However, a social platform that contains a recommendation product (e.g. GoodReads) manifests influence from two sources: the RS and the social network. Moreover, the magnitude of one of such influences could have a positive or a negative effect on the other (Peng, G. Wang, and D. Xie, 2017). Therefore, it seems relevant to include both effects in the model. Ziegler and Golbeck (2007) have already shown empirically the tight correlation of trust between users and their interest similarity, where the more similar two users are, the more they trust each other and vice versa, without necessarily a causal relationship. This result is closely related to RS since a collaborative-based RS takes advantage of peer-to-peer interactions to come up with recommendations. Since I am focusing on a content-based RS, the social influence exerted upon some user will be derived from the consumption history of the most similar user. The model will simulate this relationship as a "following-follower" one, where a user can receive notifications when someone they follow has read or reviewed a book. Given the close trust and similarity between these users in terms of preferences, the consumption history will have more relevance than the recommendations from the platform.

2.2.4. Transparency

One way to make a system overt is by making it transparent. Nevertheless, transparency can be understood from different perspectives and can have many implications. Given that the purpose of this research is not to characterize transparency, the term will be used for operationalizing the shift from covert to overt by the proportion of the population who is exposed to the intentions of the algorithm, as it breaks the first condition of the covert influence operationalization described in section 2.2.1. Therefore, it is briefly discussed below.

In particular, technological transparency has been discussed from a sociological perspective - related to the acclimation of humans with technology to a point that they perceive it as clear (Van Den Eede, 2011) - and from an ethical one - related to the ways in which technology can or should be open to its users (Andrada, Clowes, and Smart, 2023). The former helps to explain why the familiarity of users with technology is a facilitator of manipulation as platform designers can exploit the inherent belief in transparency from the technologies. We will now focus on the latter.

According to Andrada et al. (Andrada, Clowes, and Smart, 2023), transparency can be divided into two categories: 1) reflective transparency, our ability to peer into a technology, and 2) transparency-in-use, our ability to see through a technology. In other words, the former is focused on the inner workings of a system (e.g. the ways in which a machine learning algorithm arrives to a conclusion) while the latter is focused on the more abstract purpose of the technology (e.g. the goals of a machine learning model designer or owner). There is a clear map of these two categories to the intent aspects discussed above. However, more emphasis has been put on the research of reflective transparency, through the use of better explanations of RS predictions (Meske et al., 2022). Yet, the sole act of providing an explanation can also be considered a form of manipulation (H. Wang, 2022), thus resulting in a complex situation with a blurry boundary of demarcation.

Therefore, for the sake of simplicity and for the purpose of this research, transparency will be defined as transparency-in-use, where users will be given or not the actual intention of the RS designers to influence their decisions. For instance, when a third-party pays for an item to be prioritized, users could be shown the item with a higher ranking but under a "sponsored" label. It will be assumed that this actions would make the intentions of the manipulator overt, allowing for a comparison with hidden influence. I refer to users who have not been exposed to the transparent mechanism as *ignorant* or *naïve*.

3

Method

3.1. Model

The following section contains the details about the ABM model. After doing a quick overview of the model, I proceed by explaining each component through 5 subsections, covering a description of the dataset, the verification and validation analysis, the experimental set-up, the sensitivity analysis, and the rationale behind each experiment. Assumptions and decisions are justified accordingly, as well as a description of the implementation of the modeling cycle (Dam, Nikolic, and Lukszo, 2013).

3.1.1. Overview

The general purpose of the model is to understand how user preferences of book categories change when the users are exposed to prioritised and non-prioritised recommendations from a RS during the course of 1.5 years.

In order to exemplify the ways in which users become aware of such prioritisation, there is a percentage of the users who are *naïve* or ignorant about the real intentions of the algorithm. Additionally, users can be subject to reviews of books consumed by their peers, resulting in an overt influence exerted upon them, not by the algorithm but by the environment.

A visual description of the flow of actions for a user is shown in figure 3.1. Let's assume that Sam is an avid reader. She reads two books per month on average and she is a frequent user of GoodReads, going into the platform everyday, even if she does not wish to get a recommendation for a new book. After finishing a book, Sam normally requests a new book from the "recommended for you" section in the platform, which returns a list or books that she might like. Since she trusts the recommendations given by the algorithm, she does not take much time scrolling through the whole list and she normally picks one of the top choices. After adding the book to her GoodReads list, she goes to the closest bookstore and buys it. After a couple of weeks, when she finishes it, she returns to the list and adds a review about the book. Each action has a corresponding method in the model (see figure 3.3).

As suggested by J. Zhang et al. (2020), the model has three elements from a conceptual perspective: user population, item population and recommendation engine. Below is an overview of what each of these elements refers to:

- User population: the users that consume recommendations. In the case of this specific implementation, this would refer to the readers.

Figure 3.1: Example of single step flow for an agent

- Item population: the items that are recommended. In the case of this specific implementation, this would refer to the books.
- Recommendation Engine: the RS that predicts items for users. In the case of this specific implementation, this would refer to the platform (i.e. GoodReads).

Figure 3.2 shows an overview of the components and its functionalities. The recommendation engine is in charge of controlling the flow of information and the interactions between users and books. The boxes represent the components and the arrows represent the flow of actions. Dotted arrows are used for optional actions. Most of the relations between users and books are managed by the recommendation engine, except for the cases where a user reviews a book. The reason why the "update" arrows are pointing in opposite directions is because a user instructs the recommendation engine to update its attributes once it consumes an item, whose attributes are then updated by the recommendation engine. In a way, the item agents are idle, in the sense that they are not dynamically changing or affecting the inner workings of the engine. It is the users that are updating the engine with their behaviour. There are mainly two types of actions: retrieve and update. The former refers to obtaining some information from the model while the latter refers to changing values within the model, similarly to the read and write functionalities in file permissions. The only reason why I did not choose that naming convention is because the retrieval of information involves many calculations that seem to be more than a mere query.

3.1.2. Dataset

The dataset used is a book reviews dataset from the Goodreads platform[1]. It contains 2,360,655 books, 876,145 users and 228,648,342 interactions, separated into multiple CSV files. The dataframe of the model was created by loading and pre-processing two datasets: `goodreads_interactions.csv` and `goodreads_book_genres_initial.json`. The former contains all user-item interactions while the latter contains information about books. Table 3.1 shows the columns and description of each column for both datasets.

[1] GoodReads dataset repository

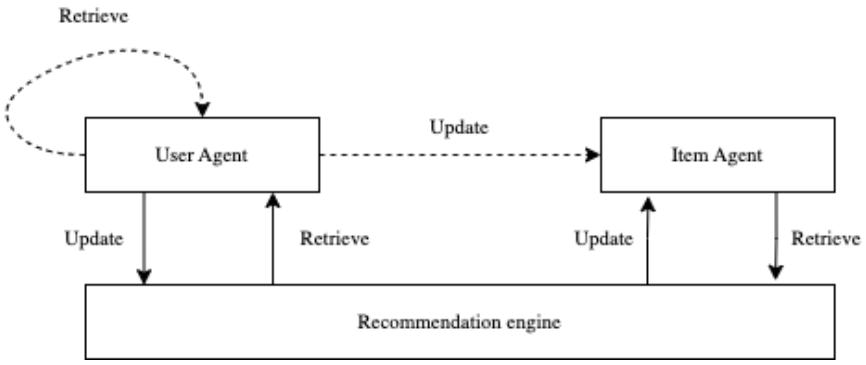


Figure 3.2: Overview of model components and methods

Table 3.1: Description of Goodreads datasets columns

Column name	Dataset	Description
user_id	Interactions	Unique user identifier
book_id	Interactions	Unique book identifier
is_read	Interactions	Boolean indicating if the book was read or not
rating	Interactions	Rating given by the user as integer between 0 and 5
is_reviewed	Interactions	Boolean indicating if the book was reviewed
book_id	Book genres	Unique book identifier
genres	Book genres	JSON object with genres as key and count of occurrence as value

Two Pandas dataframes were created from these datasets. Below is the list of book genres:

- fantasy
- non fiction
- mystery
- young adult
- graphic
- thriller
- paranormal
- romance
- history
- biography
- historical fiction
- comics
- poetry
- crime
- children
- fiction

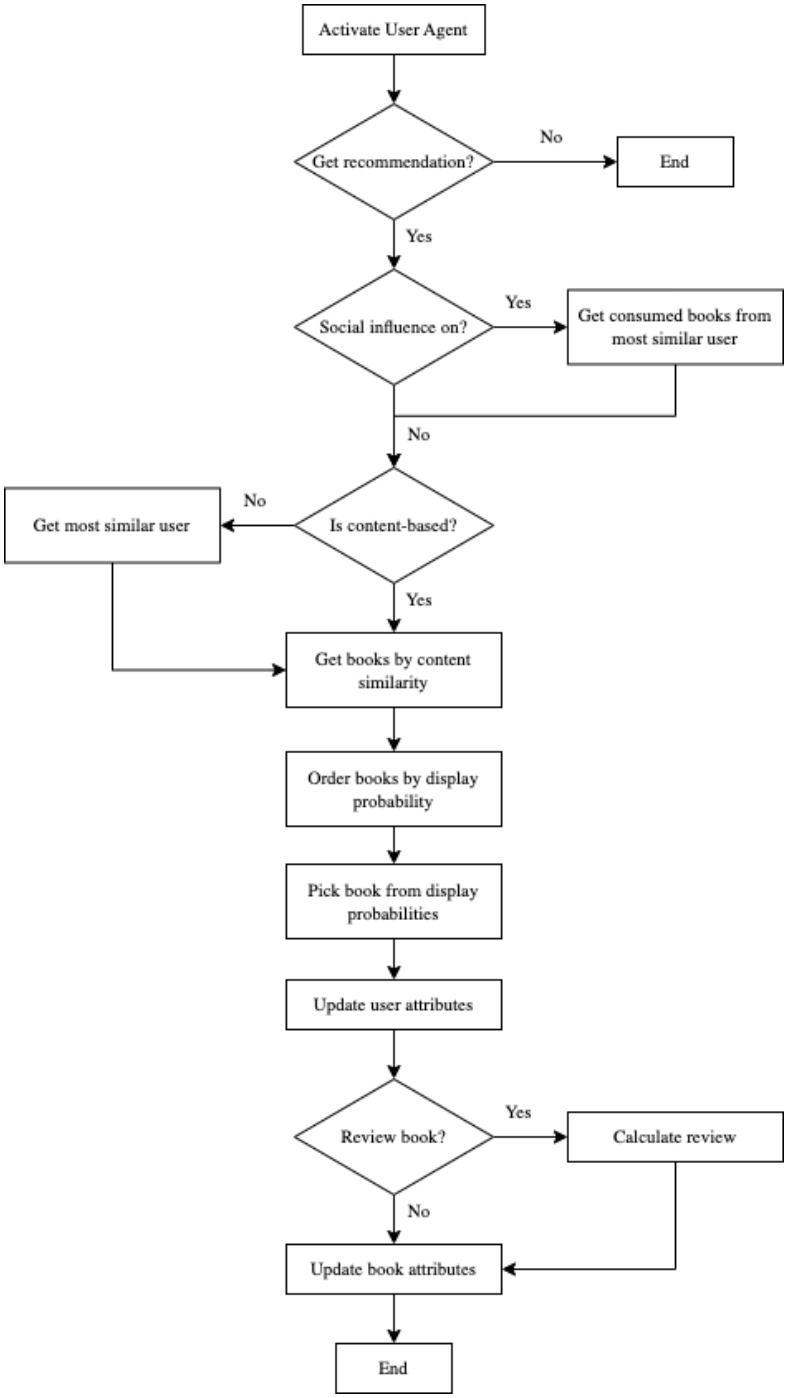


Figure 3.3: Flow diagram of one simulation run step

3.1.2.1.Data Pre-processing

The pre-processing of the model dataframe consisted on the following steps:

1. Sampling: extracting a subset of random users from the Interactions dataframe
2. Processing of raw users:
   (a) Filter thresholds: extracting users that read books between the threshold values: 5, 20 and 50 (see section 3.1.3)
   (b) Sampling personas: sampling the same number of users per reader persona (see section 3.1.3) to match initial number of users parameter
   (c) Add persona: add reader persona as the corresponding category: low, mid or high
   (d) Add ignorance: making the same proportion of users ignorant in all cohorts of reader personas
3. Ratings transform: normalizing ratings to float values between 0 and 1
4. Book filtering: selecting from Book genres the books that were part of the interactions that resulted from the sampling of users
5. Genres reformat: reformat genres to a dictionary with each separate genre as key and total count as value
6. Merging: merging Interactions and Book genres by book ID as an inner join

The model dataframe contains all information relevant for the model. Two more dataframes were derived from it, one belonging to processed user data and another to processed book data. This way the input for each agent model could be managed separately.

The pre-processing of the books dataframe consisted on the following steps:

1. Group by book: segmenting the model dataframe by book ID
2. Aggregating values: perform aggregations of columns based on the following functions:
   (a) is_read: sum, number of users that read the book
   (b) is_reviewed: sum, number of users that reviewed the book
   (c) rating: mean, average rating received
3. Genres transform: transform genres into a "vector" column as a Numpy one-dimensional array with each genre as column and the count of occurrences as row
4. Calculate priority: add a "priority" column with values between 0 and 1 for each book based on the parameter of the model. The possible options are discussed in section 2.

The pre-processing of the users dataframe consisted on the following steps:

1. Boolean genre column: adding a column with a boolean value if the genre count is larger than zero
2. Group by user: segmenting the model dataframe by user ID
3. Aggregating values: perform aggregations of columns based on the following functions:
   (a) is_reviewed: sum, number of total books reviewed
   (b) is_read: sum, number of total books read
   (c) rating: mean, average rating given
   (d) book_id: list of book IDs interacted with
4. Reading probabilities: calculate read probability for each user based on their user persona (see section 3.1.3)
5. Genres transform: transform genres into a "vector" column as a Numpy one-dimensional array with each genre as column and the count of occurrences as row

6. Book score calculation: adding a book score to each book ID by normalizing the user's vector and calculating the cosine similarity for each book in the book_id list as suggested by Kibe (2023), where the cosine similarity between two vectors A and B is defined by equation 3.1:

$$S_C(A, B) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\|\|\mathbf{B}\|} \tag{3.1}$$

7. Book ID transform: transform book_id into a dictionary with the book ID as key and the cosine similarity as value
8. Social influence: if toggled on, get a list of the 5 followed users per user, based on vector similarity
9. Cache similarities: for content-based RS, pre-calculate similarities with top books in order to avoid running the calculations for all users in all iterations

3.1.3. User Personas

As mentioned before, the degree of consumption of items in RS does have an impact towards how much can users be influenced. Not all users consume books with the same frequency or ratio. According to Pew Research, the average number of books read per adult in the U.S. in 2015 was 12, while the median was 4 (Perrin, 2016). From another poll, it was concluded that less than 1% of the population reads more than 50 books per year (Van Dam, 2024). Moreover, while the data from GoodReads might suggest that there are users that read many more books[2], this is often exaggerated by the people who participate in those polls or by the way in which users demonstrate that they have read a book; there is no formal proof, so anyone who clicks on "read" is counted as someone who fully read the book. Taking all of this information into account, I came up with three user personas for dividing the population into different reader personalities:

- Casual reader: 0-5 books per year
- Selective reader: 6-20 books per year
- Avid reader: 21-50 books per year

The reason why I capped the last category up to 50 books was to avoid any outliers of users who might have "read" thousands of books, thereby introducing biases to the data. Lastly, when initializing the model, I made sure that I was always selecting an even amount of users per category, so that the results were nicely distributed among each user persona.

These personalities are operationalized in the model by modifying the probability that a user will ask for a recommendation. In other words, they determine how often are users interacting with the model. Given that in this model, one step corresponds to one day, the associated probability is the middle-point number of books that each persona consumes per year divided by the number of days in a year. For simplicity, I rounded the number of days to 360 and I rounded each probability to four decimal points. Therefore, the reading probabilities of the default threshold values (i.e. $[5, 20, 50]$) for each reader persona would be the following:

- Casual reader: $\text{round}(\frac{5}{2} \cdot \frac{1}{360}, 4) = 0.0069$
- Selective reader: $\text{round}((\frac{20-5}{2} + 5) \cdot \frac{1}{360}, 4) = 0.0347$
- Avid reader: $\text{round}((\frac{50-20}{2} + 20) \cdot \frac{1}{360}, 4) = 0.0972$

Regardless of the number of steps, these probabilities would yield the expected value of books read during the time window. In every iteration, each user agent gets a random value between 0 and 1 (without seed). If this value is lower than the read probability, then the user gets a recommendation list. Otherwise they skip the current iteration. This behavior will return the average consumption after many steps. More details about the logic can be found in section 3.1.4.2.

[2] GoodReads poll - How many books do you read in a year on average?

3.1.4. Model Set-up

The general simulation framework proposed here follows the functioning of a traditional RS, where each user is able to get a list of recommended items from the system and consume one if they wish to do so. Afterwards, they can review the item so that other users are able to view the item's rating or the opinion of the user if made public. Since the purpose of the model is to determine an evolution of user preferences, there is a time-step function in charge of advancing the state of the simulation in each tick. For each step, all agents are activated randomly and proceed to decide if they should interact with the RS or not. Such decision is a probabilistic one depending on their user persona. If they happen to request a new recommendation, the items that they have already consumed are removed from the list when they get a set of recommendations, as it is rarely the case that a user would consume the same book twice within the same platform. Since books are items that are consumed mostly on a weekly or monthly basis, the size of the steps was chosen to symbolize one day. From the perspective of the items, their evolution was not part of the scope of this thesis, thus the only relevant information became the history of consumption. Even though other researchers introduce temporal variables to the model (e.g. item lifespan, item creation or item destruction (J. Zhang et al., 2020))), GoodReads is a platform that maintains its items static, as its core product lies on having the largest possible book catalogue. Likewise, from the perspective of the users, there is no temporal change in the number of agents through time, under the assumption that using a sample of 150 users is already too small for a platform the size of GoodReads.

In a real RS, the recommendation engine that is in charge of generating the recommendations would be typically a machine learning model, taking in the profiles of the users as input and returning a list of items based on a measurement of each item's probability of consumption. Since these models can quickly become very complex and since they are not the object of study for this thesis, I followed the existing approach in the literature for simulating such generation (Kibe, 2023). In this case, the only information relevant for the recommendation engine is a vector representation of the user preferences, determined by the item attributes in their consumption history. In the case of a book RS, if a user read more books about thriller than about history, then their vector of preferences would be shifted towards this genre. As books also have a vector representation of their own attributes, I can use a similarity score between a user and a book to obtain how close they are (see equation 3.1). Some RS models use a feedback mechanism to update the digital profiles that they might have about a user. The feedback could be direct (i.e. providing a review) or indirect (i.e. triggering an event, such as consuming a book or searching for a book). For the present case, the vector of preferences from a user gets updated when a user consumes an item, effectively simulating the feedback needed by the engine to modify its predictions.

The actual implementation of the model does not have the components described in section 3.1.1 clearly separated, since some functionalities that belong to the recommendation engine capabilities might be performed within the item or user agent models. Table 3.2 contains the main components of the user agents, where subtable 3.2a describes the initial values of each variable as provided by the user dataframe and sub-table 3.2b describes the main functions. Furthermore, table 3.3 has the same information but for the items. For user agents, the main difference between books and books_consumed is that the dictionary in books contains all books, including those before and after the simulation, while books_consumed only contains a list of the books that have been consumed after the simulation begins. This is only for making easier the access to a user's consumption history after the run ends.

3.1.4.1. Model Initialization

From the general model perspective, the initialization parameters can be read in table 3.4. Additionally, there is a Results object that manages the creation of directories when each simulation starts, as well as the storing of CSV files containing the results of the simulation. This way the data becomes accessible for data analysis without requiring any changes to the main code. There are multiple variables whose purpose is to speed up the initialization and to maintain uniformity in the loaded data:

- dummy: mostly for testing purposes. Uses a very small pre-loaded CSV file with user-item interactions

Table 3.2: User agent model set-up

(a) Variables

Name	Description	Initial Value
unique_id	Unique identifier for agent	unique_id
model	RecommenderSystem Mesa model	RecSysModel
user_id	User ID from original dataset	user_id
books	Dictionary with books interacted and similarity scores	book_id
n_reviews	Number of reviews given by user	is_reviewed
mean_rating	Average rating given by user	rating
n_books	Number of books read	is_read
vector	User preferences vector from genres distribution	vector
read_proba	Probability of reading an item	read_proba
ignorant	Whether the user is ignorant of the manipulation or not	ignorant
similarities	Dictionary with top books and their similarity scores	similarities
should_update_similarities	Whether the similarites should be updated	False
books_consumed	List of books consumed after simulation starts	empty list
following	List of followed users for social influence	following

(b) Methods

Name	Description
get_review_probability	Calculate probability of reviewing an item as number of reviews divided by number of books read
find_most_similar_agent	Find the most similar user agent by computing the cosine similarity between their vectors
get_social_influence_books	Get list of consumed books by followed users if social influence is True
get_top_books	Get top n books by vector similarity. If content-based, then calculate against all books. If collaborative-filtering, then calculate against most similar agent's books
get_recommendations	Get a fixed number of recommendations by combining the result from get_social_influence_books and get_top_books, depending on the RS type. Items are ordered and displayed according to the inverse exponential equation 2.1, excluding books from the books_consumed list
pick_choice	Pick a random choice from the recommendations weighted by the inverse exponential of their order
update_similarities	Update similarities of content-based recommendations when a book has been consumed, in order to avoid calculating values in each iteration
update	If item was consumed, update vector and increase the list of books read with it

- df: user-item interactions dataframe, with the same format as the one described in section 3.1.2.1
- df_items: processed items dataframe. If it is not given, the model attempts to generate it from df
- df_users: processed users dataframe. If it is not given, the model attempts to generate it from df and df_items

Since the pre-processing of items was independent of the experiment, both df and df_items were passed as constant values when there were multiple runs of the same configuration. However, the pre-processing of df_users was dependent on the experiment. Therefore, this dataframe was generated from df and df_items in each exper-

Table 3.3: Item agent model set-up

(a) Variables

Name	Description	Initial Value
unique_id	Unique identifier for agent	unique_id
model	RecommenderSystem Mesa model	RecSysModel
book_id	Book ID from original dataset	book_id
n_reviews	Number of reviews given to book	is_reviewed
mean_rating	Average rating given to book	rating
n_read	Number of users that read the book	is_read
priority	Whether the book has a hidden priority or not	priority
vector	Vector from genres distribution	vector

(b) Methods

Name	Description
update	If item was consumed, update vector and increase n_read and n_reviews accordingly

iment. The seed was left as a constant per experiment, hence each run per experiment was initialized with the same configuration, making the execution of the simulation easy to reproduce.

Lastly, even though there was an option to modify the number of recommendations that should be displayed as a list, the exponential decrease of the probabilities of consumption yielded very low values beyond 10 items. The initial value of 50 was left as a constant for all experiments, as it had very little impact on computational performance and it would have needed to be lower than 5 to have a relevant effect on the outcome, which does not seem too realistic.

Table 3.4: Model initialization parameters

Parameter	Description	Default Value
n_users	Total number of users to extract from datasets	2
steps	Number of steps per simulation run	1
priority	Type of priority	None
dummy	Use of pre-loaded data	False
seed	Random state	None
thresholds	Book limit thresholds for reader personas	[5, 20, 50]
ignorant_proportion	Proportion of ignorant population	1.0
rec_engine	Type of RS	content-based
df	Pre-loaded model dataframe	Empty dataframe
df_items	Pre-loaded items dataframe	Empty dataframe
df_users	Pre-loaded users dataframe	Empty dataframe
initial_store_path	Path to directory to store new files or None for new directory	None
n_recs	Number of recommendations to display	50
social_influence	Whether recommendations can be prioritized based on social influence	False
run_type	“results” or “sensitivity” for sensitivity analysis	results
verbose	Print non-essential outputs to terminal	False

3.1.4.2.Model Logic

Once the model has been initialized and the agents have been created using the dataframes provided by the pre-processing steps, then the model is run by executing the logic described in the pseudo code algorithm 1. This fragment contains the specific steps executed; figure 3.3 shows the same logic as a flow diagram. The procedures in lines 5, 7, 8, 9, 11 and 13 are methods from the user agent model, as described by table 3.2b, whereas the procedure in line 17 is a method from the item agent model. Lastly, the procedures in lines 20 and 22 belong to generic model methods:

- collect_data: Mesa's datacollector helper class contains this method for gathering data in each iteration. The variables to be gathered need to be declared beforehand. The method is called after a complete step for all agents has been finished, thus it records data for every agent in every step, even when the agent did not have any interactions. The variables collected are:
  - agent_type: type of Mesa model agent
  - vector: agent vector
  - user_books_consumed: list of books consumed by user
  - item_n_read: number of times a book was read
  - item_n_reviews: number of reviews given to a book
  - item_mean_rating: average rating given to a book
- store_results: store the results of the simulation run as a CSV file using a Pandas dataframe, in the location selected by the initialization parameters or the directory created

Algorithm 1 General model logic

```
1:  for i ← 1 to steps do
2:      for j ← 1 to n_users do
3:          agent ← random user agent
4:          random_activation ← random value between 0 and 1
5:          read_probability ← get_read_probability(agent)
6:          if random_activation < read_probability then
7:              recommendations ← get_recommendations(agent)
8:              book ← pick_choice(agent, recommendations)
9:              similarity ← update(agent, book)
10:             random_review ← random value between 0 and 1
11:             review_probability ← get_review_probability(agent)
12:             if random_review < review_probability then
13:                 user_review ← similarity
14:             else
15:                 user_review ← None
16:             end if
17:             update(book, user_review)
18:         end if
19:     end for
20:     collect_data()
21: end for
22: store_results()
```

The complete Python implementation of the model and agents objects is detailed in appendix B.

3.1.4.3.Type of Recommendations

The original implementation of the recommendation engine included a purely content-based RS, where predictions of items would be calculated based on the similarity between all items and a specific user. However, since a simple set-up of 100 users yielded more than 40k items on average, for every activated user (i.e. a user that passes the threshold of consumption with which it was initialized) in every iteration there would be 40k cosine similarity calculations to get the list of recommendations. This was computationally demanding for a laptop. Hence, the type of recommendation was switched to a collaborative filtering one, where the activated user would be compared first with the rest of the users to get the most similar one and then compute the cosine similarity only between the items consumed by the most similar agent and the activated user's vector. For instance, if the user with the maximum number of items consumed was 50 in a sample of 100 users, then for each activated user there would be 150 cosine similarity calculations, which is drastically lower than the 40k that would be needed for a purely content-based recommendation engine.

Nevertheless, this decision was made before the reader personas were implemented. Given that the reading probabilities of getting a list of recommendations were decreased after the implementation, the algorithm was sufficiently sped up to try the content-based type of RS again. This is slightly preferable since it can include books that are not restricted to a single user's consumption history and it has been the most widely used type since the first applications of RS (Aggarwal, 2016). Additionally, I added a restriction on the number of books to be loaded, since there were users who interacted with many books without reading them. This caused a difference in magnitude of 6 times the pool of initial books than what would be expected based on a normal person's reading habits. Lastly, to make the computations more efficient, I mimicked the way a cache works by calculating all book similarities initially and only updating them when a user consumed a book, thereby changing their own vector of user preferences. If a user did not get a recommendation list, no similarities would be calculated.

3.1.4.4.Model Outcome

Since the purpose of the model is to understand the change in user preferences of a user, the way to obtain such change is by comparing the initial state of the user's attributes to their end state. The preferences of a user are modeled as a vector representing their consumption of items by book genre. Therefore, the outcome of the model should be a measure of the vector change in the period of the simulation, specifically the change in direction, as that would represent a change in book genres consumed. One metric that has been used in similar studies for measuring such change is cosine similarity (Kibe, 2023). The general formula for calculating the cosine similarity between two vectors has been described as equation 3.1. Since there were 20 runs per experiment, the cosine similarities were aggregated as averages per user. Therefore, the final value of the difference for user agent A was obtained with equation 3.2.

$$VectorDiff(A) = \frac{1}{20}\sum_{i=1}^{20} S_C(A_{i_1}, A_{i_{500}}) \tag{3.2}$$

Where S_C is equation 3.1, A_i represents the set of vectors of A in the i^{th} simulation run and the sub-script of i represents the specific step within that run.

Moreover, in order to compare the results from the different experiments more directly, I calculated the average change in cosine similarity (ACVD) using equation 3.3. ACVD measures the change in user preferences in each experiment as a single percentage value.

$$ACVD = (1 - \frac{1}{150}\sum_{i=1}^{150} VectorDiff(A_i)) \cdot 100 \tag{3.3}$$

The input for these formulas is a 16-dimensional vector, with each book genre representing a dimension. An example of such vector would be the following:

$$[965, 336, 0, 14576, 0, 0, 14576, 336, 88, 1253, 1253, 1253, 0, 0, 14576, 0, 12041]$$

Where the indexes with the maximum value of 14576 are 3, 6 and 14, which correspond to mystery, thriller and crime respectively. The second highest value of 12041 is at index 16, which is fiction. Lastly, the third highest value of 1253 was shared between indexes 9, 10 and 11, which are history, biography and historical fiction. If the cosine similarity is computed between this vector and another one, the result would show how different is the angle between both, in other words how similar are the values of each genre relative amongst them. If a user with this vector ended up consuming many mystery books, then the value in the third index would increase substantially, changing the overall direction of the vector. This would mean that the user is changing their user preferences as mystery was not part of their initial preferences.

3.1.5. Verification and Validation

When creating an ABM, it is imperative to make sure that the model represents what the designer intended and that the output is relevant for the experiment tested. These requirements are defined as model verification and model validation. The following subsections contain a brief description of each and an explanation for how they are being accounted for in this model.

3.1.5.1. Model Verification

Model verification is aimed at checking that the conceptual model was translated correctly into a computational model. Since computers will perform whatever they are told, we need to make sure that they are being told the correct information. This is not a trivial task, specially when dealing with socio-technical systems where agents might be acting based on non-deterministic rules (Dam, Nikolic, and Lukszo, 2013). However, there are some guidelines for reducing the chances of erroneous design choices. In particular, Dam et al. (Dam, Nikolic, and Lukszo, 2013) propose four aspects to take into consideration. Table 3.5 contains the way in which each will be addressed.

For this specific model, the verification was done by selecting one random user from the initial database and running a simulation of 50 steps with the default parameters: no strategy, ignorance towards the hidden priority, no social influence and content-based. Given that a casual reader user persona would have a probability of 0.007 of reading a book, it could be the case that the chosen user would not have consumed any book in those 50 steps. Therefore, the user chosen for the simulation belonged to the selective reader user persona. In order to guarantee that the user would consume at least one book, the dataframe of items was initialized previously. Otherwise, the user would not have picked a recommendation because all items were fetched from their own list of books consumed. The results of the verification run were recorded per step and stored as a CSV file after the simulation ended.

The ID of the user chosen was 711908. They were a selective reader with 15 books in their consumption history, mostly fiction and fantasy. When the simulation finished, they had consumed one fantasy book at step 34. Since the number of books they had previously read was not very large, but the consumed book had a very similar vector of attributes to the user's vector, their preferences did not change much in the end. Nevertheless, it was clear that the simulation ran smoothly as the position of fantasy and fiction in their vector increased by 1.

Since there was no social influence and no other users, the interactions of the user were limited to the recommendation engine, which yielded a list of recommendations just once. The default strategy of no hidden influence provoked a choice of item that was very similar to the user's preferences. Given that they were ignorant to the hidden strategy, it did not make them "suspicious" for scrolling down the list of recommendations for something more according to their taste.

Table 3.5: Model verification steps and the action(s) taken to address them

Step name (Dam, Nikolic, and Lukszo, 2013)	Implementation
Recording and tracking agent behaviour	For each step, all agent behaviour is stored using Mesa's Data Collector object. After the simulation run is completed, the data is extracted as a Pandas dataframe and stored in memory as a CSV file using the Results helper object.
Single-agent testing	The model allows for selecting a minimum of one user but with items from the rest. Even though the type of RS is content-based, the items loaded cannot belong entirely to the user's consumption history.
Limited interaction testing	Type of priority can be adjusted to not prioritize any item, so that interactions between users and items are carried out purely from the similarity between both. Additionally, the number of steps can be set to one to minimize number of interactions.
Multi-agent testing	By comparing the previous single-agent and limited testing with the benchmark model, we can assess the emergent behaviour of multiple agents interacting. Furthermore, some variability can be introduced by keeping the benchmark model's parameters static except for the number of agents.

3.1.5.2. Model Validation

Model validation is aimed at checking that the computational model represents a real-world system. When designing a model, there is one or more hypothesis to be tested. If a model is built correctly, its output should help to validate the hypothesis. Nevertheless, depending on the specific case, it can be very difficult to gather data from a real system (Dam, Nikolic, and Lukszo, 2013), either because the simulation takes place in future states or because the real system does not have this data available (as is the case with the Goodreads dataset; there is no temporal variable in the dataset that could help us to infer future agent behaviour). In such cases, the validation process focuses on the usefulness of the model. As with verification, Dam, Nikolic, and Lukszo (2013) provide four steps to validate the output, which are described in table 3.6. While the verification steps should all be required, the validation steps are not. Hence, not every step is implemented for this model.

As mentioned in section 2.1, there have not been many studies that model manipulation with an ABM, regardless of the specific system. This lack of evidence poses an obstacle for properly validating both the model and the results. Furthermore, in terms of effect, there has not been any study that measures the effect of manipulation on user preferences using an ABM. Nevertheless, there are three studies that can serve as a baseline for determining the validity of the framework chosen, even if the actual implementation diverges in comparison to this model: the work of J. Zhang et al. (2020) for the creation of the RS through an ABM, the work of Kibe (2023) for the simulation of a book RS using cosine similarity instead of a machine learning model, and the work of Zhu et al. (2024) for the modeling of user preferences and its changes derived from a RS.

From the perspective of user-item interactions, J. Zhang et al. (2020) build a design framework where the RS is divided into three components. These components have already been discussed extensively, but it is worth noting that the mechanisms in which they operate between them are very similar to the setup proposed in this study. Their model uses data from Netflix and Yahoo! Music. Both of these datasets contain items that can be described with the same variables as the ones used for books; mainly, a vector representation of the item's attributes. Their work also provides a way for randomness to be incorporated to the model, thus mimicking the decision-making process of a user when interacting with the platform. In terms of results, they measure the

accuracy of their RS, as well as the diversity and relevance of the items recommended. Since there is no mention of manipulation, both the consumption strategies and the outcomes of the model have little similarities with the purpose of this thesis.

From the perspective of generating recommendations, modern RS employ a combination of multiple machine learning and deep learning models that are highly complex and which interact between them for different tasks (Çano and Morisio, 2017). Building a model such as these is hard and computationally demanding. However, some authors have explored the use of cosine similarity for calculating the items that best fit a user, based on a vector representation of the attributes of both. In particular, the work of (Kibe, 2023) provides a manual for building a book RS using such metric instead of a deep learning model. They also use a dataset from GoodReads (though not the same one) to test the accuracy of their model. The calculation of the recommendations is very similar to the one used here. Yet, as with the previous study, they do not mention manipulation nor they build an ABM. Their study is mostly a showcase of how to simulate a book RS.

Lastly, and perhaps most importantly, the work of Zhu et al. (2024) provides a very similar experiment to the one of this model. They do attempt to measure the effect of manipulation on user preferences using some form of simulation, although it is not clear that they built an ABM, as the type of simulation is never explicitly referenced. Moreover, their definition of manipulation is discussed very briefly and diverges slightly from the covert influence account, as they assume that a RS has successfully manipulated a user's preferences if a) the user's preferences change from their initial state to their final state after interacting with the system, and b) the change results in an increase in the platform's revenue. The second clause of this definition is not relevant for this thesis, however the first one is one of the requirements I proposed in the operationalization of manipulation (see section 2.2.1). This is an important validation for the way in which the present model attempts to simulate the effect of a manipulative RS as a change in user preferences. Furthermore, they segment their work in four stages:

1. Initial preference calculation: this step refers to the pre-processing steps described in section 3.1.2.1 for calculating each agent's vector
2. Training data collection: since they make use of machine learning models, this step refers to the collection of the input data for its training. This could be similar to the extraction of user-item interactions from the public dataset, but there is a clear difference as to how the data is used
3. Algorithm training and interaction: this refers to the training of the models, which is not relevant for this thesis
4. Metrics calculation: in order to compare their experiments, they define a set of metrics with which to measure the results. Some of them are very similar to the ones I proposed

It seems like stage 1 and 4 are the most relevant for my thesis, as they touch upon processes that are very similar between both models. Regarding the fourth stage, two of the metrics that they define are click-through rate (CTR) and preference shift (PS). The former is basically an extension of the equation 3.1. The latter is calculated as 1 minus the similarity between the set of recommendations favored by the user in the initial step and the final step, which is a very similar measurement as the complement of equation 3.2. Since they are focused on the individual relevance of each item in the list of recommendations, the variables are somewhat different. However, the general purpose of both metrics is the same as the one I aim for: the relevance of a recommended item for a user to choose and the change in user preferences between the start and the end of the simulation. In section 4 I proceed to compare the actual results of their work and the ones obtained from my model.

With these three considerations, it can be argued that the chosen implementation of the model is a valid one, despite its differences and the lack of a study that contains all elements: manipulation, book RS, and ABM.

Table 3.6: Model validation steps and the action(s) taken to address them

Step name (Dam, Nikolic, and Lukszo, 2013)	Implementation (if existent)
Historical replay	Unfortunately there is no historic data available for this dataset. Therefore, the only way to try to reproduce the emergent behaviour is by choosing different samples of users for each experiment. The randomization can be controlled by setting the initial seed
Expert validation	The RS aspect of the model can be validated with the advisor for this thesis. However, there is no clear research done on modelling manipulation of RS, so experts in this type of setup would be scarce.
Literature validation	Most of the validation is done through the literature review. As mentioned before, the experimental setup was based on the work of J. Zhang et al. (2020). The works of Kibe (2023) and Zhu et al. (2024) are used as benchmark too.
Model replication	For the scope of this thesis, there is no second model that can be used for comparison. Further research could explore this option as a way to improve results (for instance, by choosing different recommendation strategies).

3.2. Sensitivity Analysis

Sensitivity analysis is a method that has proven to be very useful in ABM for exploring the impact of the model's parameters on its output (Thiele, Kurth, and Grimm, 2014). This information is helpful for understanding the behaviour of the model to better draft the hypothesis or experiments to test. Broadly speaking, it consists on varying the parameters of the model and running simulations under each variation while measuring the outcomes. There is not a single method to perform this, but the most popular is OFAT (one-factor-at-a-time) (Ten Broeke, Van Voorn, and Ligtenberg, 2016). This consists on selecting a fixed map of parameter settings while modifying the value of a variable parameter, then repeating this for the rest of the parameters. The most important contribution of this method is that it gives a clear description of how a single parameter affects the output of the model and the magnitude of such influence.

For the current model, the main parameters that could be modified upon starting the model were priority, ignorant proportion and social influence. The scenarios for each sensitivity experiment resulted from obtaining the possible combinations of the values shown in table 3.7.

The possible values of the priority variable were the domain of rational numbers between 0 and 1, and the list of book genres available. When the priority was equal to 0, the model would perform as if there was no hidden strategy selected. This corresponds to the model returning recommendations purely based on the cosine similarities between user and item vectors. Likewise, if the priority was 1, then all books would be prioritized, which would be equivalent to a completely random generation of recommendations. For the strategies involving a random sample of prioritized books, the values chosen were the decimals between 0 and 1. This selection is granular enough for assessing the differences between small variations but large enough to avoid running too many simulations. As for the strategies involving book genres, since it is a categorical set of values, all genres were chosen. From the model implementation perspective, the chosen strategy determined the place of the prioritized items in the recommendation list returned to the user, overwriting the position that would have been reserved for the most similar items.

Similarly, the possible values of the ignorant proportion variable were the domain of rational numbers between 0 and 1. However, the purpose of varying this parameter is to determine if having a sample of users being aware of the hidden strategy would have an effect on the change in user preferences, not how many users would be

Parameter	Type	Value(s)
Priority	Variable	0.0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0, fantasy, fiction, non_fiction, mystery, young_adult, graphic, thriller, paranormal, romance, history, biography, historical_fiction, comics, poetry, crime, children, fiction
Ignorant proportion	Variable	0.5, 1.0
Social influence	Variable	True, False
Steps	Static	360
N users	Static	150
Thresholds	Static	[5, 20, 50]
Engine	Static	content-based
N recs	Static	50

Table 3.7: Sensitivity analysis parameters

needed for causing such effect. Therefore, the selection of possible values was reduced to either 0.5 or 1: either half of the population was ignorant of the hidden strategy or all the population. By running a simulation for each priority value in the sensitivity analysis, it would be clear to determine if ignorance contributed to the effect. Even though the results of the analysis could have shown that further granularity would be needed, the numbers showed a clear pattern and the selected values were chosen for the experiments. Furthermore, deeper granularity was not needed given that the purpose of the model was to understand how a specific strategy with a heterogenous population would change the user preferences. For each new value simulated, every strategy would require two new simulation runs, greatly increasing the amount of computational effort required.

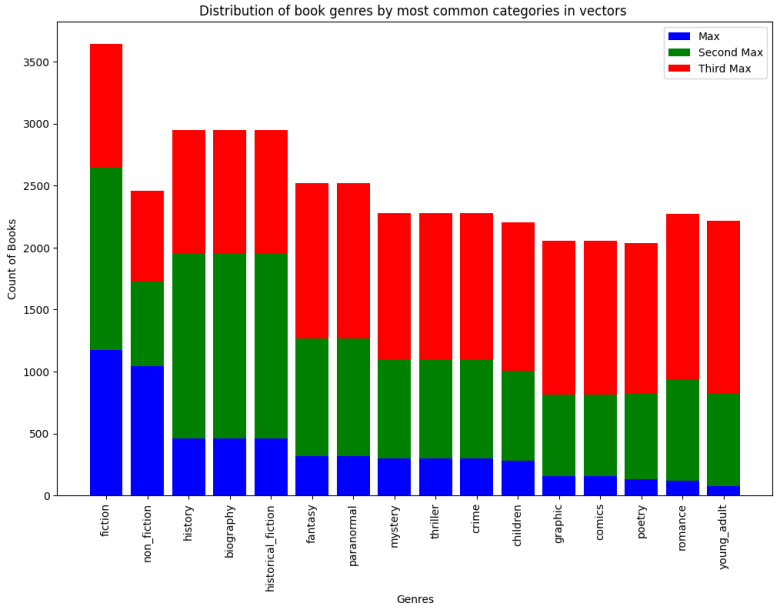
Lastly, since the social influence parameter was a boolean variable, it could only take two possible values. Hence, all simulations of the analysis were run with and without social influence.

The rest of the parameters were left constant. The number of steps was chosen to be 360 in order to simulate an entire year. Since the casual reader user persona would have consumed less than 5 books per run on average, running the simulations with less than 360 steps could have shown many outliers for this group, which would in turn give a misleading trend. On the other hand, more steps would certainly improve the conclusions of the analysis, but at a higher computational cost. The number of users was also left constant based on a reduced version of other studies. For instance, the works of Yin et al. (2019) and Ross et al. (2019), contained 500 and 1000 actors respectively. Furthermore, they did not vary this values, but the proportion of those agents belonging to different categories. This was expected since that was their research goal. As my model was limited by the computational capacity of my computer, I chose a smaller number of users that guaranteed 1) a multiple of 3 (I wanted to have the same number of agents per user persona), and b) a sample large enough to produce a statistically valid result. For example, the wolf-sheep predation model and the Boltzmann wealth model both have a population of 100 to 150 agents[3]. The thresholds were left constant based on the actual literature about the number of books that people read per year. The recommendation engine was chosen to be only the content-based one since it is the most popular type of RS, and the way it generates recommendations is optimal for testing a hidden priority strategy for a selection of items. And the number of recommendations on the list was left constant because of the reasons exposed in section 3.1.5.2: the exponential decay in the probability values produced a negligent presence for those items that were low on the list and most of the interactions happened between the first 10 books.

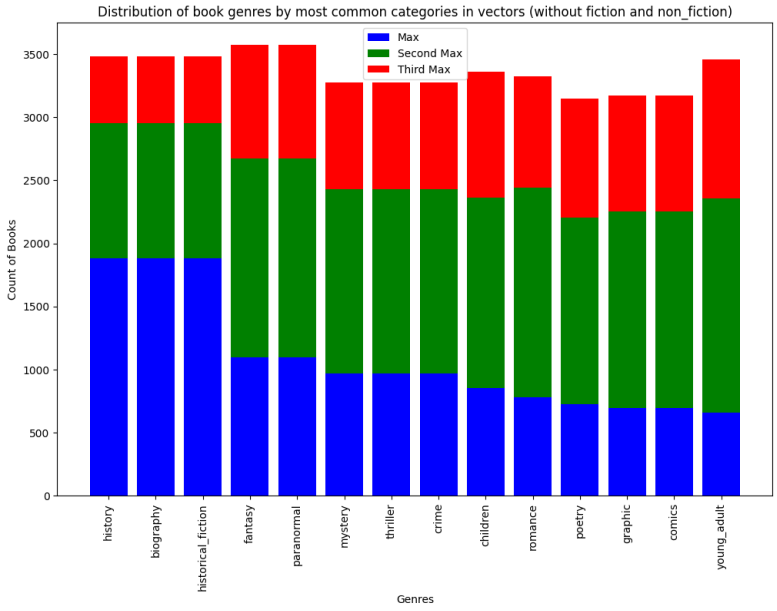
With the number of users being 150 and the random state seed set to 256788, the total number of interactions

[3] Mesa examples on Github

that were pulled from the general dataframe was 5,757. Out of these interactions, there were 3,916 unique books. From table 3.7, only the "variable" type parameters were measured during the sensitivity analysis, resulting in a total of $27 * 2 * 2 = 108$ experiments ran. The reason why these were the only variables used for the analysis was based on the assumption that the rest are of no particular interest for the hypothesis of this thesis.



(a) All genres



(b) Filtered

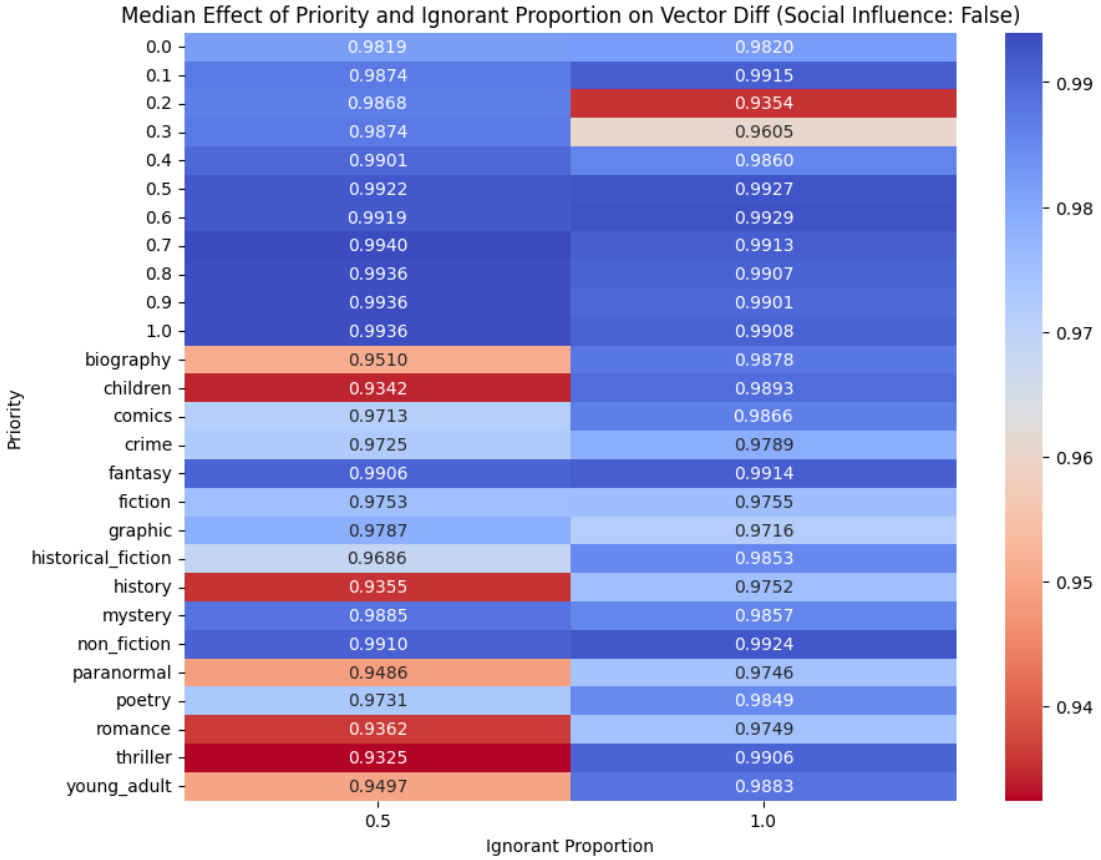
Figure 3.4: Histogram of book genres for sensitivity analysis items dataframe

Before running the simulations, the distribution of each genre was analysed. Figure 3.4a contains the results. The color represents the number of books that had a specific genre as the main genre of the book or as the second and third main ones; in other words, the sorted count of votes. If the maximum amount of votes was equal for two genres, then both would get that book as representative. The second maximum and third maximum labels follow the same logic. For example, book with ID 965 had the same vector displayed above in section 3.1.4.4:

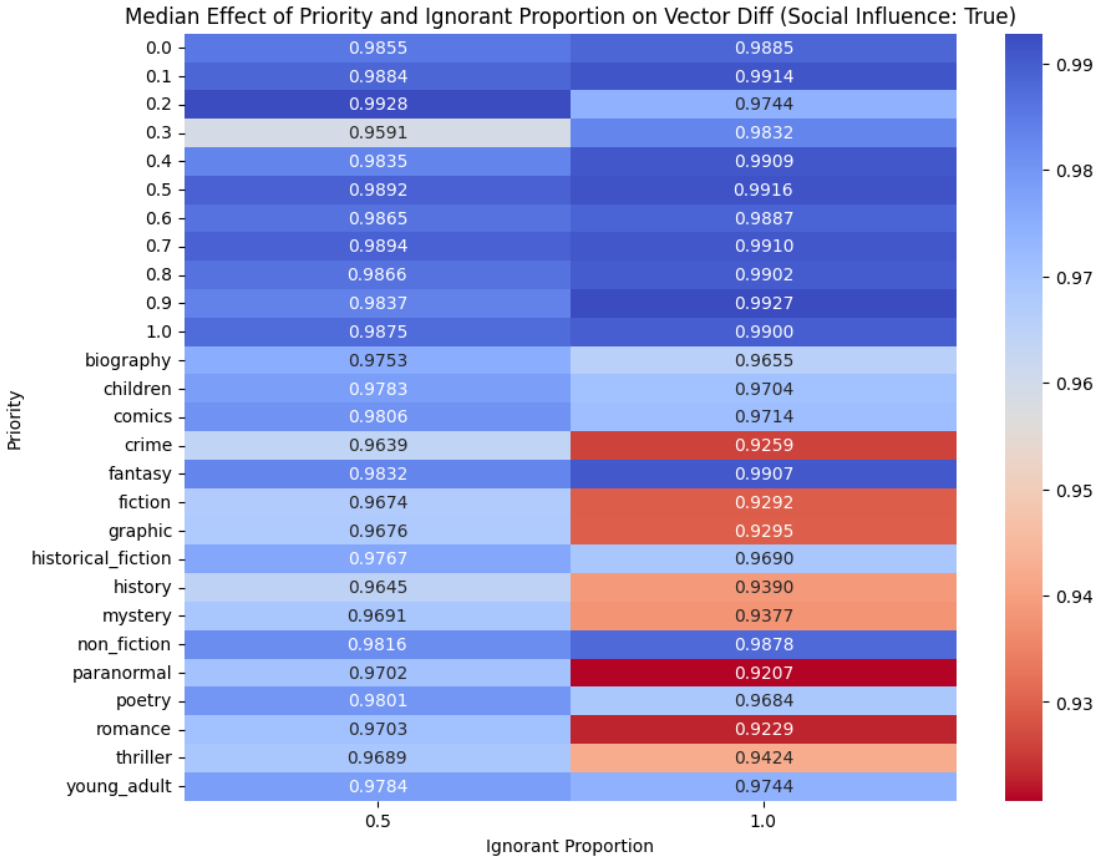
$$[965, 336, 0, 14576, 0, 0, 14576, 336, 88, 1253, 1253, 1253, 0, 0, 14576, 0, 12041]$$

These distributions are important because they give a sense of how diverse is the set of books genres in the dataset.

The distribution showed that fiction and non-fiction contained most of the books. This seems intuitive given that they are general and almost mutually exclusive categories on a different hierarchy than the rest: almost any book can be labeled as fiction or non-fiction. For this reason, a second histogram was extracted without those two genres. The chart is shown in figure 3.4b. The distribution looks more uniform by total counts, yet the maximum counts are now biased towards history, biography and historical fiction. After computing the standard deviation for the maximum value in both cases, it became apparent that there was actually an increase from 310.58 to 455.16 when filtering. This suggests that there is a less uniform distribution for the top categories if fiction and non-fiction are removed. Therefore, it was decided to keep them both for the actual simulation runs.



(a) Without social influence



(b) With social influence

Figure 3.5: Sensitivity analysis results

The results are shown in figure 3.5, where the median difference in the cosine similarity of the first and last vectors of each user per experiment is plotted. The x-axis contains the proportion of ignorant population and the y-axis contains the prioritization strategy. The first sub-plot does not have social influence activated while the second one does. Finally, the color code is a visual tool to display the magnitude of the differences. The same charts were extracted with the average value of the differences, but the distributions are highly skewed for the manipulative models, given that a few users had a considerable change in their user preferences while some had no change whatsoever. Therefore, using the median seems to bear a better representation of the actual changes.

An interesting conclusion from the charts in figure 3.5 is that the vector difference seems to worsen (i.e. decrease) when half of the population is ignorant of the priorities of the algorithms and there is no social influence between users. In other words, as more users become aware that some books are being prioritised, the less they are influenced by the hidden intentions of the RS. This is expected, since users that know that they are being shown results for specific reasons are less inclined to consume the first elements (Gedikli, Jannach, and Ge, 2014). However, when there is social influence, the opposite seems to be the case: the more users know about the intentions of the algorithm, the more they seem to be influenced. This could be explained by the fact that users can put more weight on the reviews of books consumed by people they follow over their own preferences. This has been shown to be the case in some instances of RS (Ross et al., 2019).

When looking at the details of how each prioritisation strategy affects the outcome of the model, it seems like there is no clear pattern. For instance, when the strategy was based on a percentage of the books being prioritised, the proportion of ignorant population set to 0.5 and the social influence was turned off, there is a proportional correlation between the cosine similarity and the percentage of books being prioritized. Yet, as soon as the social influence is active, the correlation disappears. One explanation for this behavior could be related to the distribution of book genres and the random sampling of items for the strategy. If there are more books being prioritized, then the top books from the recommendation list would all appear based on the priority and not the similarity with the user's preferences. It would seem counter-intuitive then that the effect on the vector differences was milder for a larger number of books prioritized, if the distribution of books from figure 3.4a was not taken into consideration. Fiction and non-fiction are quite predominant genres because they belong to a higher hierarchy of categories. Since all users have their preferences aligned with this tendency, it seems plausible that a larger proportion of those books would yield an option that a random user would deem suitable. Even though the trend is not as clear with the other parameter combinations, it is true that the lowest random prioritization strategies had more impact on the median change in user preferences.

The existence and non-existence of these tendencies was the main driver for not choosing the random prioritization strategy for the experiments. The strategies that prioritized a specific genre seemed to have a larger effect on the vector differences, since the books from the group that is being prioritized are closely related amongst them, for the cases that do not include fiction and non-fiction as the main genre. For instance, thriller, romance and paranormal had some of the lowest differences and they were also less frequent books. One factor that had an impact on these results was the set relationship between genres. Some genres tended to be grouped constantly (e.g. mystery, crime and thriller, or children and graphic), while others tended to appear by themselves (e.g. poetry). Therefore, when one genre was being prioritized, other genres could be as well indirectly.

3.3. Experimental Set-up

The following section describes the three experiments that were conducted, as well as the benchmark model for comparison. By experiment I am referring to the generation of an hypothesis, the choice of initialization parameters for the simulation and the results for each group of simulation runs. All experiments used the same random seed, hence the subset of users that were picked from the complete database was the same. This allowed for a reproducible and standard point of comparison. Moreover, each experiment consisted in 20 runs. The metric for measuring the results of each experiment is the cosine similarity between a user's initial and final vector of preferences. The results are aggregated values of all runs per experiment.

Given that the extraction of the users was constant, the independent variables used as parameters for the experiments were the following:

1. Priority: the priority strategy
2. Ignorant proportion: the proportion of users that were naïve with respect to the manipulation
3. Social influence: the existence of influence made from other users

These three parameters were chosen because they have different effects on the existence (or not) of a manipulative action. As it has already been discussed in section 2.1, the prioritization strategy represents the intention of the algorithm. Intention is one of the core components of manipulation, even when it is not a necessary condition (Jongepier and Klenk, 2022). Additional to those parameters, the number of users and the number of steps were left constant for all experiments but with different values than the default ones (see table 3.4 for the references). The former was set to 150 users (same as with the sensitivity analysis) and the latter was set to 500 steps. The reason for the increase from the 360 steps of the sensitivity analysis was due to the exploration of what a longer time span could provoke. One year seemed enough when the probabilities of getting a recommendation list were small. However, specially for casual readers, it could be hard to get an actual grasp of manipulation when there are so few items consumed.

The rest of the parameters were left with their default values as shown in table 3.4.

Table 3.8: Parameter initialization per experiment

Experiment	Parameter	Value
Benchmark	Priority	None
	Ignorant proportion	1.0
	Social influence	False
Covert	Priority	thriller
	Ignorant proportion	1.0
	Social influence	False
Overt	Priority	thriller
	Ignorant proportion	0.5
	Social influence	False
Overt with social influence	Priority	thriller
	Ignorant proportion	0.5
	Social influence	True

The proportion of ignorant population can be seen as a countermeasure against manipulation (see section 2.2.4), since users who are aware of the prioritization strategy behind an algorithm could choose options from the recommendations that are not the top ones or the ones labelled as such. Some researchers have already pointed out that manipulation can entail a disregard from the manipulator towards the methods in which a target becomes aware of the manipulation, not necessarily only the action (Klenk, 2022). Even though based on this approach a user could still be counted as being manipulated when they become conscious of the intentions of the algorithm, there could be a quantifiable consequence of their change in user preferences, which is the main aim of the research question.

Lastly, the social influence could be seen as either enhancing or detrimental to the effect of manipulation on the population (see section 2.2.3). It is certain that, as a form of influence, it can have some impact on how user preferences are modified. For instance, a user that follows another user with a very different set of user preferences would be more influenced than one who follows a very similar user. Sometimes, this influence can be present without an explicit link between follower and leader (Ross et al., 2019). For example, if I get a book recommendation from the platform and I read an anonymous bad review, I could be influenced to avoid consuming such

item without following the user who wrote the review. Nevertheless, for simplicity, this type of interaction will not be modelled. Therefore, social influence will be related to explicit links between users.

Table 3.8 contains a summary of the initialization parameters for each experiment, which will be described in detail afterwards.

3.3.1. Benchmark

As mentioned in section 2.1, when a RS does not have any hidden priority, then the results of the recommendations are calculated explicitly from the similarity between a user and an item's vector, purely from a content-based approach. The proportion of ignorant population would not have much impact on the results, given that even if they were aware of the strategy, it would be an overt strategy for them. Hence, the proportion of users that are ignorant is 100% for this model. Finally, there would not be any social influence, since users are basing their choice of options solely on the recommendation list from the algorithm. If we think about the model with this configuration, then it seems that this is a good initialization for what a non-manipulative RS would yield. From a user preferences perspective, given that this model is the least manipulative one, it should be expected to return the lowest changes in user preferences for the users. Therefore, when comparing with the rest of the experiments, all other models should generate a more significant deviation from the initial preferences of each user after running them.

3.3.2. Covert

The first experiment is running a purely covert model. Under the assumptions that the hidden influence is exerted through an explicit item prioritization mechanism without the users being aware of it, the way to implement covertness would be through a specific prioritization strategy. Even though there are multiple options to choose from, restricting by book genre and picking "thriller" in particular seemed appropriate after I ran the sensitivity analysis. The results of the analysis can be seen in section 4, but one of the findings was that "thriller" had a considerable impact on user preferences. Furthermore, it is not a high level category and practically all books labelled as such have the genre as the main category with the maximum count of votes (see figure 3.4). In terms of the number of books, it is right in the middle; neither too many nor too little. If the influence is to remain hidden, then all users should be ignorant to the strategy, thus the proportion of ignorant population is 100% again. Since this experiment is modelling only the influence exerted by the algorithm, there should be no social influence applied from other users.

3.3.3. Overt

We have now one non-manipulative and one manipulative model for the experiments. The following ones aim to determine how can the manipulation be affected when some users become aware of it. I refer to this model as the overt model, given that some fraction of the population will be notified about the hidden prioritization strategy (akin to how advertisements are labelled), rendering the influence as an overt one for them. As with the sensitivity analysis, the chosen proportion of the population to be ignorant was initialized to 50%. The reason why I chose this number is because it is large enough to give a sense of the impact that it would have while maintaining a reasonable number of users in the ignorance. In reality, while it is true that users in general are aware of the goals of a RS (Ghori et al., 2021), it would seem highly improbable that all users are conscious of the manipulative strategy from the algorithm. From an operational perspective, if a user becomes aware of the strategy, the hidden strategy of the prioritized items is removed when calculating the list of recommendations for that user, thus their options are generated based on the cosine similarity exclusively. This is equivalent to a user scrolling down the advertised content on a list until they reach the non-paid results.

3.3.4. Overt With Social Influence

Finally, the last experiment consists on a replica of the overt model with the addition of social influence. The purpose of this is to determine if having some interactions amongst users has a consequence for the change

in user preferences. As mentioned before, it is not clear whether such consequence would be detrimental or enhancing, but it is quite clear that it would have some effect. In practice, this experiment is the one that most resembles reality, given that RS nowadays have a mix of all its parameters: there are some hidden priorities from the algorithms, some of the users are aware of such strategies, and their is a direct influence from related users because of the network characteristics of these platforms. Yet, this is not necessarily guaranteed everywhere. For example, Spotify provides connections with users such that anyone can view what a connected user is listening to, thereby supporting the social influence proposition. Netflix, however, offers no connections, thus the influence within the platform is exerted directly from the RS.

4

Results

The following section contains the results of the model. In general, for the results that discuss vector differences, the magnitude of the variation in users' preferences is calculated as the cosine similarity between the first and last vector of the simulation steps. When the values of the cosine similarity for a specific user are equal to 1, then it can be argued that the user's preferences did not change, as their first and last state are completely similar. On the other hand, if the cosine similarity is 0, then the direction of their vectors is opposite, which means that their preferences diverged completely. For the general results, the values are calculated as 1 minus the average magnitude of change in the vector differences.

4.1. Experiments Results

Since all experiments shared the same model dataframe, they all shared the same number of items. With 150 users and a random state seed set to 123321, the total number of interactions from the general dataframe was 5,196, which yielded 3,387 books in total. The items dataframe between experiments only changed in terms of priority when there was a hidden priority enabled. The users dataframe did change for every experiments, since all priority, social influence and ignorant proportion had some effect on the generation of the dataframe. Hence, the main difference between all the simulation runs for each experiment (20 per experiment) was the randomness associated to a user's consumption in each iteration.

The general results of the model are shown in table 4.1, where the average change in vector difference is the result obtained with equation 3.3.

Table 4.1: General results

Experiment	Average change in vector difference	% change w.r.t. benchmark
Benchmark	2.58%	-
Covert	9.79%	279.5%
Overt	4.01%	55.4%
Overt with social influence	3.64%	41.1%

The findings indicate that the covert model significantly influences user preferences, almost quadrupling the likelihood of manipulation compared to the benchmark model. In contrast, the overt model and the over model with social influence have a much smaller impact on changing user preferences than the covert model, but they still show noticeable effects compared to the benchmark. The last experiment had the smallest impact on user

preferences. The range of these values, between 0 and 10%, matches the findings in Zhu et al. (2024), confirming that the model aligns with recent research.

As with the sensitivity analysis, before the execution of the simulations, I explored the distribution of book genres. Figure 4.1. The distribution is practically identical to the one from the sensitivity analysis dataset (figure 3.4a. The same pattern is shown, being the main difference the absolute values of the counts in the y-axis. Given the randomness associated to the sampling of users with a specific seed, the number of interactions in the sensitivity analysis data was larger than for the results data, which caused a smaller number of books to be extracted for the experiments. There was no need to extract a filtered histogram without the generic genres, so the simulations were carried out with this book dataset.

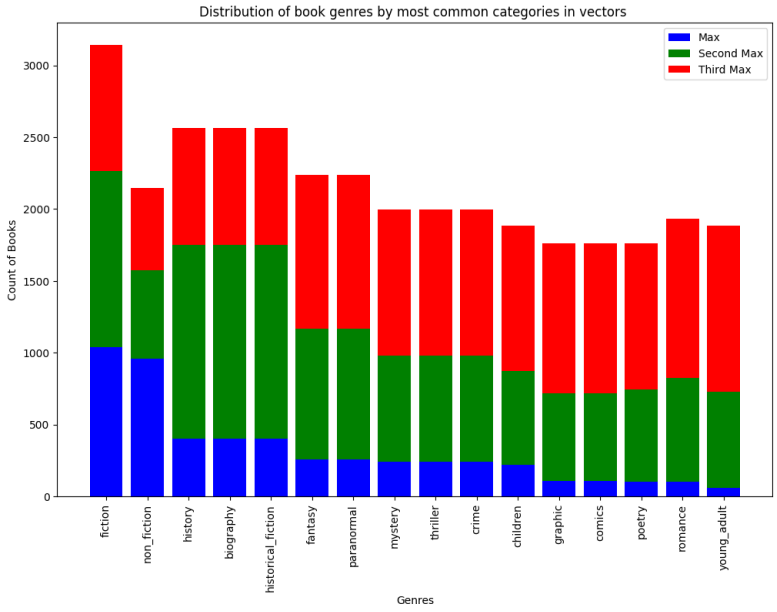


Figure 4.1: Histogram of book genres for results items dataframe

The results are shown as sub-plots in figure 4.2. The x-axis represents the cosine similarity between the first and the last vector of the user, which correspond to the first and last iteration of the simulation. Its range was set to [0.5, 1.0] to maintain the same wide ratio for easier comparison between them. The y-axis is the count of users that had such values. I added the median value as a vertical line with a legend to have a grasp of the main driver between each experiment.

By looking at both the median and the majority of bins, the experiment with the highest cosine similarity was the benchmark model, as expected from the initial hypothesis. Similarly, the one with the lowest cosine similarity was the covert model, which was also expected. In order to differentiate the distributions per reader persona, the charts were grouped by persona in the sub-plots of figure 4.3. The distributions exhibit similar patterns. For instance, the lowest values in the distributions belong to the casual readers cohort, while the avid readers cohort tends to be grouped closer to 1. Most of the values have a cosine similarity above 0.8 with the exception of the covert model, where the distribution is more spread out, reaching as low as 0.5 similarities for one user. Similarly, only the covert model showed a faster declining trend of avid reader counts towards the upper limit of the chart, whereas the selective and casual cohorts displayed the same skewed but spread out tendency. The results suggest that the prioritization strategy does play a role in modifying the user preferences, with the proportion of ignorant population having a counterbalancing effect on the change. Allowing for social influence yields the results that

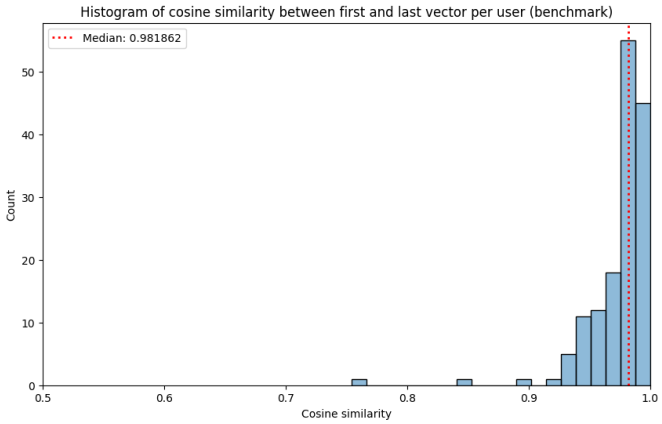
are closest to the ones from the benchmark.

Furthermore, the number of books consumed per user can be visualized in the four sub-plots of figure 4.4. As with some of the previous distributions, the four plots display a very similar distribution. It seems quite clear how the values are grouped per reader persona given the constant probabilities of requesting a recommendation list. The expected number of books is slightly larger than the one that was mentioned in chapter 3 because of the extra number of days calculated: 360 for the average values and 500 for the actual simulation runs. There were no users who on average consumed zero books. Even though some users might have consumed individually a number of books that falls between the gaps of the charts, when the values were aggregated their contribution to the mean was negligent.

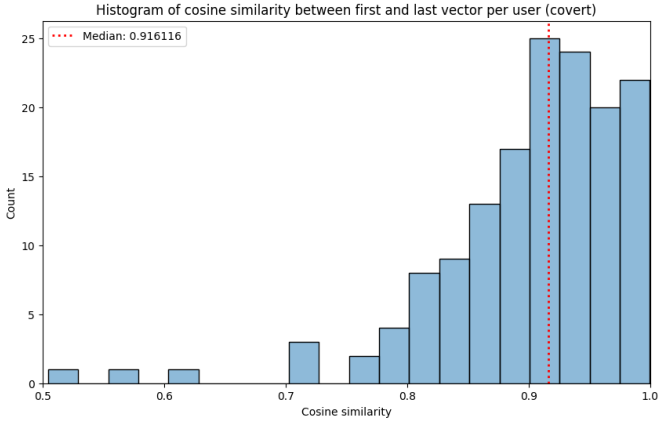
Lastly, there are no results with respect to the evolution of books, despite their attributes being measured and collected on each iteration. The only results concerning items are shown in table 4.2, where the top books are shown by the average count that they got per experiment, along with the main genre associated. The purpose of this table is to capture the prioritization strategy from the perspective of the items as well. The goal of the main research question does not concern the impact that manipulation could have on the items that enable it, it is only concerned with the effect on the user preferences of the users.

Table 4.2: Top items consumed

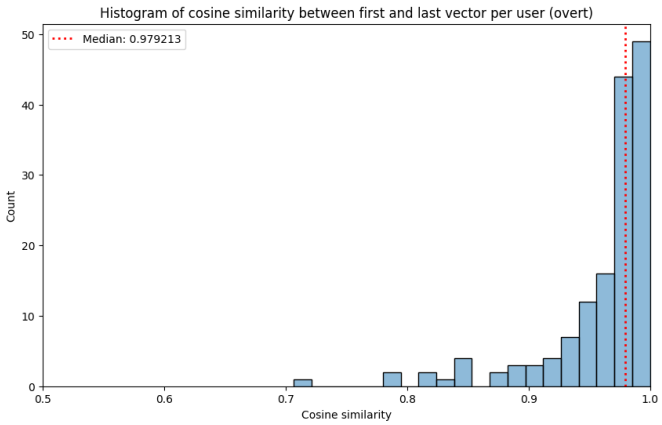
Experiment	Book ID	Mean Count of Read	Main Genre
Benchmark	123504	84.60	Fiction
	799266	83.05	Fiction
	3974	79.10	Fiction
Covert	205	141.45	Mystery, thriller, crime
	218	136.95	Mystery, thriller, crime
	228	130.35	Mystery, thriller, crime
Overt	123504	68.55	Fiction
	799266	68.40	Fiction
	3974	65.80	Fiction
Overt with social influence	14831	110.70	Non-fiction, history, biography
	205	105.60	Mystery, thriller, crime
	799266	102.55	Fiction



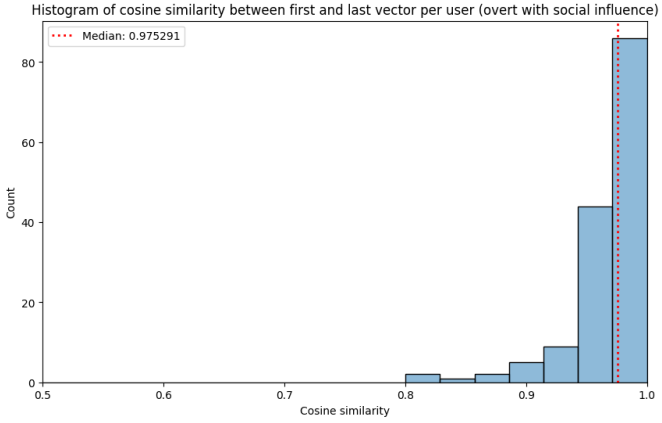
(a) Benchmark



(b) Covert

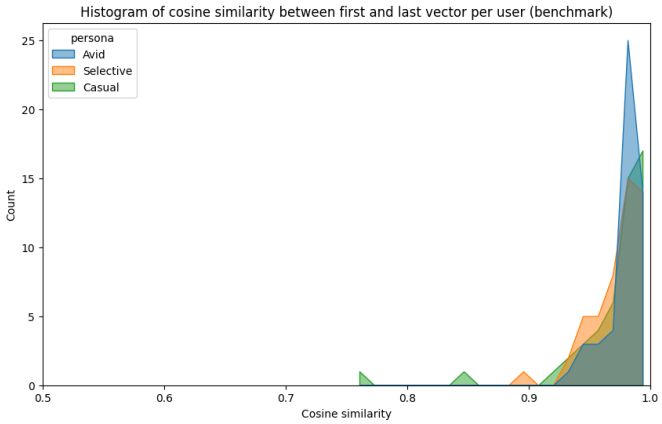


(c) Overt

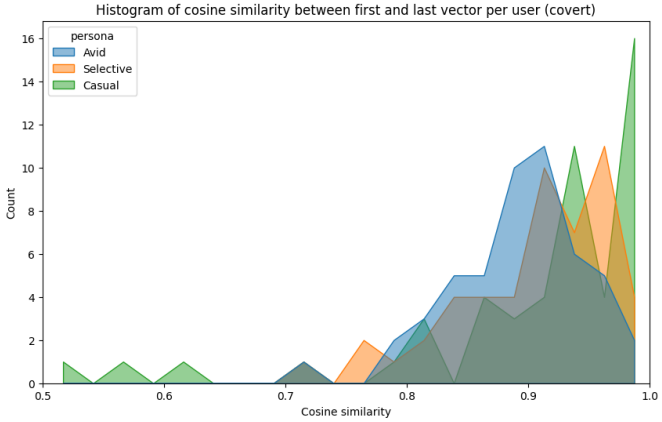


(d) Overt with Social Influence

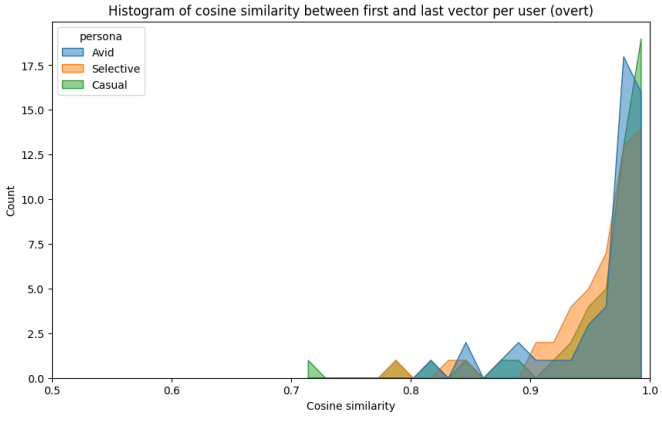
Figure 4.2: Experiments results



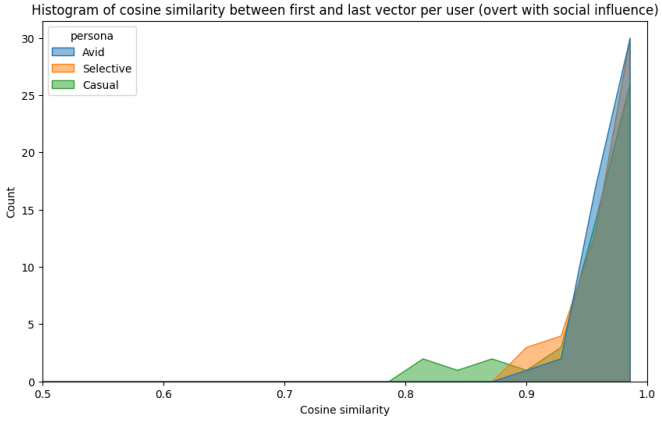
(a) Benchmark



(b) Covert

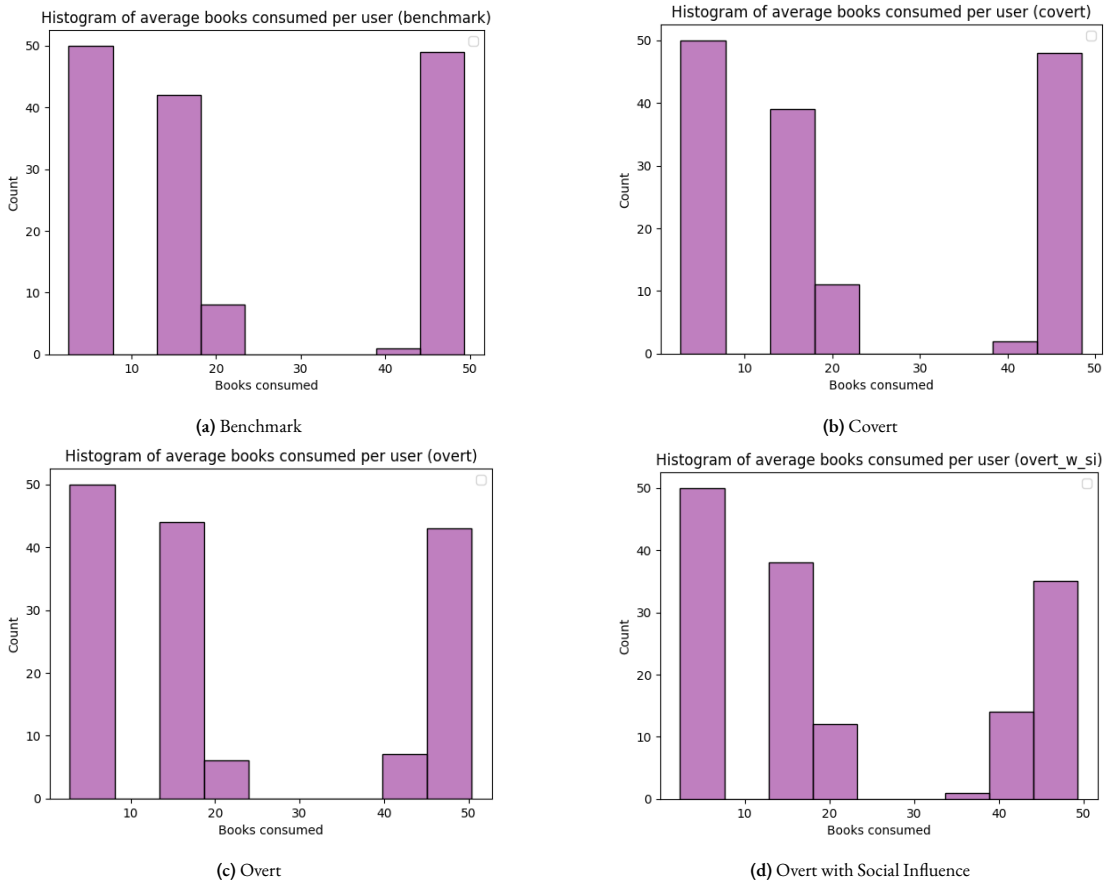


(c) Overt



(d) Overt with Social Influence

Figure 4.3: Experiments results by reader persona



(a) Benchmark

(b) Covert

(c) Overt

(d) Overt with Social Influence

Figure 4.4: Distribution of average books consumed per user

5

Discussion

The following section contains a discussion of results presented in chapter 4, segmented as an overview of the main results and a detailed discussion of the results, followed by the strengths and weaknesses of the research and the practical and theoretical implications. Lastly, observations about future research are given.

5.1. Discussion of Results

The goal of this thesis was to answer the research question: how significantly does covert influence alter user preferences, compared to overt influence and no influence? Furthermore, it investigates the extent to which covert influence affects user preferences in a book RS, compared to overt influence and no influence. The data in table 4.1 indicate that covert influence alters user preferences by 9.79%. This means users' choices deviate nearly 10% from their original preferences after 1.5 years of exposure to the manipulative RS. When users are aware of the covert tactics, preference change falls to 4.01%, significantly less than under covert conditions. Adding social influence among users further reduces the change to 3.64%. Without any covert influence, preferences shift by 2.58%. Using this as a baseline, covert influence has a 3.7 times greater effect, while overt influence and socially-influenced overt scenarios are 0.5 and 0.4 times the baseline change, respectively.

Regarding the results of the experiments, the charts from figure 4.2 support the change in user preferences hypothesis. When users were being influenced covertly, they exhibited a larger magnitude of change in their user preferences. There was even one user who had a 50% change in their preferences, suggesting a considerable shift. Moreover, the effect segmented by user persona did not differ much from the general results. All cohorts followed the same distributions. An observation can be made on the specific users that had the largest change in preferences. In all cases, those users were casual readers. This is expected since the algorithms of the RS improve their recommendations when there is more information available from the users (Burke, Felfernig, and Göker, 2011). When there is little information, the digital representation that they have from a user is less accurate, thus the recommendations have a higher degree of uncertainty, which can certainly cause a larger effect on their preferences than someone who is highly predictable. This issue is referred in the RS literature as the "cold-start problem" (Çano and Morisio, 2017), and the results reflect this phenomenon. Furthermore, as seen in the distributions of figure 4.4, casual readers read on average a very small number of books after the simulations ended. For selective and avid readers, more books read could help to balance out the impact of one wrong recommendation, whereas for casual readers there might not have been a subsequent book to compensate for it. For example, let's say I am a new user on Netflix. Netflix does not yet know what my preferences are and I only consume two movies from similar genres. If Netflix suggests a very different genre in an effort to explore my taste in movies (Bouneffouf, Bouzeghoub, and Gançarski, 2012) and I consume the item but I consume nothing else, then that

last recommendation would have had a great impact on the digital representation of my preferences, even if I do not review or show my actual opinion about it.

For the benchmark and the covert model experiments in particular, the results are unsurprising with respect to the total distribution of vector differences. However, when comparing the same prioritization strategy of the covert model with the overt and overt with social influence models experiments, there are interesting insights to extract. Firstly, having a percentage of the population being aware of the intention seemed to decrease the change in user preferences considerably. The median of the change for the covert model (sub-figure 4.2b) is 0.91 while the median for the overt one (sub-figure 4.2c) is 0.97, very close to the 0.98 of the benchmark model (sub-figure 4.2a). From a conceptual point of view, this is explained by the fact that a knowledgeable user may have the tools to make a decision that better fits their own user preferences (Burke, Felfernig, and Göker, 2011). From Susser, Roessler, and H. F. Nissenbaum (2018), there are three key characteristics of manipulative practices: 1) hidden, 2) exploitation of target's vulnerabilities, and 3) targeted. For the covert model, these characteristics hold, since 1) all users are ignorant of the algorithm's intention (i.e. the prioritization strategy), 2) the RS is taking advantage of the user's belief in a transparent recommendation to force unwanted items, and 3) the digital representation of the user is used for selecting the chosen items. It could be argued that the last one is not a strong argument since the recommendation engine does not discriminate between users, it pushes the prioritized items equally for any user. However, the authors follow by stating that, under strict scrutiny, the only necessary condition is the first one. Therefore, the overt model breaks this condition entirely by offering the users some information regarding the manipulative action. Under the covert influence account, they are not being manipulated anymore and the results reflect this as a consequence of the reduction in vector differences. Yet, the values are sill lower than for the benchmark model since the algorithm's intentions remain hidden for half of the population. Interestingly though, this does not result in a proportional reduction: decreasing the ignorant population by half does not yield a decrease in vector differences by half.

As described in section 2.2.3, social influence is a process of cause and effect where an individual's actions can exert power on another individual, so that the latter changes their opinion based on the former's activity. By looking at the overt with social influence experiment results (sub-figure 4.2d) it seems like the social influence poses little change with respect to the overt model. Having 0.97 as the median similarity, the overall results look very similar, but the specific distributions are slightly less spread out (sub-figure 4.3d. An important observation could be made about the ticks in the y-axis. The total count of users is much higher than the overt experiment. This is also displayed by the number of bins in sub-figure 4.2d, where the majority of the frequencies rested on the last bin close to 1. The social influence seems to decrease the impact of the prioritization strategy even more. Out of the four experiments, the ones that have the closest results are the benchmark model and the overt with social influence model. From a conceptual point of view, it was not clear beforehand what the effect of social influence would have on top of a manipulation strategy from the algorithm. These results are more inclined to the detrimental hypothesis, where having social influence is a countermeasure to manipulation, further reducing the effect on users' preferences. A possible explanation for this could be the initial similarity between users. Given that users follow other users who think alike them, they books that a followed user consumed would be quite similar to the user in question. If the user was ignorant of the intentions of the algorithm, then the recommendations from the followed user would have a higher probability of being consumed than other items. Although this type of recommendation is closer in meaning to a collaborative-filtering, the key difference is that the items have a higher probability not because they are deemed more likely to be consumed by the algorithm, but because the user sees them as more interesting, since another user just consumed it. Despite this fact, when both users are very similar, the line between content-based and collaborative-filtering becomes blurry; items could score the same predictability regardless of the RS type. This phenomenon has been observed in the literature as well (Mathew, Kuriakose, and Hegde, 2016), which opens the door to hybrid systems (Çano and Morisio, 2017). However, it has been shown that users who follow not the most similar users but slightly similar ones can be influenced towards a different direction (Cercel and Trausan-Matu, 2014). This scenario was not part of any experiments so it is left as part of further research.

Lastly, the results from table 4.2 support the claims made above, with the benchmark and the overt model having the same books as the most popular ones. Unsurprisingly, all three most consumed books for the covert model were part of the thriller genre. For the overt with social influence experiment, it was unexpected to find different books with such counts. All three top books belonged to completely different genres and the times they were read were much higher than for the overt experiment. This is also a reason why the distribution of the vector differences for the last experiment is more skewed to the right. However, on average, the number of books that were consumed was the same, as seen in sub-figure 4.4d. This is a clear example of the network effect, where a few items are consumed by many users because both the RS prioritizes them and because users follow other users who consume them, in a positive feedback loop that only makes them more popular than others. When this happens, usually less popular items get segregated and are consumed less. Moreover, since these items are usually the ones that belong to the minority categories, they do not get the opportunity to influence the user preferences of the largest group of users. For instance, research using Spotify has shown that users who highly depend on the recommendations of the algorithm for their consumption, are more inclined to reduce the diversity of the music they listen to (Anderson et al., 2020). Those that have a wider diversity are mostly searching music through their own means without relying on the RS.

5.2. Strengths and Weaknesses

The primary strength of this research lies in its innovative approach, using an ABM to simulate manipulation through the interaction of a RS and its users. It is an interdisciplinary research that bridges subjects that are distant in meaning, offering a novel method for capturing ethical implications from RS in a simulation study. It provides a quantifiable measure for the change in user preferences due to manipulation. The existing literature contains either research done on simulating RS with ABM, or simulating manipulation - defined broadly - with ABM. The model developed for this thesis fits the intersection of both types of models, providing a systematic way in which a manipulative RS can be simulated. Moreover, although this thesis addressed a book RS, the implementation is highly extensible to other types of RS. By focusing on the operationalization of manipulation and the modelling of user-item interactions, the model's input dataset could be changed for other fields of study and it would still provide a measurement of the change in user preferences as outcome. Additionally, the simulated RS mimicked a real intelligent system by calculating cosine similarities between users and items, yet a machine learning model could be easily added to the simulation by replacing the recommendation engine's predictions.

This research is limited as to the chosen definition of manipulation. Hence, the operationalization of the concept is restricted to the covert influence account defined in chapter 2. However, manipulation can be studied from other points of view and yield different results. Moreover, the way in which items are prioritized is based on an explicit intention introduced into the system by its designers. Even though this type of strategy can influence user preferences, there are mechanisms that emerge from digital platforms which are not explicitly coded into the system but which can result in a covert influence. For instance, when bots drive the popularity of some content in particular within a social media platform, it is not the designers of the RS the ones who decide which content to prioritize, but the methods intrinsic to the deep learning algorithms that operate them. Lastly, the process by which users can become aware of the prioritization strategy (i.e. the overt experiment) is tightly coupled to the recommendation list display, which assumes that users ignore prioritized items when they have knowledge about them. Nevertheless, human behavior is not something necessarily deterministic, and some users choose items even when they know about the hidden strategy. These nuances are *hard-coded* into the model, limiting its findings to a rational user activity.

Additionally, even though the experiments were implemented in increasing level of realism, users that become aware of the hidden strategy would not necessarily disregard the prioritized items completely. Users can consume (and will) consume prioritized items, regardless of their ignorance towards the main purpose of the algorithm.

5.3. Practical and Theoretical Implications

From a theoretical point of view, this work extends the discourse on RS manipulation by providing empirical evidence of its impact and mechanisms. It offers a way in which manipulation can be operationalized, by selecting the covert influence view and mapping its general conditions to inputs and outputs of a model. Moreover, it exemplifies the phenomenon as a RS with real data, showcasing its evolution and impact on users of a book reading platform. This thesis does not demonstrate that RS are inherently manipulative; if a different account of manipulation was chosen, the results would have been potentially different. Furthermore, even when manipulation can be measured, it is a matter of debate whether it is morally problematic or not. I approached the change in user preferences as an operationalization of harm, in line with existing research, which has a negative connotation. Under this assumption, the book RS that was modelled would have effectively harmed its users. Nevertheless, people do change their own preferences organically after some time. There is a high chance that my taste in literature changed over the course of my life, and blaming solely a RS for this change might be too quick to conclude.

From a practical perspective, this work offers an empirical point of reference for regulators to address the growing concern of manipulation from intelligent systems. Much has been said about the current limitations of the planned regulations for AI, specifically referring to the lack of proper definitions and ways to measure them. This work contributes to the discussion by narrowing the scope of manipulative systems to a specific account while demonstrating its effect on users. With the tremendous growth of AI development and its implications for all segments of the population, it is paramount for policymakers to understand how these systems work and how can they regulate them accordingly, without adding unnecessary obstacles for innovation while ensuring a responsible progress for all. The rate of growth requires actions taken as soon as possible, but rushing new policies can leave many manipulative actors out of the regulation. In particular, policymakers could make use of the 279.5% change in user preferences with respect to the benchmark model as a parameter for determining if a RS is influencing the user preferences of their users, by designing A/B tests where a control group of accounts is used for consuming akin items while a test group of accounts is used for consuming the top recommendations always. However, careful consideration must be placed on the type of platform that is being tested; book RS can differ from other RS. Moreover, even when a RS could be changing the user preferences of the users, the policymakers could propose mitigation strategies such as displaying information about the generation of recommendations or promoting social interactions among their users before showing the recommended items. These two mechanisms can reduce the impact of their preferences by more than half. Lastly, the difference that the proportion of ignorant population has on the change in user preferences could suggest that it is desirable for the majority of the population to become aware of the mechanisms with which these systems operate. Policymakers could take advantage of this conclusion to promote digital training for users, hence increasing their knowledge and reducing the probability of their biases being exploited by the algorithm.

5.4. Future Research

This thesis addressed manipulation under the covert influence account. Though popular, this definition has proven to be insufficient for taking into account instances of manipulation that do not require the same conditions. One further line of research could explore the operationalization and implementation of other accounts of manipulation, perhaps even comparing the results among them. This would enrich the theoretical body of both the conceptual and the applied usage of the term. For instance, how can a broad definition such as the careless view be used for modelling a RS as an ABM?

Furthermore, this thesis does not attempt to discuss the moral implications of a manipulative RS. Given that regulations are based on ethical definitions, it would help to have an empirical understanding of the consequences that changing user preferences might have. It could be the case that this form of manipulation is not problematic, since filtering and ranking books has more positive effects than negative ones for their users.

Regarding the implementation, RS are only a subset of intelligent systems. Even though they are one of the

most used ones, further research could explore simulating a different kind of system. For instance, how can autonomous cars manipulate their users or the people that are affected by its systems, such as pedestrians? While RS have been studied extensively, newer technologies pose new ventures to explore, since manipulation can arise in almost any intelligent system.

Finally, the chosen dataset of a book reviewing platform is only one example of many different RS. The decision to use it was made from a combination of public data availability and personal interest. However, there are many other systems that use RS, from art, movies and music, to health, engineering and cooking. The effect on user preferences could be different between these scenarios. Moreover, each of these creates a separate set of moral implications. Could a health RS pose the same threats to well-being than a music one? Future work could explore this comparisons.

6

Conclusion

This thesis aimed to understand how manipulation within RS affected their users' preferences by simulating a book RS using an ABM. Manipulation, understood as a form of influence from the philosophical perspective, was defined under the specific account of covert influence, which has been thoroughly studied conceptually but not empirically. The results showed that, under the covert influence conditions, a book RS can manipulate their users' preferences by suggesting items that are not aligned with a user's preferences, but that have some hidden priority for the algorithm. This effect was measured to be 3.8 times the change that a RS with no hidden priority would have. In particular, the priority was determined *a priori* as a strategy implemented by the designers of the RS, in order to maximize an internal metric (e.g. user engagement). This type of prioritization is different than an unknown "learned" priority from an intelligent system, after being trained and used extensively to make predictions. By performing multiple experiments with different parameters, the results also showed that being transparent to the users about the prioritization strategy helps them make choices that are more aligned with their own preferences, thus reducing the magnitude of the change after iterations of consumption. Similarly, allowing for interactions between users as a social network where they can follow the activity of one another reduces the magnitude of the change even further, as users akin to each other get socially influenced between themselves. Even though research has shown that social influence can in fact modify user preferences in a problematic way - through polarization or radicalization of ideas - the experiments here were mostly comparing the effect of the social influence when a RS contains a predisposed priority, as opposed to letting users come up with their own consumption strategies.

References

Adamopoulos, Panagiotis and Alexander Tuzhilin (Jan. 2015). "On Unexpectedness in Recommender Systems: Or How to Better Expect the Unexpected". en. In: *ACM Transactions on Intelligent Systems and Technology* 5.4, pp. 1–32. DOI: 10.1145/2559952. (Visited on 05/06/2024).

Aggarwal, Charu C. (2016). *Recommender Systems*. en. Cham: Springer International Publishing. DOI: 10.1007/978-3-319-29659-3. (Visited on 04/10/2024).

Anderson, Ashton et al. (Apr. 2020). "Algorithmic Effects on the Diversity of Consumption on Spotify". en. In: *Proceedings of The Web Conference 2020*. Taipei Taiwan: ACM, pp. 2155–2165. DOI: 10.1145/3366423.3380281. (Visited on 06/22/2024).

Andrada, Gloria, Robert W. Clowes, and Paul R. Smart (Aug. 2023). "Varieties of transparency: exploring agency within AI systems". en. In: *AI & SOCIETY* 38.4, pp. 1321–1331. DOI: 10.1007/s00146-021-01326-6. (Visited on 04/04/2024).

Annalyn, Ng et al. (July 2020). "Predicting Personality from Book Preferences with User-Generated Content Labels". en. In: *IEEE Transactions on Affective Computing* 11.3, pp. 482–492. DOI: 10.1109/TAFFC.2018.2808349. (Visited on 06/12/2024).

Ashton, Hal and Matija Franklin (2022). "The problem of behaviour and preference manipulation in AI systems". en. In: vol. 3087.

Baron, Marcia (2003). "Manipulativeness". en. In: *Proceedings and Addresses of the American Philosophical Association* 77.2, pp. 37–54.

Benn, Claire and Seth Lazar (Jan. 2022). "What's Wrong with Automated Influence". en. In: *Canadian Journal of Philosophy* 52.1, pp. 125–148. DOI: 10.1017/can.2021.23. (Visited on 04/04/2024).

Bermúdez, Juan Pablo et al. (May 2023). "What Is a Subliminal Technique? An Ethical Perspective on AI-Driven Influence". en. In: *2023 IEEE International Symposium on Ethics in Engineering, Science, and Technology (ETHICS)*. West Lafayette, IN, USA: IEEE, pp. 1–10. DOI: 10.1109/ETHICS57328.2023.10155039. (Visited on 03/14/2024).

Botes, Marietjie (Feb. 2023). "Autonomy and the social dilemma of online manipulative behavior". en. In: *AI and Ethics* 3.1, pp. 315–323. DOI: 10.1007/s43681-022-00157-5. (Visited on 04/23/2024).

Bouneffouf, Djallel, Amel Bouzeghoub, and Alda Lopes Gançarski (2012). "A Contextual-Bandit Algorithm for Mobile Context-Aware Recommender System". en. In: *Neural Information Processing*. Ed. by David Hutchison et al. Vol. 7665. Series Title: Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 324–331. DOI: 10.1007/978-3-642-34487-9_40. (Visited on 04/10/2024).

Burbach, Laura et al. (Sept. 2018). "User preferences in recommendation algorithms: the influence of user diversity, trust, and product category on privacy perceptions in recommender algorithms". en. In: *Proceedings of the 12th ACM Conference on Recommender Systems*. Vancouver British Columbia Canada: ACM, pp. 306–310. DOI: 10.1145/3240323.3240393. (Visited on 05/22/2024).

Burke, Robin, Alexander Felfernig, and Mehmet H Göker (2011). "Recommender Systems: An Overview". en. In: *AI Magazine* 32.3. DOI: 10.1609/aimag.v32i3.2361.

Burr, Christopher, Nello Cristianini, and James Ladyman (Dec. 2018). "An Analysis of the Interaction Between Intelligent Software Agents and Human Users". en. In: *Minds and Machines* 28.4, pp. 735–774. DOI: 10.1007/s11023-018-9479-0. (Visited on 05/20/2024).

Cai, Jie and Gang Li (Jan. 2024). "Exercise or lie down? The impact of fitness app use on users' wellbeing". en. In: *Frontiers in Public Health* 11, p. 1281323. DOI: 10.3389/fpubh.2023.1281323. (Visited on 05/01/2024).

Çano, Erion and Maurizio Morisio (Nov. 2017). "Hybrid Recommender Systems: A Systematic Literature Review". en. In: *Intelligent Data Analysis* 21.6. arXiv:1901.03888 [cs], pp. 1487–1524. DOI: 10.3233/IDA-163209. (Visited on 04/10/2024).

Carare, Octavian (Aug. 2012). "THE IMPACT OF BESTSELLER RANK ON DEMAND: EVIDENCE FROM THE APP MARKET*". en. In: *International Economic Review* 53.3, pp. 717–742. DOI: 10.1111/j.1468-2354.2012.00698.x. (Visited on 05/06/2024).

Carroll, Micah et al. (Oct. 2023). "Characterizing Manipulation from AI Systems". en. In: *Equity and Access in Algorithms, Mechanisms, and Optimization*. Boston MA USA: ACM, pp. 1–13. DOI: 10.1145/3617694.3623226. (Visited on 03/14/2024).

Cemiloglu, Deniz et al. (Jan. 2023). "Explainable persuasion for interactive design: The case of online gambling". en. In: *Journal of Systems and Software* 195, p. 111517. DOI: 10.1016/j.jss.2022.111517. (Visited on 05/02/2024).

Cercel, Dumitru-Clementin and Stefan Trausan-Matu (June 2014). "Opinion Propagation in Online Social Networks: A Survey". en. In: *Proceedings of the 4th International Conference on Web Intelligence, Mining and Semantics (WIMS14)*. Thessaloniki Greece: ACM, pp. 1–10. DOI: 10.1145/2611040.2611088. (Visited on 06/13/2024).

Chandrashekar, Ashok et al. (July 2017). *Artwork Personalization at Netflix*. (Visited on 05/01/2024).

Christiano, Thomas (Jan. 2022). "Algorithms, Manipulation, and Democracy". en. In: *Canadian Journal of Philosophy* 52.1, pp. 109–124. DOI: 10.1017/can.2021.29. (Visited on 05/20/2024).

Cohen, Shlomo (Aug. 2023). "Are All Deceptions Manipulative or All Manipulations Deceptive?" en. In: *Journal of Ethics and Social Philosophy* 25.2. DOI: 10.26556/jesp.v25i2.1998. (Visited on 04/10/2024).

Cosley, Dan et al. (2003). "Is Seeing Believing? How Recommender Interfaces Affect Users' Opinions". en. In: *NEW HORIZONS* 5.

Dam, Koen H., Igor Nikolic, and Zofia Lukszo, eds. (2013). *Agent-Based Modelling of Socio-Technical Systems*. en. Dordrecht: Springer Netherlands. DOI: 10.1007/978-94-007-4933-7. (Visited on 03/14/2024).

Dragiewicz, Molly et al. (July 2018). "Technology facilitated coercive control: domestic violence and the competing roles of digital media platforms". en. In: *Feminist Media Studies* 18.4, pp. 609–625. DOI: 10.1080/14680777.2018.1447341. (Visited on 05/02/2024).

Ehsan, Upol and Mark O. Riedl (Sept. 2021). *Explainability Pitfalls: Beyond Dark Patterns in Explainable AI*. en. arXiv:2109.12480 [cs]. (Visited on 04/10/2024).

Fischer, Alexander (May 2022). "Then again, what is manipulation? A broader view of a much-maligned concept". en. In: *Philosophical Explorations* 25.2, pp. 170–188. DOI: 10.1080/13869795.2022.2042586. (Visited on 04/29/2024).

Fogg, Brian Jeffrey (2003). *Persuasive technology: using computers to change what we think and do*. en. Vol. 2002. Morgan Kaufmann. (Visited on 05/02/2024).

Gausen, Anna, Wayne Luk, and Ce Guo (Mar. 2023). "Using Agent-Based Modelling to Evaluate the Impact of Algorithmic Curation on Social Media". en. In: *Journal of Data and Information Quality* 15.1, pp. 1–24. DOI: 10.1145/3546915. (Visited on 03/14/2024).

Gedikli, Fatih, Dietmar Jannach, and Mouzhi Ge (Apr. 2014). "How should I explain? A comparison of different explanation types for recommender systems". en. In: *International Journal of Human-Computer Studies* 72.4, pp. 367–382. DOI: 10.1016/j.ijhcs.2013.12.007. (Visited on 04/10/2024).

Genovesi, Sergio, Katharina Kaesling, and Scott Robbins, eds. (2023). *Recommender Systems: Legal and Ethical Issues*. en. Vol. 40. The International Library of Ethics, Law and Technology. Cham: Springer International Publishing. DOI: 10.1007/978-3-031-34804-4. (Visited on 04/04/2024).

Ghori, Muheeb Faizan et al. (Sept. 2021). *How does the User's Knowledge of the Recommender Influence their Behavior?* en. arXiv:2109.00982 [cs]. (Visited on 06/19/2024).

Gorin, Moti (2014). "DO MANIPULATORS ALWAYS THREATEN RATIONALITY?" en. In: *American Philosophical Quarterly* 51.1, pp. 51–61.

Grisse, Karina (2023). "Recommender Systems, Manipulation and Private Autonomy: How European Civil Law Regulates and Should Regulate Recommender Systems for the Benefit of Private Autonomy". en. In: *Recommender Systems: Legal and Ethical Issues*. Ed. by Sergio Genovesi, Katharina Kaesling, and Scott Robbins. Vol. 40. Series Title: The International Library of Ethics, Law and Technology. Cham: Springer International Publishing, pp. 101–128. DOI: 10.1007/978-3-031-34804-4_6. (Visited on 04/04/2024).

Hall, Lars, Petter Johansson, and Thomas Strandberg (Sept. 2012). "Lifting the Veil of Morality: Choice Blindness and Attitude Reversals on a Self-Transforming Survey". en. In: *PLoS ONE* 7.9. Ed. by Luis M. Martinez, e45457. DOI: 10.1371/journal.pone.0045457. (Visited on 05/02/2024).

Himelboim, Itai et al. (Jan. 2017). "Classifying Twitter Topic-Networks Using Social Network Analysis". en. In: *Social Media + Society* 3.1, p. 205630511769154. DOI: 10.1177/2056305117691545. (Visited on 06/17/2024).

Hinds, Joanne, Emma J. Williams, and Adam N. Joinson (Nov. 2020). ""It wouldn't happen to me": Privacy concerns and perspectives following the Cambridge Analytica scandal". en. In: *International Journal of Human-Computer Studies* 143, p. 102498. DOI: 10.1016/j.ijhcs.2020.102498. (Visited on 05/02/2024).

Hodgson, Geoffrey M (2012). "On the Limits of Rational Choice Theory". en. In: *Economic Thought*.

Hu, Nan et al. (Feb. 2012). "Manipulation of online reviews: An analysis of ratings, readability, and sentiments". en. In: *Decision Support Systems* 52.3, pp. 674–684. DOI: 10.1016/j.dss.2011.11.002. (Visited on 05/22/2024).

Ie, Eugene et al. (Sept. 2019). *RecSim: A Configurable Simulation Platform for Recommender Systems*. en. arXiv:1909.04847 [cs, stat]. (Visited on 03/14/2024).

Ivanova, Olga and Michael Scholz (Dec. 2017). "How can online marketplaces reduce rating manipulation? A new approach on dynamic aggregation of online ratings". en. In: *Decision Support Systems* 104, pp. 64–78. DOI: 10.1016/j.dss.2017.10.003. (Visited on 06/12/2024).

Jongepier, Fleur and Michael Klenk (June 2022). *The Philosophy of Online Manipulation*. en. 1st ed. New York: Routledge. DOI: 10.4324/9781003205425. (Visited on 03/14/2024).

Kasten, Vance (1980). "Manipulation and Teaching". In: *Journal of Philosophy of Education* 14.1, pp. 53–62. DOI: 10.1111/j.1467-9752.1980.tb00539.x.

Kerstein, Samuel (2023). "Treating Persons as Means". In: *The Stanford Encyclopedia of Philosophy*. Ed. by Edward N. Zalta and Uri Nodelman. Winter 2023. Metaphysics Research Lab, Stanford University.

Kibe, Kevin (2023). *Building a Book Recommendation System with Cosine Similarity and Tf-idf Vectorization Techniques*. (Visited on 05/22/2024).

Klenk, Michael (2020). "Digital Well-Being and Manipulation Online". en. In: *Ethics of Digital Well-Being*. Ed. by Christopher Burr and Luciano Floridi. Vol. 140. Series Title: Philosophical Studies Series. Cham: Springer International Publishing, pp. 81–100. DOI: 10.1007/978-3-030-50585-1_4. (Visited on 03/14/2024).

— (Jan. 2022). "(Online) manipulation: sometimes hidden, always careless". en. In: *Review of Social Economy* 80.1, pp. 85–105. DOI: 10.1080/00346764.2021.1894350. (Visited on 03/14/2024).

Kligman, Michael and Charles Culver (1992). "An Analysis of Interpersonal Manipulation". In: *Journal of Medicine and Philosophy* 17.2, pp. 173–197. DOI: 10.1093/jmp/17.2.173.

Kopp, Carlo, Kevin B. Korb, and Bruce I. Mills (Nov. 2018). "Information-theoretic models of deception: Modelling cooperation and diffusion in populations exposed to "fake news"". en. In: *PLOS ONE* 13.11. Ed. by Yong Deng, e0207383. DOI: 10.1371/journal.pone.0207383. (Visited on 03/14/2024).

Kotkov, Denis et al. (Apr. 2018). "Investigating serendipity in recommender systems based on real user feedback". en. In: *Proceedings of the 33rd Annual ACM Symposium on Applied Computing*. Pau France: ACM, pp. 1341–1350. DOI: 10.1145/3167132.3167276. (Visited on 07/20/2024).

Kousha, Kayvan, Mike Thelwall, and Mahshid Abdoli (Aug. 2017). "Goodreads reviews to assess the wider impacts of books". en. In: *Journal of the Association for Information Science and Technology* 68.8, pp. 2004–2016. DOI: 10.1002/asi.23805. (Visited on 06/12/2024).

Liao, Mengqi, S. Shyam Sundar, and Joseph B. Walther (Apr. 2022). "User Trust in Recommendation Systems: A comparison of Content-Based, Collaborative and Demographic Filtering". en. In: *CHI Conference on Human Factors in Computing Systems*. New Orleans LA USA: ACM, pp. 1–14. DOI: 10.1145/3491102.3501936. (Visited on 03/14/2024).

Liu, Yidan, Min Xie, and Laks V.S. Lakshmanan (Oct. 2014). "Recommending user generated item lists". en. In: *Proceedings of the 8th ACM Conference on Recommender systems*. Foster City, Silicon Valley California USA: ACM, pp. 185–192. DOI: 10.1145/2645710.2645750. (Visited on 06/12/2024).

Martin, Gregory J and Ali Yurukoglu (2017). "Bias in Cable News: Persuasion and Polarization". en. In: *National Bureau of Economic Research*.

Mathew, Praveena, Bincy Kuriakose, and Vinayak Hegde (2016). "Book Recommendation System through content based and collaborative filtering method". In: *SAPIENCE*. India: IEEE, pp. 47–52. DOI: 10.1109/SAPIENCE.2016.7684166.

McCloskey, H.J. (1980). "Coercion: Its nature and significance". In: *Souther Journal of Philosophy* 18.3, pp. 335–351. DOI: 10.1111/j.2041-6962.1980.tb01390.x. (Visited on 05/02/2024).

Meske, Christian et al. (Jan. 2022). "Explainable Artificial Intelligence: Objectives, Stakeholders, and Future Research Opportunities". en. In: *Information Systems Management* 39.1, pp. 53–63. DOI: 10.1080/10580530.2020.1849465. (Visited on 04/04/2024).

Mills, Claudia (1995). "Politics and Manipulation". In: *Social Theory and Practice* 22.1, pp. 97–112. DOI: 10.5840/soctheorpract199521120.

Mitchell, Thomas and Thomas Douglas (Mar. 2024). "Wrongful Rational Persuasion Online". en. In: *Philosophy & Technology* 37.1, p. 35. DOI: 10.1007/s13347-024-00725-z. (Visited on 05/02/2024).

Noggle, Robert (1996). "Manipulative Actions: A Conceptual Analysis". In: *American Philosophical Quarterly* 33.1, pp. 43–55.

— (Sept. 2021). "Manipulation in Politics". en. In: *Oxford Research Encyclopedia of Politics*. Oxford University Press. DOI: 10.1093/acrefore/9780190228637.013.2012. (Visited on 05/01/2024).

— (2022). "The Ethics of Manipulation". In: *The Stanford Encyclopedia of Philosophy*. Ed. by Edward N. Zalta. Summer 2022. Metaphysics Research Lab, Stanford University.

Peng, Sancheng, Guojun Wang, and Dongqing Xie (Jan. 2017). "Social Influence Analysis in Social Networking Big Data: Opportunities and Challenges". en. In: *IEEE Network* 31.1, pp. 11–17. DOI: 10.1109/MNET.2016.1500104NM. (Visited on 06/13/2024).

Perrin, Andrew (2016). *Book Reading 2016*. Tech. rep. Pew Research Center, p. 4.

Richens, Jonathan G., Rory Beard, and Daniel H. Thompson (Nov. 2022). *Counterfactual harm*. en. arXiv:2204.12993 [cs, stat]. (Visited on 05/06/2024).

Ross, Björn et al. (July 2019). "Are social bots a real threat? An agent-based model of the spiral of silence to analyse the impact of manipulative actors in social networks". en. In: *European Journal of Information Systems* 28.4, pp. 394–412. DOI: 10.1080/0960085X.2018.1560920. (Visited on 03/14/2024).

Roy, Deepjyoti and Mala Dutta (Dec. 2022). "A systematic review and research perspective on recommender systems". en. In: *Journal of Big Data* 9.1, p. 59. DOI: 10.1186/s40537-022-00592-5. (Visited on 04/04/2024).

Sar Shalom, Oren et al. (Apr. 2016). "Beyond Collaborative Filtering: The List Recommendation Problem". en. In: *Proceedings of the 25th International Conference on World Wide Web*. Montréal Québec Canada: International World Wide Web Conferences Steering Committee, pp. 63–72. DOI: 10.1145/2872427.2883057. (Visited on 06/13/2024).

Seaver, Nick (Dec. 2019). "Captivating algorithms: Recommender systems as traps". en. In: *Journal of Material Culture* 24.4, pp. 421–436. DOI: 10.1177/1359183518820366. (Visited on 04/04/2024).

Spencer, Shaun B. (2019). "The Problem of Online Manipulation". en. In: *SSRN Electronic Journal*. DOI: 10.2139/ssrn.3341653. (Visited on 05/01/2024).

Susser, Daniel and Vincent Grimaldi (July 2021). "Measuring Automated Influence: Between Empirical Evidence and Ethical Values". en. In: *Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society*. Virtual Event USA: ACM, pp. 242–253. DOI: 10.1145/3461702.3462532. (Visited on 04/04/2024).

Susser, Daniel, Beate Roessler, and Helen Nissenbaum (June 2019). "Technology, autonomy, and manipulation". en. In: *Internet Policy Review* 8.2. DOI: 10.14763/2019.2.1410. (Visited on 03/14/2024).

Susser, Daniel, Beate Roessler, and Helen F. Nissenbaum (2018). "Online Manipulation: Hidden Influences in a Digital World". en. In: *SSRN Electronic Journal*. DOI: 10.2139/ssrn.3306006. (Visited on 04/23/2024).

Ten Broeke, Guus, George Van Voorn, and Arend Ligtenberg (2016). "Which Sensitivity Analysis Method Should I Use for My Agent-Based Model?" en. In: *Journal of Artificial Societies and Social Simulation* 19.1, p. 5. DOI: 10.18564/jasss.2857. (Visited on 06/10/2024).

Thaler, Richard H. and Cass R. Sunstein (2008). *Nudge: improving decisions about health, wealth, and happiness*. en. New Haven (Conn.): Yale university press.

Thiele, Jan C., Winfried Kurth, and Volker Grimm (2014). "Facilitating Parameter Estimation and Sensitivity Analysis of Agent-Based Models: A Cookbook Using NetLogo and 'R'". en. In: *Journal of Artificial Societies and Social Simulation* 17.3, p. 11. DOI: 10.18564/jasss.2503. (Visited on 06/10/2024).

Tintarev, Nava and Judith Masthoff (Oct. 2012). "Evaluating the effectiveness of explanations for recommender systems: Methodological issues and empirical studies on the impact of personalization". en. In: *User Modeling and User-Adapted Interaction* 22.4-5, pp. 399–439. DOI: 10.1007/s11257-011-9117-5. (Visited on 04/10/2024).

Van Dam, Andrew (2024). "How many books did you read in 2023? Are you in the top 1 percent?" In: *The Washington Post*.

Van Den Eede, Yoni (May 2011). "In Between Us: On the Transparency and Opacity of Technological Mediation". en. In: *Foundations of Science* 16.2-3, pp. 139–159. DOI: 10.1007/s10699-010-9190-y. (Visited on 04/04/2024).

Wan, Mengting and Julian McAuley (Sept. 2018). "Item recommendation on monotonic behavior chains". en. In: *Proceedings of the 12th ACM Conference on Recommender Systems*. Vancouver British Columbia Canada: ACM, pp. 86–94. DOI: 10.1145/3240323.3240369. (Visited on 06/12/2024).

Wang, Hao (Sept. 2022). "Transparency as Manipulation? Uncovering the Disciplinary Power of Algorithmic Transparency". en. In: *Philosophy & Technology* 35.3, p. 69. DOI: 10.1007/s13347-022-00564-w. (Visited on 04/04/2024).

Wei, Tianxin et al. (Aug. 2021). "Model-Agnostic Counterfactual Reasoning for Eliminating Popularity Bias in Recommender System". en. In: *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*. Virtual Event Singapore: ACM, pp. 1791–1800. DOI: 10.1145/3447548.3467289. (Visited on 04/17/2024).

Wijnhoven, Fons and Oscar Bloemen (Mar. 2014). "External validity of sentiment mining reports: Can current methods identify demographic biases, event biases, and manipulation of reviews?" en. In: *Decision Support Systems* 59, pp. 262–273. DOI: 10.1016/j.dss.2013.12.005. (Visited on 06/12/2024).

Wilson, Robert A. and Frank C. Keil, eds. (1999). *The MIT encyclopedia of the cognitive sciences*. en. Cambridge, Mass: MIT Press.

Wood, Allen W. (2014). "Coercion, Manipulation, Exploitation". en. In: *Manipulation: Theory and Practice*. Ed. by Christian Coons and Michael Weber. Oxford University Press, pp. 17–50. (Visited on 05/02/2024).

Woodlock, Delanie et al. (July 2020). "Technology as a Weapon in Domestic Violence: Responding to Digital Coercive Control". en. In: *Australian Social Work* 73.3, pp. 368–380. DOI: 10.1080/0312407X.2019.1607510. (Visited on 05/02/2024).

Yesilada, Muhsin and Stephan Lewandowsky (Mar. 2022). "Systematic review: YouTube recommendations and problematic content". en. In: *Internet Policy Review* 11.1. DOI: 10.14763/2022.1.1652. (Visited on 03/14/2024).

Yeung, Karen (Jan. 2017). "'Hypernudge': Big Data as a mode of regulation by design". en. In: *Information, Communication & Society* 20.1, pp. 118–136. DOI: 10.1080/1369118X.2016.1186713. (Visited on 05/22/2024).

Yin, Xicheng et al. (Oct. 2019). "Agent-based opinion formation modeling in social network: A perspective of social psychology". en. In: *Physica A: Statistical Mechanics and its Applications* 532, p. 121786. DOI: 10.1016/j.physa.2019.121786. (Visited on 03/14/2024).

Zhang, Guangqian and Wei Sun (2012). "User preferences to attributes of books for personalized recommendation". In: Beijing: IEEE, pp. 681–684. DOI: 10.1109/ICSESS.2012.6269558.

Zhang, Jingjing et al. (Mar. 2020). "Consumption and Performance: Understanding Longitudinal Dynamics of Recommender Systems via an Agent-Based Simulation Framework". en. In: *Information Systems Research* 31.1, pp. 76–101. DOI: 10.1287/isre.2019.0876. (Visited on 03/14/2024).

Zhu, Zhengbang et al. (July 2024). "Understanding or Manipulation: Rethinking Online Performance Gains of Modern Recommender Systems". en. In: *ACM Transactions on Information Systems* 42.4, pp. 1–32. DOI: 10.1145/3637869. (Visited on 03/14/2024).

Ziegler, Cai-Nicolas and Jennifer Golbeck (Mar. 2007). "Investigating interactions of trust and interest similarity". en. In: *Decision Support Systems* 43.2, pp. 460–475. DOI: 10.1016/j.dss.2006.11.003. (Visited on 06/13/2024).

A

Theoretical Background

A.1. What is manipulation?

The problem of defining manipulation is one of setting boundaries. Roughly speaking, manipulation is a kind of influence on someone (Susser, Roessler, and H. F. Nissenbaum, 2018). In order to demarcate the subset of cases characteristic of manipulation, we can review the conceptual analysis performed by Noggle (1996), since it is one of the most systematic works on the topic (Susser, Roessler, and H. Nissenbaum, 2019). Within it, Noggle exposes a series of cases of influence that have mostly the same format:

> An actor *A* (i.e. the manipulator) acts in a way *B* (i.e. the action) over an actor *C* (i.e. the target) in such a way that *A* gets achieves something from *B* by influencing *C*

And analyzes them under different accounts of influence at that time. This way of describing manipulation has multiple ways of being interpreted. Hence, each component requires further analysis to narrow down the definition to an elemental conceptualization. Nevertheless, there is not a unique way of demarcating what could or could not be considered manipulation (Susser, Roessler, and H. Nissenbaum, 2019). For instance, one could focus on each of the components *A*, *B* or *C* to define what manipulation is or is not. Therefore, the main accounts of manipulation relevant to this work will be briefly explained.

A.1.1. Accounts of Manipulation

This section begins by exposing the three accounts of manipulation described by Noggle (2022), with the most relevant piece of content from each for this research. When appropriate, an example from a digital platform is provided.

A.1.1.1. Manipulation as Bypassing Rationality

Under this view, manipulation is a process that subverts the rational capacities of the target (Gorin, 2014). Persons have a "rational self" that is in charge of thinking based on the capacity to understand a situation from their own perspective and make a decision. How this rational self is defined can lead to different instances of manipulative behavior (Jongepier and Klenk, 2022). For instance, our rational capacities can be related to the accordance between our own beliefs and basic logic (Gorin, 2014). Hence, manipulation could be a way of bypassing these rational capacities in order to reach the more primitive decision-making processes of an individual, in order to alter their behavior. Yet, the usage of the term "bypass" can also be problematic, as it is not clear that acting emotionally or irrationally would directly imply a subversion of the rational capacities,

since users might still be acting with full understanding of the situation. Moreover, even if the target is being subject to a manipulative action under this definition, it could be the case that the manipulator is not being immoral (Jongepier and Klenk, 2022). For instance, if tobacco smokers become nauseated by the explicit images of sick patients on cigarette packs, they could make a decision not to buy them, even when they do not fully understand the hazardous implications that smoking has on their health; they could be making the decision based only on a visceral feeling towards the image.
For the sake of this work, the most important aspect of this account is the bypassing understood as a way to hide the intention of the manipulator from the rational capacities of the target, as users might have a different conception of what the intentions of a RS are.

From the perspective of digital platforms, a good example of manipulation as bypassing rationality comes from Netflix's personalized artwork (Chandrashekar et al., 2017): they have multiple thumbnails for their main content items that get displayed to users based on their history of activity. If a user has a preference for a specific actress, they will prioritize artwork with that actress, appealing to psychological factors that go beyond the rational capacity. This could result in a user choosing to consume an item based on their gut that they would not have consumed if properly assessed otherwise.

A.1.1.2. Manipulation as Trickery

Under this view, manipulation is a process that induces a faulty mental state on the target (Noggle, 2021). If someone is tricked, they might perform actions that are not in their self-interest, as they would result in outcomes that go against their ideals and beliefs (C. Mills, 1995). As with the bypassing rationality account, there is some debate as to how "faulty mental state" is defined (Noggle, 2022). For instance, it could be understood as a difference between what one intends to do and what one actually does (Kasten, 1980) or it could be understood as the difference between pleasant and unpleasant outcomes from our actions (Fischer, 2022). Additionally, there is a close connection between this account and deception, although the nature of this relationship is not clear: is manipulation a subset of deception or is deception a subset of manipulation (Cohen, 2023)? It seems like there is a partial overlap between them with no vertical hierarchy (Cohen, 2023). Furthermore, trickery as causing a faulty state is naturally associated with a bad action for the target, which would categorize this account of manipulation as an immoral one. Yet, some argue that there is an objectionable way of understanding the concept since there is always a free will from the target to choose one option or the other, regardless of the presentation of the options as pleasant or unpleasant (Fischer, 2022).
For the sake of this work, the most important aspect of this account is the emphasis on users acting on their best self-interest, since this requires some clear understanding of what one's beliefs and ideals are.

From the perspective of digital platforms, a good but morally biased example comes from marketing as digital advertisements (Spencer, 2019). Marketers are aware of the biases and vulnerabilities that people have, and they exploit them to induce a sense of necessary opportunities in the users of digital platforms. The users then consume an item (which in most cases with advertisements results in an economic transaction) based on a mental state which they were subjected to.

A.1.1.3. Manipulation as Pressure

Under this view, manipulation is a process where the target is influenced by some form of pressure (Kligman and Culver, 1992). The manipulator makes it difficult to resist the influence that they are exerting, making the target subject to a decision that is based on a type of fear. The problematic with this account is that coercion could be seen as a form of pressure taken to the extreme, thus manipulation as pressure and coercion would be defined under the same continuum (Noggle, 2022). One could assert that a mild level of pressure can be labeled as manipulation, but then what is the difference between mild and high? It seems like the continuous nature of such a social mechanism is highly dependent on a subjective interpretation, even though there have been attempts at properly defining these boundaries, for example by taking other properties into consideration

(Baron, 2003).
For the sake of this work, the most important aspect to take into consideration is the non requested aspect of such pressure, as some digital platforms can make use of different display mechanisms to prioritize certain recommendations without the user's explicit demand.

From the perspective of digital platforms, a good example is the influence that fitness apps have on users' behaviour (Cai and Li, 2024). Some of these apps use a combination of social and personal notifications to promote physical activity for their users. Despite the potential reasons of engagement from the app, users respond to such influences by increasing the activity. This has been shown to have a positive impact on users' well-being (Cai and Li, 2024). However, the apps exert this influence by putting pressure on a user, either because they will not reach their goals (i.e. personal) or because a friend in common is beating them (i.e. social). Even though the outcome could be positive, it could be seen as a form of manipulation as pressure.

A.1.2. Persuasion and Coercion

One of the most used ways to describe what manipulation is as a form of influence that is neither persuasion or coercion (Noggle, 2022). More than being the complement of both, the distinction seems to refer to a linear continuum, similar to the one proposed in section A.1.1.3, where manipulation lies in between both terms as in figure A.1. Yet, the boundaries require some precision. Hence, both persuasion and coercion are briefly discussed here. Even though Noggle finds this model somewhat inaccurate (Noggle, 2022), I choose to include it since overt RS appeal to a user's rational capacities to understand the system's intentions to make a knowledgeable choice, which could be seen as a shift from manipulation towards persuasion.

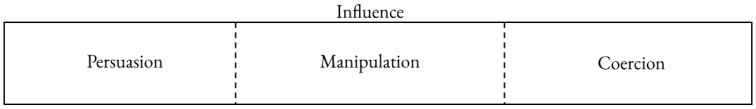


Figure A.1: Distinction between persuasion, manipulation and coercion as forms of influence

A.1.2.1. Persuasion

As with the term manipulation, persuasion can be understood in a broad and a narrow way, where the broad view refers to all forms of influence exerted on someone (similarly to manipulation as pressure) and the narrow view refers to rational persuasion in particular (Susser, Roessler, and H. F. Nissenbaum, 2018). Since this section is aimed at demarcating the concept of manipulation and differentiating it from other forms of influence, the narrow view is discussed hereby. As exactly opposed to the manipulation by bypassing rationality account, rational persuasion refers to an influence that appeals to the rational capacities of an individual by giving reasons for them to reflect on and evaluate (Susser, Roessler, and H. F. Nissenbaum, 2018). The act of decision-making in this case is based on rational choice theory, where it is assumed *a priori* that individuals have well-defined preferences that obey rational behaviour (Wilson and Keil, 1999). Even though defining individuals as purely rational decision makers has been shown to be problematic (Hodgson, 2012), the abstract idea is useful for this work as it distinguishes between two components of the influence: expanding the decision space and changing the decision-making process (Susser, Roessler, and H. F. Nissenbaum, 2018). The former refers to increasing the number of options available for an individual to choose from, whereas the latter refers to modifying the ways in which the individual understands an option. Persuading someone can be caused by using any of these two components. Lastly, this form of influence is most of the times seen as morally unproblematic, although some have argued that it could still be wrongful when certain conditions are met (Mitchell and Douglas, 2024).

Within the scope of digital platforms, much of the discussion around persuasion in general refers to the broad view of the term, for instance with persuasive technology (Fogg, 2003). Moreover, an example specific

to the rational persuasion mentioned here is the case of explainable persuasion, where digital platforms aim to disclose information about their design through explanations that are in many cases created with XAI (Cemiloglu et al., 2023). For example, by providing users of an online gambling platform with explanations about the games they were playing, users took more rational decisions than the ones typically exploited in risk assessment scenarios (Cemiloglu et al., 2023).

A.1.2.2.Coercion

As opposed to persuasion, coercion leaves no choice or no acceptable choice to the target when making a decision (Wood, 2014). The emphasis on acceptable choice is hereby taken, since there is some capacity to make a conscious decision even when an individual is deprived from a choice (Susser, Roessler, and H. F. Nissenbaum, 2018); the alternative becomes unacceptable. The coercer may remove the options that are acceptable but the target is still the one to make the decision by understanding the situation. For instance, a manager that wishes to impose extra work on someone who is highly economically dependent on their role might threaten the employee to fire them or punish them if they do not accept the extra work. The employee feels like the only acceptable option is to do the extra work even though they could quit and search for a new job if desired. The consequences of such a decision might be negative so they choose to take the extra work. Coercion is usually related to an imbalance of power, where the coercer has a position of power with respect to the target (McCloskey, 1980). For these reasons, coercion is typically seen as a morally problematic form of influence, since it interferes with the target's options, undermining their autonomy (Noggle, 2022).

In the realm of digital platforms, the term has been used to described coercive behavior from the users of social media in particular (Dragiewicz et al., 2018). Even though the cases are clear situations of coercion, they are mostly related to peer-to-peer interactions, where the platforms serve only as a medium. For instance, a user harassing a target by posting intimate content without their consent if they do not get something in return is a form of coercion employed on this platforms. However, the coercer is the user, not the platform, although it can be debatable if prioritizing coercive content results in some responsibility from the platform (Woodlock et al., 2020).

A.1.2.3.Relation with Manipulation

Perhaps the main difference between manipulation and these two other forms of influence is that manipulators transfer the actual decision-making process away from the target to the manipulator, as opposed to persuasive and coercive actors who influence their targets but leave the conscious choice to the user (Susser, Roessler, and H. F. Nissenbaum, 2018). Even when they are faced with unacceptable options, it is assumed that targets of persuasion and coercion have the understanding of the situation and make a decision based on an evaluation, whereas the hidden component of manipulation can lead to targets feeling like they are making a decision by themselves when in fact they unconsciously succumbed to the manipulator's intentions.

A.1.3. Manipulation and Deception

Deception refers to making someone hold a false belief about something[1]. Even though manipulation (particularly under the trickery account) might seem like a deceptive action, both concepts are related in a partially overlapping way, with some manipulations being deceptive and some deceptions being manipulative, but not all instances of one belong to the other (Cohen, 2023). It has been pointed out that deception can be intentionless, rendering most traditional views of manipulation non deceptive for this cases. However, for the scope of this research, it is suggested that digital platforms are aware of the intentions by which they designed their systems. In such a case, the question is less about the existence or not of an intention (Klenk, 2022), but about its appearance to the user. Let us take as an example a digital platform that uses personalization for their recommendations. If we assume rationality among their users, then the user should be aware of their own interests and choose to consume content based on such beliefs. However, could it be the case that there is a mismatch

[1] Merriam-Webster

between the actual preferences of a user and their digital profile, in such a way that the user is deceived upon thinking that they like or dislike what the platforms tells them? If the answer is affirmative, then such a platform would be clearly deceiving their users. It would not matter what the intentions of the algorithm are, only that the process of reaching their goal remains hidden from the user.

A.2. Covert Influence

Regarding technology, one of the most cited requirements for manipulation to occur is the covert intention from the manipulator (Susser, Roessler, and H. Nissenbaum, 2019). According to this condition, an agent is being manipulated when someone exploits their decision-making vulnerabilities to intentionally and covertly influence their decision-making process. This means that the agent is not aware of the way in which they are being influenced. This is closely related to Noggle's proposal of manipulation as non-rational influence, since the target could have made a different decision had they not been influenced. Moreover, the hidden aspect of this influence could be related to the intention of the manipulator or to the action itself. Even when the intent of the manipulator becomes overt for the target, the action could still be manipulative.

Exploitation of decision-making vulnerabilities can be understood as the exploitation of the cognitive biases of users, as seen from a nudging perspective. Nudging refers to the alteration someone's decision-making context in order to influence their outcomes (Thaler and Sunstein, 2008). Even though nudging is not necessarily morally wrong, it do wrong to a user when the alteration of the context does not benefit the user or society. One important aspect of this relation is that nudging might stop being exploitation when a user becomes conscious of the bias, since they can make decisions on a fully aware rational state[2]. Therefore, Susser et al. (Susser, Roessler, and H. Nissenbaum, 2019) are clear on the emphasis of the "hidden" condition: when a user finds out the intention of the manipulator they stop being manipulated. However, this creates a problem with the separation between action and intention described above.

As it will be seen in the following sections, for RS in particular, users might still be manipulated even when finding out either the intentions of the system designers or the inner workings of the machine learning algorithms behind the recommendations. This has led to some researchers to propose new definitions of manipulation, such as the careless influence account by Klenk (Klenk, 2022). In this sense, the exploitation is not so much a deception (users are not being made to believe false things) but an ignorance.

A.3. Harm

One of the main goals of the EU AI Act is to avoid any harmful practices from AI systems[3]. In particular, characterizing manipulation from such systems is a derived goal, since the term has a negative connotation (Carroll et al., 2023). However, not all instances of manipulation are necessarily harmful. As it has been discussed already, there are manipulative cases that are not morally problematic. Moreover, harm is typically associated with a threat to an individual's well-being, even if it is not our physical well-being (Klenk, 2020). Two of the ways in which harm can be introduced as a consequence of manipulative actions are frustration of self-interest and loss of autonomy (Jongepier and Klenk, 2022). Since this research is concerned with the effect that RS have on their users, these concepts are at the core of the research question, thus they are described briefly in this section.

A.3.1. Frustration of Self-Interest

One of the main assumptions of manipulation is that it typically goes against the target's self-interest (Jongepier and Klenk, 2022). This has been discussed as part of every account of manipulation described so far. Under this assumption, individuals have a conception of their own beliefs, preferences and ideals, which requires

[2] Ethical consideration for nudging - British Parliament
[3] EU AI Act

some form of rational capacity, thus frustration of self-interest would be directly related to a frustration of the rational capacities of an individual (Gorin, 2014). However, a wider definition or rationality should be adopted, since manipulating an individual who holds false beliefs would not be harmful even though it could be frustrating their self-interest.

When it comes to digital systems, the above definition of frustration of self-interest can be applied to the modification of user preferences. For instance, it has been shown that some users of digital platforms that have personalized profiles fail to identify changes made to their preferences, suggesting that these platforms could in fact frustrate the self-interest of their users without them even being aware of (Hall, Johansson, and Strandberg, 2012).

A.3.2. Undermining Autonomy

In practice, autonomy refers to respecting an individual's rights and choices (Botes, 2023). This is closely related to the Kantian claim that people should be treated as personas and not as a means to an end (Kerstein, 2023). It is related to frustration of self-interest in the sense that its undermining is an outcome of manipulation but it is not necessarily a requirement for an action to be manipulative (Jongepier and Klenk, 2022). Related to digital platforms, a threat to the individual capacity of a person to make their own choices is a concern of the EU AI Act, since it goes against the principles of transparency and fairness. By exploiting biases and vulnerabilities from users, these platforms can reduce the options available or alter their decision-making process. A good example of this is Cambridge Analytica, where the detailed profiling from Facebook helped to target users who were unsure of their voting decision in order to alter their opinions, effectively undermining their autonomy through biased content (Hinds, Williams, and Joinson, 2020).

A.4. Recommender Systems

A RS is basically a filtering tool that uses data from two types of entities - users and items - to obtain and rank results (Roy and Dutta, 2022). They have been widely used since the beginnings of the world-wide-web, when navigating the vast amounts of data available quickly became an issue. Nowadays, RS have evolved into complex algorithms that mostly use AI to select the content that users get (Grisse, 2023). They are everywhere, from music to e-commerce platforms. Even though RS per se are not problematic, there have been multiple concerns about their implications for society and how they can influence human behavior (Susser, Roessler, and H. Nissenbaum, 2019). The EU Digital Services Act (DSA) even contains a definition of what constitutes a RS[4]. The main rationale behind this is that there could be a gap between the user's interests and the system's interests (Grisse, 2023).

In their beginnings, RS were optimized to find and rank the best results among a set of items by calculating the root mean square error (RMSE) between the user and the content, yielding recommendations that were completely based on the user's preferences. However, they quickly transitioned to engagement algorithms, calculating the items that would maximize the attention of the users in the platform (Susser and Grimaldi, 2021). When such systems align the engagement with the profiles of the users, RS can be morally neutral. Yet, as recent regulations of digital platforms have shown, this is not the case for all RS (Seaver, 2019).

A.4.1. Types

There are many ways in which a RS can be implemented. The core ideas are based on predicting a user's recommendation by the similarity between the item and the user's preferences. Initially, there were two ways to do so: 1) *content-based*, using the item's content to find the most suitable option, and 2) *collaborative filtering*, using another user's item preferences to find the most suitable item. As research and development of RS progressed, new forms of recommendations appeared, including variations of these existing two and completely new

[4] EU Digital Services Act

methods (Aggarwal, 2016). For instance, *demographic filtering* is a variation of collaborative filtering where the most similar user is extracted by comparing the demographic data of the user, whereas *bandits* are random recommendations for exploring new directions by introducing some variability to the overall system (Bouneffouf, Bouzeghoub, and Gançarski, 2012). Nowadays, most RS are *hybrid*: they employ a combination of multiple methods in a dynamic way (Çano and Morisio, 2017). This has greatly improved the inherent and perceived accuracy of the recommendations for the users, but they have become increasingly complex and difficult to explain. However, most of them still use both content-based and collaborative filtering as the main sources of recommendations. Thus, this research will explore the use of collaborative filtering for an initial extraction of the most similar user, followed by content-based for the ranking of the user's items in order to get the predictions sorted.

A.4.2. Explanations

In order to give a better experience for the users of a RS, designers incorporated the use of explanations for the reasons why a RS gave a certain recommendation. Originally, the aim of such explanations was to improve the efficiency, effectiveness, persuasiveness, transparency, satisfaction, scrutability and trust of a RS (Tintarev and Masthoff, 2012). Nevertheless, privacy concerns about user data management have accelerated the rise in explainable AI (XAI) as a way to provide insights into black-box machine learning techniques (Meske et al., 2022).

From the perspective of the users, these explanations can help them feel like they are making a better decision that suits their needs (Gedikli, Jannach, and Ge, 2014). However, it is important to clarify what the explanations are about. They use different methods for displaying the parameters that had the largest weight for the predictions of the system. For example, when a user within the Amazon platform gets a list of recommendations, they are normally shown alongside a disclaimer that points to other similar users' activity (e.g. "other users also bought...") (Gedikli, Jannach, and Ge, 2014). Using the concepts discussed in 2.2.4, these explanations are providing reflective transparency, offering information of how an algorithm works. The transparency-in-use remains hidden. Moreover, despite the perceived benefits of explanations for a user's comprehension of their decision-making process, opening up the inner workings of an algorithm does not necessarily erase the manipulative action from the situation (Ehsan and Riedl, 2021). Regardless of the concerns for the effectiveness of XAI as an anti-manipulation method, this research will not explore its discussion. The explanations that will be modelled will be the exposure of the hidden intentions (if existent) of the system designers.

B

Code Fragments

The entire code base of the model can be found on the project's public Github repository. It follows a very similar folder structure as the general Mesa examples. The few differences are related to the Jupyter notebooks used for running the model and performing the analysis, as well as the data pre-processing functions, which are specific to the GoodReads dataset.

This appendix contains the most important Python scripts, which correspond to the model and agents files.

B.1. model.py

```
import mesa
import pandas as pd
import numpy as np
from model.agents import ItemAgent, UserAgent
from data.data_preparation import get_model_df, get_users_df, get_items_df,
    get_categories
from data.results import Results


def get_vector(agent: mesa.Agent) -> np.array:
    """
    Helper method to get user agent vector for data collection
    """
    if isinstance(agent, UserAgent):
        return agent.vector.copy()
    return None

def get_books_consumed(agent: mesa.Agent) -> list[int]:
    """
    """
    if isinstance(agent, UserAgent):
        return agent.books_consumed.copy()
    return None

def get_agent_type(agent: mesa.Agent) -> str:
    """
    Helper method to get agent model type
    """
    return agent.__class__.__name__

```

```python
def get_item_n_read(agent: mesa.Agent) -> int:
    """
    Helper method to get count of item reads
    """
    if isinstance(agent, ItemAgent):
        return agent.n_read
    return None

def get_item_n_reviews(agent: mesa.Agent) -> int:
    """
    Helper method to get count of item reviews
    """
    if isinstance(agent, ItemAgent):
        return agent.n_reviews
    return None

def get_item_mean_rating(agent: mesa.Agent) -> int:
    """
    Helper method to get mean item rating
    """
    if isinstance(agent, ItemAgent):
        return agent.mean_rating
    return None


class RecommenderSystemModel(mesa.Model):
    """
    Recommender system model
    """

    def __init__(
        self,
        n_users: int = 2,
        steps: int = 1,
        priority: str | None = None,
        dummy: bool = False,
        seed: int | None = None,
        thresholds: tuple[int, int, int] = [5, 20, 50],
        ignorant_proportion: float = 1.0,
        rec_engine: str = "content-based",
        df: pd.DataFrame = pd.DataFrame(),
        df_items: pd.DataFrame = pd.DataFrame(),
        df_users: pd.DataFrame = pd.DataFrame(),
        initial_store_path: list[str] | None = None,
        n_recs: int = 50,
        social_influence: bool = False,
        run_type: str = "results",
        verbose: bool = False
    ):
        """
        Create a new recommender system model instance

        Args:
            n_users: number of users to simulate as agents
            steps: number of steps per simulation run
            priority: item priority for hidden agenda
            dummy: use of pre-loaded data for faster data loading
            seed: random state seed for sampling users
            thresholds: book limit thresholds for low-mid and mid-high reader personas
            ignorant_proportion: proportion of users that are ignorant to the
                intentions of algorithm
            rec_engine: type of RS ('content-based' or 'collaborative-filtering')
```

```
91              df: model df if already loaded
92              df_items: items df if already loaded
93              df_users: users df if already loaded
94              initial_store_path: path to preloaded files or None to store new files
95              n_recs: number of recommendations to return
96              social_influence: whether recommendations can be prioritized based on
                    social influence
97              run_type: 'results' or 'sensitivity' for sensitivity analysis
98          """
99
100         # Model initialization
101         if verbose:
102             print("Initializing␣model...\n")
103         super().__init__()
104         self.num_users = n_users
105         self.schedule = mesa.time.RandomActivation(self)
106         self.steps = steps
107         self.priority = priority
108         self.csv_filepaths = []
109         self.rec_engine = rec_engine
110         self.n_recs = n_recs
111         self.social_influence = social_influence
112         self.run_type = run_type
113         self.verbose = verbose
114
115         # Check at least 2 users
116         if self.num_users < 2:
117             raise Exception("At␣least␣2␣users␣expected.")
118
119         # Model dataframe extraction
120         if df.empty:
121             df = get_model_df(
122                 sample_users=n_users,
123                 dummy=dummy,
124                 seed=seed,
125                 thresholds=thresholds,
126                 ignorant_proportion=ignorant_proportion
127             )
128
129         # Items dataframe extraction
130         if df_items.empty:
131             df_items = get_items_df(
132                 df=df,
133                 priority=self.priority,
134                 verbose=self.verbose
135             )
136         len_df_items = len(df_items)
137
138         # Users dataframe extraction
139         if df_users.empty:
140             df_users = get_users_df(
141                 df=df,
142                 df_items=df_items,
143                 thresholds=thresholds,
144                 n_recs=self.n_recs,
145                 social_influence=self.social_influence,
146                 ignorant_proportion=ignorant_proportion,
147                 seed=seed,
148                 verbose=self.verbose
149             )
150         len_df_users = len(df_users)
151
```

```
152         # User agents creation
153         if self.verbose:
154             print("Creating␣user␣agents...")
155         df_users["unique_id"] = range(1, len_df_users + 1)
156         user_agents = df_users.apply(self.create_user, axis=1)
157         for a in user_agents:
158             self.schedule.add(a)
159         if self.verbose:
160             print(f"␣␣␣␣-␣Users␣added")
161 
162         # Item agents creation
163         if self.verbose:
164             print("Creating␣item␣agents...")
165         df_items["unique_id"] = range(len_df_users + 1, len_df_users + 1 + len_df_items
                )
166         item_agents = df_items.apply(self.create_item, axis=1)
167         for i in item_agents:
168             self.schedule.add(i)
169         if self.verbose:
170             print(f"␣␣␣␣-␣Items␣added")
171             print("Finished␣model␣initialization!")
172 
173         # Data collection setup
174         self.datacollector = mesa.DataCollector(
175             agent_reporters={
176                 "agent_type": lambda a: get_agent_type(a),
177                 "vector": lambda a: get_vector(a),
178                 "user_books_consumed": lambda a: get_books_consumed(a),
179                 "item_n_read": lambda a: get_item_n_read(a),
180                 "item_n_reviews": lambda a: get_item_n_reviews(a),
181                 "item_mean_rating": lambda a: get_item_mean_rating(a)
182             }
183         )
184 
185         # Create results folder
186         self.results = Results()
187         if not initial_store_path:
188             self.results.create_new_directory(run_type=run_type, verbose=self.verbose)
189             self.csv_filepaths.extend(
190                 self.results.store(
191                     prefix="initial",
192                     data=[("interactions", df), ("items", df_items), ("users", df_users
                            )],
193                     verbose=self.verbose
194                 )
195             )
196         else:
197             self.results.path = initial_store_path
198 
199     def step(self) -> None:
200         """
201         Advance model by one step
202         """
203         self.schedule.step()
204         self.datacollector.collect(self)
205 
206     def create_user(self, user_row: pd.Series) -> UserAgent:
207         """
208         Create user agent
209         """
210         return UserAgent(user_row, self)
211 
```

```
212     def create_item(self, item_row: pd.Series) -> ItemAgent:
213         """
214         Create item agent
215         """
216         return ItemAgent(item_row, self)
217 
218     def run_model(self) -> None:
219         """
220         Run model simulation
221         """
222         for i in range(self.steps):
223             self.step()
224             print(f"Step␣{i␣+␣1}/{self.steps}␣executed.", end="\r")
225         self.csv_filepaths.extend(
226             self.results.store(
227                 prefix="run", data=[("raw", self.get_raw_df())], verbose=self.verbose
228             )
229         )
230 
231     def get_raw_df(self) -> pd.DataFrame:
232         """
233         Get raw dataframe with results
234         """
235         df_raw = self.datacollector.get_agent_vars_dataframe().copy()
236         return df_raw
237 
238     def get_processed_df(self) -> pd.DataFrame:
239         """
240         Get processed dataframe with results
241         """
242         df_vectors = self.datacollector.get_agent_vars_dataframe().copy()
243         df_vectors.dropna(inplace=True)
244 
245         # Convert vector to single dimension pandas series
246         df_vectors["vector"] = df_vectors["vector"].apply(lambda x: x[0])
247         df_vectors_wide = df_vectors["vector"].apply(pd.Series)
248         df_vectors_wide.columns = get_categories()
249         return df_vectors_wide
```

B.2. agents.py

```
1 from __future__ import annotations
2 import mesa
3 import pandas as pd
4 import numpy as np
5 import random
6 from sklearn.metrics.pairwise import cosine_similarity
7 from utils import unit_normalize_vector
8 
9 class ItemAgent(mesa.Agent):
10     """
11     Item agent model
12     """
13 
14     def __init__(
15         self,
16         item_row: pd.Series,
17         model: mesa.Model
18     ) -> None:
19         """
20         Create a new item agent instance
```

```
21
22          Args:
23              item_row: pandas series row from interactions dataframe with columns:
24                  unique_id: unique ID of item
25                  is_read: number of users that have read the book
26                  rating: average rating
27                  is_reviewed: number of users that have reviewed the book
28                  priority: hidden priority of item
29                  vector: category vector as numpy array
30          """
31          super().__init__(unique_id=item_row["unique_id"], model=model)
32          self.book_id = item_row.name
33          self.n_read = item_row["is_read"]
34          self.mean_rating = item_row["rating"]
35          self.n_reviews = item_row["is_reviewed"]
36          self.priority = item_row["priority"]
37          self.vector = item_row["vector"]
38
39      def normalize_vector(self) -> np.array:
40          """
41          Normalize category vector between 0 and 1
42          """
43          return (self.vector - self.vector.min()) / (self.vector.max() - self.vector.min
                ())
44
45      def update(self, review: float | None = None) -> None:
46          """
47          Update item after interaction with user
48          """
49          self.n_read += 1
50          self.n_reviews += 1 if review else 0
51          self.mean_rating += (review / self.n_reviews) if review else 0
52
53
54  class UserAgent(mesa.Agent):
55      """
56      User agent model
57      """
58
59      def __init__(
60          self,
61          user_row: pd.DataFrame,
62          model: mesa.Model
63      ) -> None:
64          """
65          Create a new user agent instance
66
67          Args:
68              user_row: pandas series row from interactions dataframe with columns:
69                  unique_id: unique ID of item
70                  is_read: number of books read by user
71                  rating: average rating given by user
72                  is_reviewed: number of books reviewed by user
73                  book_id: list of book IDs interacted by user
74                  vector: category vector as numpy array
75                  rec_proba: probability of getting a recommendation
76          """
77          super().__init__(unique_id=user_row["unique_id"], model=model)
78          self.user_id = user_row.name
79          self.books = user_row["book_id"]
80          self.n_reviews = user_row["is_reviewed"]
81          self.mean_rating = user_row["rating"]
```

```
82          self.n_books = user_row["is_read"]
83          self.vector = user_row["vector"]
84          self.read_proba = user_row["read_proba"]
85          self.ignorant = user_row["ignorant"]
86          self.similarities = user_row["similarities"]
87          self.should_update_similarities = False
88          self.books_consumed = []
89          self.following = user_row["following"] or []
90
91      def get_read_probability(self) -> float:
92          """
93          Calculate read probability as proportion of read books from total interacted
94          """
95          return self.read_proba
96
97      def get_review_probability(self) -> float:
98          """
99          Calculate review probability as proportion of reviewed books from total read
100         """
101         return self.n_reviews / self.n_books if self.n_books else 0
102
103     def calculate_cosine_similarity(self, agent_b: UserAgent | ItemAgent) -> np.ndarray
            :
104         """
105         Calculate cosine similarity between user's own vector and other agent's vector
                (user or item)
106         """
107         return cosine_similarity(self.vector, agent_b.vector)
108
109     def find_most_similar_agent(self) -> UserAgent | None:
110         """
111         Find most similar agent by comparing cosine similarity between vectors
112         """
113         max_similarity = -1
114         most_similar_agent = None
115         for other_agent in self.model.get_agents_of_type(UserAgent):
116             if other_agent != self:
117                 similarity = self.calculate_cosine_similarity(other_agent)
118                 if similarity > max_similarity:
119                     max_similarity = similarity
120                     most_similar_agent = other_agent
121         return most_similar_agent
122
123     def get_recommendations(self, n: int) -> dict:
124         """
125         Get item recommendations as dict with item ID and probability
126
127         Args:
128             n: number of recommendations
129         """
130         alpha = 2
131         recs = {}
132         social_influence_books = self.get_social_influence_books()
133         top_books = self.get_top_books(n - len(social_influence_books))
134         rec_list = social_influence_books[:]
135         rec_list.extend(book for book in top_books if book not in rec_list)
136         for idx, rec in enumerate(rec_list):
137             prob = (alpha - 1) * (alpha ** (-idx - 1))
138             recs.update({rec: prob})
139         return recs
140
141     def get_social_influence_books(self) -> list[int]:
```

```
142         """
143         Get list of books from social influencers
144         """
145         if not self.model.social_influence:
146             return []
147         rec_list = []
148         for user_id in self.following:
149             agent = [agent for agent in self.model.get_agents_of_type(UserAgent) if
                    agent.user_id == user_id]
150             rec_list.extend(book for book in agent[0].books_consumed if book not in
                    rec_list)
151         return rec_list
152 
153     def pick_choice(self, recs: dict) -> ItemAgent | None:
154         """
155         Pick choice from dictionary of books and probabilities based on random choice
                from probabilities
156 
157         Args:
158             recs: dictionary of recommendations
159         """
160         recs = {k: v for k, v in recs.items() if k not in self.books and k not in self.
                books_consumed}
161         if not recs:  # there could be a slight chance that the agent can't get more
                recommendations
162             return
163         books = list(recs.keys())
164         probabilities = list(recs.values())
165         choice = random.choices(books, weights=probabilities, k=1)[0]
166         item = [i for i in self.model.get_agents_of_type(ItemAgent) if i.book_id ==
                choice]
167         return item[0]
168 
169     def get_top_books(self, n_books) -> list[int]:
170         """
171         Get the top books by vector similarity with agent or all items
172 
173         Args:
174             n_books: number of books to return
175             all: boolean if the books should be compared to agent (False) or all items
                    (True)
176         """
177         if self.model.rec_engine == "content-based":
178             if not self.should_update_similarities:
179                 return list(self.similarities.keys())
180             return self.update_similarities(n_books)
181         elif self.model.rec_engine == "collaborative-filtering":
182             most_similar_agent = self.find_most_similar_agent()
183             items = most_similar_agent.books
184             sorted_items = sorted(items.items(), key=lambda x: x[1], reverse=True)
185             return [item[0] for item in sorted_items[:n_books]]
186 
187     def update_similarities(self, n_books) -> list[int]:
188         """
189         Updates cosine similarities for user given all items and returns top n
190 
191         Args:
192             n: number of items to return
193         """
194         items = self.model.get_agents_of_type(ItemAgent)
195         items_matrix = np.array([book.vector.flatten() for book in items])
196         items_matrix = np.array([unit_normalize_vector(v) for v in items_matrix])
```

```
197         vector = unit_normalize_vector(self.vector.flatten())
198         similarities = np.dot(items_matrix, vector)
199         results = {
200             item.book_id: 1.0 if item.priority == 1.0 and self.ignorant == True else
                round(cosine_sim, 4)
201             for item, cosine_sim in zip(items, similarities)
202         }
203         sorted_items = sorted(results.items(), key=lambda x: x[1], reverse=True)
204         self.similarities = dict(sorted_items[:n_books])
205         self.should_update_similarities = False
206         return [item[0] for item in sorted_items[:n_books]]
207 
208     def update(self, item: ItemAgent) -> float:
209         """
210         Update user agent after interaction with item
211 
212         Args:
213             item: item agent interacted with
214         """
215         item_vector = np.where(item.vector > 0, 1, 0)
216         self.vector += item_vector
217         similarity = self.calculate_cosine_similarity(item)
218         self.books.update({item.book_id: similarity[0][0]})
219         self.books_consumed.append(item.book_id)
220         self.should_update_similarities = True
221         return similarity[0][0]
222 
223     def step(self) -> None:
224         """
225         Single step of user agent
226         """
227 
228         # Should agent get recommendations?
229 
230         if random.random() < self.get_read_probability():
231             recs = self.get_recommendations(self.model.n_recs)
232             book = self.pick_choice(recs)
233             if not book:
234                 return
235 
236             similarity = self.update(book)
237 
238             # Should agent review book?
239 
240             if random.random() < self.get_review_probability():
241                 review = round(similarity * 5) / 5
242             else:
243                 review = None
244             book.update(review)
```