

# Exploring Children's Exposure to Stereotypes via Recommender Systems

**Msc Thesis Computer Science & Engineering**

Murtadha Al Nahadi



# Exploring Children's Exposure to Stereotypes via Recommender Systems

Msc Thesis Computer Science & Engineering

Thesis report

by

Murtadha Al Nahadi

to obtain the degree of Master of Science  
at the Delft University of Technology  
to be defended publicly on Tuesday, November 14, 2023, at 15:00 PM

*Thesis committee:*

Chair: Dr. Sole Pera  
Supervisor: Dr. Sole Pera  
Daily Supervisor: Dr. Sole Pera  
External examiner: Dr. Luciano Cavalcante Siebert  
Place: Faculty of Electrical Engineering, Mathematics, Computer Science, Delft  
Project Duration: March 31, 2023 - November 14, 2023  
Student number: 4584260

An electronic version of this thesis is available at <http://repository.tudelft.nl/>.



Copyright © Murtadha Al Nahadi, 2023  
All rights reserved.



# Abstract

*“Girls are bad at math, but are better than boys in linguistic subjects!”* Such stereotypes, potentially manifesting through various sources, can impact children’s development and negatively affect their academic performance. In this work, we study the presence of stereotypes among the Top-10 suggestions of Recommender Algorithms. In particular, we conduct an empirical exploration using an extensive suite of Recommender Algorithms on two well-known datasets from different domains: *MovieLens-1M* (movies) and *Goodreads* (books). We aim to assess the presence of *gender*, *race*, and *religion* stereotypes. We utilize three stereotype detection models, based on machine learning and Large Language models, and we leverage performance metrics to contextualize the Recommender Algorithms and stereotype prominence metrics to measure the extent of their presence in the recommendations. Outcomes from this work evidence that stereotypes are not equally prominent across all Recommender Algorithms, with certain content-based and deep-learning models showing higher tendencies to recommend stereotypical content to children. Findings emerging from our exploration result in several implications for researchers and practitioners to consider when designing and deploying Recommender Algorithms, especially when children are also interacting with these systems. Furthermore, this work presents a blueprint in which stereotype detection can be expanded to other domains, other types of stereotypes, and other demographic user groups.

# Preface

I would like to express my deepest gratitude to my Professor and supervisor, Miss Pera, for her guidance and advice throughout my thesis journey. Her support had a great impact on the shape of this work and provided clarity and direction during the most challenging periods of my research.

I would also like to thank Alisa Riegers for offering her valuable perspectives during the First Stage Review and Greenlight Process as someone not connected to this work.

I am very thankful to the authors of BiasMeter, who responded to my request and pointed me to the missing data necessary for using their work. Their assistance enabled the continuation and success of my research.

To my family and friends, thank you for supporting me through this journey and I apologize for the constant complaining you had to hear for these past 7 months.

# Contents

<b>List of Figures</b>	<b>v</b>
<b>List of Tables</b>	<b>vii</b>
<b>I Preliminary Analysis</b>	<b>1</b>
<b>1 Introduction</b>	<b>2</b>
<b>2 Background and Related Work</b>	<b>5</b>
2.1 Background . . . . .	5
2.2 Related Work . . . . .	7
<b>II Implementation</b>	<b>10</b>
<b>3 Methodology</b>	<b>11</b>
3.1 Datasets . . . . .	11
3.2 Recommender Algorithms . . . . .	12
3.3 Stereotype Detection Models . . . . .	14
3.4 Metrics . . . . .	17
3.5 Experimental Setup . . . . .	19
<b>4 Ethical Considerations</b>	<b>23</b>
4.1 Data Management . . . . .	23
4.2 Ethics . . . . .	23
<b>5 Results</b>	<b>24</b>
5.1 Experiment 1: Performance of RAS . . . . .	24
5.2 Experiment 2: Exploring Stereotype presence with SDMs . . . . .	29
5.3 Discussion . . . . .	50
<b>III Closure</b>	<b>53</b>
<b>6 Conclusion</b>	<b>54</b>
<b>Bibliography</b>	<b>56</b>
<b>References</b>	<b>63</b>
<b>A Supplementary Results and Figures</b>	<b>64</b>
A.1 Hyperparameter Optimization . . . . .	64
A.2 Supplementary Results RA Performance . . . . .	68
A.3 NGIM Stereotype Prominence Results . . . . .	72
A.4 BiasMeter Stereotype Prominence Results . . . . .	73
A.5 ChatGPT 3.5 Stereotype Prominence Results . . . . .	76

# List of Figures

3.1	Heaven to Betsy, a children’s book, as seen on the Goodreads website. . . . .	14
3.2	Output showing Gender Guesser’s assignment of genders based on the first name of the characters extracted from Heaven to Betsy’s description. . . . .	15
3.3	Results of BiasMeter masking words it detected in a sentence from Heaven to Betsy’s description. It shows that BERT has a higher probability of using female terms than male terms. The subgroup word denotes what type of word is masked, in the case of gender, either man or woman. . . . .	16
3.4	Response of ChatGPT when asked for assistance with stereotype detection. . . . .	16
3.5	ChatGPT’s response detecting stereotypes in the description of the book Heaven to Betsy. . . . .	17
3.6	The proposed workflow of our empirical exploration shows how we apply Elliot and the SDMs to perform our analysis. . . . .	19
3.7	Mean and std of the nDCG for all RAS on $ML_{Ch,S}$ and $GR_{Ch}$ . The black stripes on top of the bar showcase the std, the longer the line the larger the std. . . . .	21
5.1	Mean of the performance metrics and paired t-test results for all RAS on the $ML_S$ dataset. Shades of colors indicate a category from Section 3.2. . . . .	26
5.2	Mean of the performance metrics and paired t-test results for all RAS on the $GR_{Ch}$ dataset. Shades of colors indicate a category from Section 3.2. . . . .	28
5.3	$HIT_{BAD,Gender}$ for all SDMs on $ML_{Ch,S}$ . . . . .	31
5.4	$HIT_{BAD,Gender}$ for all SDMs on $GR_{Ch}$ . . . . .	32
5.5	$MRR_{BAD,Gender}$ for all SDMs on $ML_{Ch,S}$ . The white dots indicate the average score. . . . .	34
5.6	$MRR_{BAD,Gender}$ for all SDMs on $GR_{Ch}$ . The white dots indicate the average score. . . . .	35
5.7	$REC-ST_{Gender}$ for all SDMs on $ML_{Ch,S}$ . The white dots indicate the average score. . . . .	37
5.8	$REC-ST_{Gender}$ for all SDMs on $GR_{Ch}$ . The white dots indicate the average score. . . . .	38
5.9	$HIT_{BAD,Race}$ for BiasMeter and ChatGPT 3.5 on $ML_{Ch,S}$ . The white dots indicate the average score. . . . .	39
5.10	$HIT_{BAD,Race}$ for BiasMeter and ChatGPT 3.5 on $GR_{Ch}$ . The white dots indicate the average score. . . . .	40
5.11	$MRR_{BAD,Race}$ for BiasMeter and ChatGPT 3.5 on $ML_{Ch,S}$ . The white dots indicate the average score. . . . .	41
5.12	$MRR_{BAD,Race}$ for BiasMeter and ChatGPT 3.5 on $GR_{Ch}$ . The white dots indicate the average score. . . . .	42
5.13	$REC-ST_{Race}$ for BiasMeter and ChatGPT 3.5 on $ML_{Ch,S}$ . The white dots indicate the average score. . . . .	43
5.14	$REC-ST_{Race}$ for BiasMeter and ChatGPT 3.5 on $GR_{Ch}$ . The white dots indicate the average score. . . . .	44
5.15	$HIT_{BAD,Religion}$ for BiasMeter and ChatGPT 3.5 on $ML_{Ch,S}$ . The white dots indicate the average score. . . . .	45
5.16	$HIT_{BAD,Religion}$ for BiasMeter and ChatGPT 3.5 on $GR_{Ch}$ . The white dots indicate the average score. . . . .	46
5.17	$MRR_{BAD,Religion}$ for BiasMeter and ChatGPT 3.5 on $ML_{Ch,S}$ . The white dots indicate the average score. . . . .	47
5.18	$MRR_{BAD,Religion}$ for BiasMeter and ChatGPT 3.5 on $GR_{Ch}$ . The white dots indicate the average score. . . . .	48
5.19	$REC-ST_{Religion}$ for BiasMeter and ChatGPT 3.5 on $ML_{Ch,S}$ . The white dots indicate the average score. . . . .	49
5.20	$REC-ST_{Religion}$ for BiasMeter and ChatGPT 3.5 on $GR_{Ch}$ . The white dots indicate the average score. . . . .	50

---

A.1 Paired t-test for  $ML_{\bar{S}}$  with  $p < 0.05$ . The colors with blue tints mean statistically significant and colors with red tints mean statistically non-significant. . . . . 69

A.2 Paired t-test for  $ML_S$  with  $p < 0.05$  and Bonferroni correction. The colors with blue tints mean statistically significant and colors with red tints mean statistically non-significant. . . . 71

# List of Tables

3.1	Characteristics of the datasets considered in our empirical exploration. . . . .	11
3.2	Details on datasets before and after pre-processing. . . . .	20
5.1	Results for the RAS performance on $ML_{Ch,S}$ . The best model is highlighted in <b>bold</b> and the second best model is highlighted with an <u>underline</u> . The best model per RA type based on nDCG is highlighted in <i>italic</i> . . . . .	27
5.2	Results for the RAS performance on $GR_{Ch}$ . The best model is highlighted in <b>bold</b> and the second best model is highlighted with an <u>underline</u> . The best model per RA type based on nDCG is highlighted in <i>italic</i> . <i>Note that '*' indicates no results due to unavailability of side information; '**' indicates that the model was excluded because it was too slow on the <math>GR_{Ch}</math> dataset</i> . . . . .	29
5.3	Amount of stereotype labels assigned based on each SDM to items of $ML_{Ch,S}$ (3880 movies) and $GR_{Ch}$ (14935 books). . . . .	30
A.1	Hyperparameter ranges used for RAs on both ML and Goodreads . . . . .	65
A.2	Hyperparameters of RAs used on ML and GR . . . . .	67
A.3	Results for the RAS performance on $ML_{\bar{S}}$ . The best model is highlighted in <b>bold</b> and the second best model is highlighted with an <u>underline</u> . The best model per RA type based on nDCG is highlighted in <i>italic</i> . Statistical significance for these results can be found in Figure A.1 . . . . .	68
A.4	Results for the RAS performance on $ML_S$ . The best model is highlighted in <b>bold</b> and the second best model is highlighted with an <u>underline</u> . The best model per RA type based on nDCG is highlighted in <i>italic</i> . Statistical significance for these results can be found in Figure A.2 . . . . .	70
A.5	All stereotype prominence metrics for <b>gender</b> stereotypes in $ML_{Ch,S}$ based on <b>NGIM</b> . . . . .	72
A.6	All stereotype prominence metrics for <b>gender</b> stereotypes in $GR_{Ch}$ based on <b>NGIM</b> . . . . .	72
A.7	All stereotype prominence metrics for <b>gender</b> stereotypes in $ML_{Ch,S}$ based on <b>BiasMeter</b> . . . . .	73
A.8	All stereotype prominence metrics for <b>gender</b> stereotypes in $GR_{Ch}$ based on <b>BiasMeter</b> . . . . .	73
A.9	All stereotype prominence metrics for <b>race</b> stereotypes in $ML_{Ch,S}$ based on <b>BiasMeter</b> . . . . .	74
A.10	All stereotype prominence metrics for <b>race</b> stereotypes in $GR_{Ch}$ based on <b>BiasMeter</b> . . . . .	74
A.11	All stereotype prominence metrics for <b>religion</b> stereotypes in $ML_{Ch,S}$ based on <b>BiasMeter</b> . . . . .	75
A.12	All stereotype prominence metrics for <b>religion</b> stereotypes in $GR_{Ch}$ based on <b>BiasMeter</b> . . . . .	75
A.13	All stereotype prominence metrics for <b>gender</b> stereotypes in $ML_{Ch,S}$ based on <b>ChatGPT 3.5</b> . . . . .	76
A.14	All stereotype prominence metrics for <b>gender</b> stereotypes in $GR_{Ch}$ based on <b>ChatGPT 3.5</b> . . . . .	76
A.15	All stereotype prominence metrics for <b>race</b> stereotypes in $ML_{Ch,S}$ based on <b>ChatGPT 3.5</b> . . . . .	77
A.16	All stereotype prominence metrics for <b>race</b> stereotypes in $GR_{Ch}$ based on <b>ChatGPT 3.5</b> . . . . .	77
A.17	All stereotype prominence metrics for <b>religion</b> stereotypes in $ML_{Ch,S}$ based on <b>ChatGPT 3.5</b> . . . . .	78
A.18	All stereotype prominence metrics for <b>religion</b> stereotypes in $GR_{Ch}$ based on <b>ChatGPT 3.5</b> . . . . .	78

# Part I

## Preliminary Analysis

# Introduction

In a world where every click, view, and preference is thoroughly tracked and analyzed, recommender algorithms (RAs) have an unseen influence in shaping our views, preferences, and most importantly, our perceptions [36, 84]. This is because RAs are designed to offer items that best align with users' interests for purchase or consumption [27]. They are prominent in various domains such as suggesting products in e-commerce (e.g., Amazon [87]), movies and music in streaming platforms (e.g., Netflix [35] and Spotify [62]), new connections and posts in social networks (e.g., Twitter [40] and Facebook [6]), and many more. RAs can also benefit numerous platforms in terms of user engagement and potential revenue generation. For instance, about 80% of the hours streamed on Netflix are due to recommendations, and Netflix values the impact of these recommendations over \$1 billion annually [35]. Furthermore, 30% of pageviews on Amazon are due to recommendations, emphasizing their influence on user browsing behavior and engagement with the platform [87]. Given the extensive reach and impact of RAs, examining their impact on various user groups, especially for vulnerable groups such as children is important.

RAs are driven by interaction and behavior, such as clicks, reviews, and ratings, from dominant user groups [79]. With adults being the traditional users for a lot of RAs, it becomes uncertain if the content suggested to children is appropriate [21]. The unpredictable and undesirable behavior of RAs is an ethical and social concern because a lot of content online can harm children such as adult content, online bullying, meeting strangers, self-harm sites and more [55]. For example, on the world's second biggest website, YouTube<sup>1</sup>, toddlers have a 3.5% chance of being exposed to inappropriate content within 10 recommendations [68]. Another example is when RAs are used for educational purposes where every unreliable, unreadable, or irrelevant recommendation, will harm the child [64]. While RAs are often algorithmically correct<sup>2</sup>, these algorithms are not guaranteed to be ethically correct when, for example, recommendations are not age-appropriate for children [90].

Among many potential harms online, one that can negatively affect children is stereotypes [5, 9]. Stereotypes are shared beliefs that link groups with certain traits or characteristics, held by a particular society or culture and transcending beliefs within an individual, i.e., they group unique individuals as identical copies [59]. Studies have shown that early exposure to gender stereotypes negatively affects children's ideas of gender roles, creativity, confidence skills, and accomplishments [75]. Stereotypes can lower children's self-esteem and limit their opportunities [17, 19, 59]. Stereotypes are also a risk for children because they can lead to unfair judgment and biased behavior like discrimination [11].

Given that RAs can influence perceptions and preferences [36, 84], and considering the potential harms of stereotypes on children [9, 59, 75], an important question arises: Could RAs expose children to stereotypes via their suggestions?

Research on this concern is still in its early stages, with only two studies addressing the issue of stereotypes in relation to children. The first is a position paper by Raj et al. [75] to attract researchers to the subject of stereotypes being suggested by RAs and the second is a preliminary analysis presented by Raj and Ekstrand [74] that shows that there are gender stereotypes present in query suggestions in

---

<sup>1</sup><https://www.similarweb.com/top-websites/>

<sup>2</sup>Algorithmically correct means that the RA works as intended by its design, i.e., correctly recommending items that the user prefers.



Amazon’s recommender systems.<sup>3</sup>

Other research that is closely related to stereotypes is bias and fairness in RAs [15, 27, 58]. For example, Ekstrand et al. [27] looks at popularity and demographic biases in recommender evaluation and effectiveness finding significant utility differences for different user demographic groups (based on age or gender). A prime example of fairness in RAs is when content from some creators is more likely to be recommended than other creators [26]. This is known as “popularity bias”, where RAs are more likely to recommend content that is already popular, i.e., items with the most interactions. This bias makes it harder for less popular content to gain visibility. Consequently, less popular creators may not be as able to gain recognition or commercial returns for their work. Notably, these works do not consider children as their main focus, but sometimes they are a subgroup, as is the case in the work of Ekstrand et al. [27].

Noticing the lack of research on the impact of stereotype exposure towards children via RAs we conduct an empirical analysis where we look at various types of RAs and different stereotypes and analyze whether stereotypes are being exposed to children. We measure the performance of RAs based on Top-N evaluation where an RA is evaluated in terms of its ability to recommend withheld items. This is widely regarded as the preferred setting since this simulates the end goal of a recommender system where users are presented with a list the users will like [27].

We use multiple performance metrics suitable for Top-N evaluation (see Chapter 3) to contextualize the various RAs we consider in this work and how they possibly propagate any stereotypes to children via their suggestions.

To detect the stereotype prominence in these Top-N suggestions, we employ three distinct metrics, giving us different perspectives on the stereotype presence in these lists. Among these is the novel REC-ST metric, an adaptation of the SERP-MS metric [44], traditionally used for quantifying misinformation in search engine results. To the best of our knowledge, our work is the first to adapt this metric for capturing stereotype presence in Top-N recommendation lists.

We conduct our empirical analysis with the following research objective in mind:

#### Research Objective

Explore stereotype presence in the Top-10 recommendations produced by RAs targeting children.

To control scope, we focus on three different stereotypes: *gender*, *race*, and *religion* stereotypes, as these can have a significant impact on the lives of children [13, 59, 60]. We look at the Top-10 results following in the footsteps of previous research of RA evaluations [3] and we do not consider more items since children seldom look past the Top-6 items of ranked lists [1]. With our work, we will address the following research question:

#### Research Question

To what extent are stereotypes related to gender, race, and religion present in the Top-10 suggestions made to children by RAs?

We conduct an empirical exploration to advance knowledge in the research field of recommenders and better understand the prominence of stereotypes in RA suggestions made to children. We analyze the presence of stereotypes in suggestions of RAs and their trade-offs in terms of performance and stereotype suggestions.

We use two datasets, MovieLens (ML) [41] and Goodreads (GR) [91, 92]. Both datasets contain a subset related to children and we apply them to an extensive suite of RAs from different categories. The wide range of RAs allows us to contextualize the RAs and gives us insights as to how different RAs possibly propagate stereotypes towards children. Furthermore, we analyze these suggestions using different

<sup>3</sup>When we refer to recommender systems, we mean the entire framework consisting of RAs, the graphical user interface, and other necessary components to make real-time recommendations.

state-of-the-art stereotype detection models (SDMs). These models are capable of detecting stereotypes related to gender, race, and religion. By employing these SDMs, we aim to gain insights from different perspectives regarding stereotype prominence in the suggestions of RAs. The SDMs are based on natural language processing (NLP) techniques, and since there is no common consensus on approaches for stereotype detection in the field of NLP (see Chapter 2), we consider multiple strategies. We consider a naive lexicon-based approach, a state-of-the-art probabilistic SDM known as BiasMeter [33], and we utilize ChatGPT, a state-of-the-art Large Language Model (LLM), as an SDM.

Our research stands at the intersection of RAs and the potential exposure of children to known stereotypes through these algorithms. The novelty of our work lies in addressing a significant research gap and introducing an innovative approach for detecting stereotypes in RA suggestions made to children. The key contributions of this study are:

#### Research Contribution

- Exploration of a wide variety of RAs that use different approaches and therefore produce different recommendation lists, possibly giving us insights into which RAs are more prone to stereotype propagation compared to others.
- We present REC-ST, a novel adaptation of the SERP-MS metric, traditionally used for quantifying misinformation in Top-N suggestions of SERPs. To the best of our knowledge, our work is the first to adapt this metric to capture stereotype presence in Top-N recommendation lists.
- Insights into the extent of stereotype exposure regarding gender, race, and religion stereotypes towards children from RA suggestions.
- A foundation for empirically exploring the presence of stereotypes in recommender system suggestions, which could be expanded to other domains and other stereotypes.

This work is important because if stereotypes are indeed present in the suggestions of RAs it opens the door for future research to mitigate these stereotypes and keep the children safe from harm caused by any stereotypes that RAs suggest.

In the rest of this manuscript, we first discuss related literature informing our work (Chapter 2). Then, we describe our methodology, including the datasets, RAs, SDMs, and metrics, which we use in the empirical explorations we conducted to answer our research question (Chapter 3). Furthermore, we present the ethical considerations for our work (Chapter 4). This is followed by an in-depth analysis and discussion of the produced results (Chapter 5). Lastly, we present the concluding remarks, limitations, and directions for future work (Chapter 6).

# Background and Related Work

In this chapter, we present background and related literature informing our work.

## 2.1. Background

We present background information regarding RAs and their approaches with a particular emphasis on the ones considered in this work.

### 2.1.1. Recommender Algorithms

RAs are algorithms that suggest items matching the interests of specific users [79]. Users have a lot of choices when it comes to e-commerce websites such as buying items on Amazon.com or watching movies on Netflix. While choice implies freedom, too much of it may become burdensome and can cause that freedom to be perceived as a kind of misery-inducing tyranny [83]. To help users overcome this overwhelming feeling of too many items, RAs are used to assist users with decision-making [79]. For example, a typical Netflix user loses interest after 60 to 90 seconds of choosing what movie to watch, and if no choice is made, Netflix risks the user abandoning their service [35]. Therefore, companies need to keep these users by assisting them with accurate recommendations to best match their interests.

“Item” is a general term to denote what an RA recommends to users. RAs usually focus on one specific type of item, such as movies, music, or news articles. Typically, the design, recommendation technique, and user interface are all tailored to provide effective and useful recommendations for that specific item to users.

RAs can be categorized as either non-personalized or personalized. Non-personalized RAs are useful when there is insufficient information regarding a target user’s preference. A primary example is the Most Popular algorithm, which suggests the most popular items to users based on the total number of interactions with these items. However, the primary focus of research is on personalized RAs.

In their simplest version, personalized RAs provide recommendations to users as ranked lists of items. RAs try to tailor these lists to the users’ preferences as much as possible. To generate these lists, RAs draw information from the behavior of users, which is primarily derived from two categories of information:

- **Explicit feedback:** This is information that the users provide for items they interacted with such as ratings, likes, and reviews.
- **Implicit feedback:** This can be information drawn from the user’s interaction with the system, such as clicks, page views, purchase history, etc.

Explicit feedback is seen as more reliable than implicit feedback because it comes directly from the users and it provides a user’s level of preference (e.g., a numerical rating ranging between 1-5). However, many users do not bother with rating items, and therefore explicit feedback is often unavailable or very sparse [79]. Implicit feedback allows for the user’s preference to be inferred from their interactions with the items on the system. Interactions with these items such as clicking them, adding them to the basket, etc. can be seen as some form of implicit user preference. This type of feedback is considered as a unary rating which does not indicate a specific level of preference like explicit feedback does.

### 2.1.2. Recommender Algorithm Approaches

As categorized in [79], there are six general recommendation approaches: content-based (CB), collaborative-filtering (CF), community-based, demographic, knowledge-based, and hybrid recommenders.<sup>1</sup> Furthermore, Ricci et al. [79] cover special types of RAs, representing the state-of-the-art recommenders that are emerging for specific use cases.

**Content-Based** The CB approach focuses on the attributes and properties of items. They analyze the historical interactions between a user,  $U$ , and an item,  $I$ , and then recommend another item  $J$ , based on the similarity between items  $I$  and  $J$ . The similarity is based on metadata of the items such as tags, descriptions, or textual content. For example, if a user likes a movie tagged with the genre “action”, the algorithm might learn to recommend to that user other movies that are tagged with the genre “action”. RAs that only use the CB approach may suffer from problems like limited content analysis and over-specialization. The former is when there is not enough content regarding the item due to privacy issues which deter users from giving their personal information, or when the data is hard to obtain from items such as music and images. The latter is when the system is too focused on the similarities between the items and as a result, the system may fail to recommend items that are different but still interesting to the user.

**Collaborative Filtering** This technique is considered the most popular and widely implemented in the field of RAs [79]. The fundamental idea of CF is that if users shared the same interest in an item in the past, they will more likely also share the same interests in the future. CF can be grouped into two categories: **neighborhood-based** and **latent factor models**. The main difference between the two categories is that neighborhood-based models use the user-item ratings stored in the system directly, and the latent factor models use the ratings to learn a predictive model.

The neighborhood-based approach can be further divided into a user-based approach and an item-based approach. In the user-based approach, the rating for an item of the target user is based on the ratings of similar users (neighbors). In the item-based approach, the rating for an item of the target user is based on that user’s ratings for similar items. For example, in a user-based approach: if  $U$  liked  $I$  and  $V$  liked both  $I$  and  $J$ , then the RA may recommend  $J$  to  $U$  because both users liked item  $I$ .

As mentioned, latent factor models, such as matrix factorization [50], utilize the user-item ratings to learn a predictive model. These models aim to uncover and learn from the underlying hidden patterns in the data. Both the users and items are represented as vectors in a shared latent factor (embedding) space. In this space, the dimensions (latent factors) represent the properties or characteristics that are not necessarily observable. The users and items are projected onto this space where these models can make predictions based on their relative positions in this space.

CF is not without its limitations. A common problem in this class is known as the “cold start” problem. When a new user or item is added to the system, it becomes a challenge for the models to make accurate recommendations to that user, because there is little or no historical data. Data sparsity is also a challenge, the neighborhood-based approach may suffer from little to no common ratings between pairs of users or items. When the data is too sparse, latent factor models such as matrix factorization may be prone to overfitting.

As **deep learning** became more popular, more and more researchers started combining deep learning models with CF RAs. In particular, matrix factorization techniques have been transformed into their corresponding deep learning solution. They are useful because deep learning models require less feature engineering saving the time of practitioners and they are useful in processing raw unstructured data such as text, images, audio, and video, which are common in many RAs.

**Community-based** In a community-based approach recommendations are based on the preferences of the user’s friends. This technique follows the saying: “Tell me who your friends are, and I will tell you who you are”. It might seem similar to CF, however, community-based RAs specifically leverage the preferences from the community surrounding a user rather than finding similarities potentially across the whole user base.

<sup>1</sup>In our work, we focus on CB, CF, and Hybrid RAs. For completeness, we give a brief overview of the well-known approaches, even if they are not part of our study.

**Demographic** Demographic RAs recommend items based on the user's demographic profile. For example, users can be routed to different pages based on their language or country. Or, suggestions may be adapted based on the user's age.

**Knowledge-based** In this approach, items are recommended based on specific domain knowledge about how item features meet users' needs and preferences. For example, a traveling agency website can recommend suitable traveling packages or destinations based on the input of the user's preferences, such as destination, budget, type of activities, etc.

**Hybrid Recommenders** To overcome the disadvantages that some of these techniques have, hybrid recommender systems emerged. They aim to use the advantage of one technique to fix the disadvantages of the other technique. For example, an item-based CF approach might be used in combination with a CB model to counter the cold-start problem. There is no historical data for the CF technique to leverage when a new item is added to the system, but the CB approach can use the item's meta-data to make recommendations.

**Special Recommendation Approaches** While the aforementioned approaches cater to broad recommendation scenarios, there are also specialized techniques tailored for unique use cases. One specific use case is when long-term user data is not available or desired, then **Session-Based Recommenders** [93] can be useful. These recommenders suggest items based on short-term anonymous user interactions during sessions in domains such as e-commerce or news platforms. Another specialization is the **Group Recommenders** [20], which caters to groups of users. The challenge here lies in combining individual users' preferences to recommend items that would appeal to the entire group, for example, group travel planning or family movie selection. **People to People Recommenders** [73] have to recommend users to other users, for example, when employers are looking for employees or in social networking and online dating apps. **Cross-Domain Recommenders** [46] are recommenders that exploit user data collected from one recommender system or domain to make suggestions in other domains. For instance, if a user shows interest in cooking books in one domain, a cross-domain recommender might suggest cooking classes or kitchenware in another domain. Furthermore, there are also **Adversarial Recommenders** [22], built to be resilient to adversarial attacks trying to manipulate the system. These systems are used to detect and counter attempts to manipulate the recommendation process, ensuring the integrity and reliability of suggestions. These specialized recommendation approaches show how adaptable and versatile recommender systems can be to cater to specific user needs in various settings.

## 2.2. Related Work

In this section we present the current research landscape of RAs with children, the different strategies currently existing in NLP for detecting stereotypes, and how our research is positioned in this landscape.

**RAs for Children** While research on RAs for adults has been well-investigated, research on RAs for children is still in its infancy [70]. The majority of the research is conducted in an educational setting [51, 64, 69, 71, 72]. Murgia et al. [64] introduces seven complex layers that need to be considered when designing RAs for children in an educational setting. To name a few layers, they discuss the complexity and importance of providing explanations for suggestions made by RAs, they also touch upon the importance of ethics, especially when children are the target users, and also discuss the challenges that come with assessing the performance of RAs in an educational setting. Pera and Ng [71] introduce BR<sub>E</sub>K12, a hybrid recommender approach that combines CB and CF to make book recommendations to "K-12"<sup>2</sup> readers based on their grade levels. They also introduce ReLAT a tool employed by BR<sub>E</sub>K12 to determine the grade levels of the books. Pera and Ng [72] introduce Rabbit, a book recommender emulating the readers' advisory service that is offered at school/public libraries. It aims to make adequate book recommendations that align with the readability levels of K-12 readers. Pera et al. [69] investigate how RAs can help and benefit children (ages 9 to 11) in the classroom. The main takeaways are that RAs have a positive impact on the completion of inquiry-related tasks. However, it seems that children did not really trust suggestions if they did not know the source of the recommendations. Kucirkova [51] discuss the importance of personalization

---

<sup>2</sup>K-12 refers to children ranging from kindergarteners to 12<sup>th</sup> grade.

in reading recommender systems for children in an educational setting and reading for pleasure where the personalization logic of reading recommender systems is critically reviewed, highlighting its (dis)alignment with Papert’s constructionist and socio-constructionist theories about learning.

Outside the educational domain, literature on RAs for children is sparse [8, 21, 25, 27, 38, 68, 89, 90]. Some of the research highlights that the current evaluation strategies and metrics for RAs may not be adequately tailored to the needs and behaviors of children. Ekstrand [25]’s position paper highlights challenges in RA evaluation for children, such as the lack of datasets and the multiple stakeholders that need to be considered. Gómez Gutiérrez et al. [38] is a recent literature review giving a more in-depth overview of such challenges and different perspectives involved in the evaluation of recommenders for children.

Ekstrand et al. [27] research fairness across different demographic groups in recommender systems. While children are not the main focus of their work, results show that there are significant differences in the evaluation of the RAs across different demographic user groups on two datasets. Deldjoo et al. [21] extend the interfaces of existing traditional recommender systems<sup>3</sup> by providing a child-friendly interaction paradigm. They provide the first results of a research-in-progress that can recognize tangible objects through image recognition and provide movie recommendations based on these objects. They want to extend this work to a scenario where children could ask for movie recommendations by showing toy objects such as a car, plane, or a doll. Papadamou et al. [68] is concerned with the amount of inappropriate content that children get recommended on YouTube. They design a classifier able to discern between inappropriate and appropriate content aimed at toddlers. Although not entirely focused on children, Tang and Winoto [90] emphasizes the importance of ethical considerations when recommendations are made, e.g., when movies are suitable for adults but not for children. Furthermore, they present a user-initiated ethical RA that filters out inappropriate content based on the user’s requests. Spear et al. [89] perform a preliminary analysis exploring the influence of RAs on children’s (aged 6 to 17) online music listening behavior. The results show that among different ages for children, there is a distinct minority with different music-listening behavior compared to the majority of the teenagers adhering to the stereotype that they listen to dark music. Coupled with the fact that most music recommenders are geared toward adults, [89] conclude that “one size fits all” recommendation strategies will not work for children. Other work such as Beyhan and Pera [8] offers a novel perspective on user modeling by examining the appeal of book covers to children. Alternative pathways are suggested for personalizing content that does not solely depend on analyzing user historical data.

While significant advancements have been made in algorithmic and user modeling for RAs with children, especially in educational contexts, there remains a notable gap in addressing potential harms. While some studies touch upon inappropriate content and ethical considerations, a thorough exploration of implications and vulnerabilities unique to children is still lacking.

**Stereotype Exploration** Stereotypes have been explored for decades by psychological researchers. However, for computer scientists in NLP, this is a relatively new topic [31]. Much of the work focuses on detecting and mitigating stereotypical bias in word embeddings or LLMs. For example, Bolukbasi et al. [10] identifies stereotypes in word embeddings by showing that word vectors such as “woman” and “homemaker” are close while the vector for “man” is close to the vector “computer engineer”. Evaluating such stereotypical bias, datasets of common stereotypes such as StereoSet [65] and CrowsPairs [66] have been introduced. However, detecting “human” stereotypes from text is still an under-explored area in the literature.

Research conducted with unsupervised NLP approaches utilizes lexicon-based sentiment analysis and statistical calculations of word co-occurrence that consider some aspects of human stereotypes. Rudinger et al. [80] use pointwise mutual information to detect stereotypical bias in a widely used NLP dataset: Stanford Natural Language Inference (SNLI). The authors are concerned with the impact of a dataset’s biases on the models and applications trained on it. Charlesworth et al. [14] used a word embedding-based unsupervised approach to explore stereotypes in language corpora (65+ million words).

The detection of stereotypes in a supervised manner has also been explored in NLP and it is often done in the context of detecting abusive behavior. While detecting abusive content expressed explicitly in

<sup>3</sup>Recommender systems refers to the entire application or service that delivers the recommendations, consisting of the RA, the graphical user interface and more.

social media posts achieves high performance, detecting subtle expressions of stereotypes and micro-aggression appears to be challenging [12]. Cryan et al. [18] experimented with lexicon-based approaches and supervised classifiers to detect gender stereotypes in text and compare the results between these approaches. Sap et al. [81] introduce SBIC, a large dataset containing abusive content from online posts annotated with the implied stereotypical meaning, and show that current generative models have a hard time detecting stereotypes present in implicit expressions of abusive content.

Other NLP studies adopt stereotype theories from the social science field to detect stereotypes in text. Joseph et al. [45] cluster tweets about racially motivated police brutality using Affect Control Theory and Semantic Relationship Theory to explain stereotypes across two dimensions. Fokkens et al. [30] extract micro-portraits from pieces of Dutch text to explore stereotypes about Muslim men in the Dutch media.

To the best of our knowledge, only one position paper and a preliminary analysis exist discussing stereotypes exploration/exposure in suggestions of RAs. Raj et al. [75] call for a need to investigate stereotypes in RAs with the learning environment as a starting point. Raj and Ekstrand [74] explore gender stereotypes in the queries and search results in the search engines of Amazon and Target. They find that e-commerce RAs frequently target gender for children's items through query suggestions and retrieved results change with the presence of gender in the queries.

**Our research** The review of the related works highlights a notable gap: the absence of comprehensive research on stereotype exposure, propagation, and mitigation in RA suggestions made to children, with only two papers touching on the subject. Our research directly addresses the gap in stereotype exposure by conducting an empirical analysis exploring gender, race, and religion stereotypes in movie and book recommendations made to children across a wide variety of RAs.

Unlike prior research in NLP, often narrowing its focus on a specific stereotype or domain, our approach is more generalized. We examine multiple stereotypes (gender, race, and religion) and utilize existing SDMs from NLP, introducing an unprecedented strategy that offers insights from multiple perspectives on stereotype exposure. Our work paves the way for future explorations into different stereotypes and different domains, such as search engines.

# Part II

## Implementation



# Methodology

In this chapter, we explain the methodology of our research. It includes the datasets, the RAs, the stereotypes, the SDMs, and the metrics we use to carry out our empirical exploration. We describe the steps we take to explore stereotypes in the suggestions of the RAs. All code needed for reproducing our results can be found in our [GitHub repository](#).

## 3.1. Datasets

In this section, we describe the two datasets used in our empirical exploration: MovieLens (ML) and Goodreads (GR). We considered these two datasets because they include data related to children and are from different domains. The summarized characteristics of the datasets can be found in Table 3.1.

### 3.1.1. MovieLens

ML [41] is a dataset that is released and maintained by GroupLens. Different ML datasets were collected over various periods of time. In this research, we use ML-1M<sup>1</sup> because it is the largest dataset from ML that still contains the user demographics such as age. The age demographic allows us to distinguish between child and adult users. ML-1M is comprised of 6.040 users, 3.883 movies, and 1.000.209 ratings. In this manuscript, whenever we mention ML, we mean the ML-1M version.

To align ML with our research objective, we use a subset of ML containing the users that are below the age of 18 to explore stereotypes in the recommendations generated by the RAs. We refer to this subset as  $ML_{Ch}$  and it comprises 222 users, 2.650 items, and 27.211 interactions for children.

### 3.1.2. Goodreads

The GR dataset [91, 92] is a large-scale collection of data scraped from the Goodreads website.<sup>2</sup> The data is collected from users' public shelves. GR includes user-book interactions, metadata about the books, and detailed book reviews from users. Similar to ML, we focus only on a subset that contains books tagged with the "children" genre. We refer to this subset as  $GR_{Ch}$ <sup>3</sup>, which is comprised of 542.145 users, 124.082 books, and 10.059.349 interactions.

Dataset	#users	#items	#interactions
<b>ML</b>	6,040	3,883	1,000,209
<b>ML<sub>Ch</sub></b>	222	2,650	27,211
<b>GR</b>	876,145	2,360,655	228,648,342
<b>GR<sub>Ch</sub></b>	542,145	124,082	10,059,349

**Table 3.1:** Characteristics of the datasets considered in our empirical exploration.

<sup>1</sup><https://grouplens.org/datasets/movielens/1m/>

<sup>2</sup><https://www.goodreads.com/>

<sup>3</sup><https://mengtingwan.github.io/data/goodreads.html#datasets>

## 3.2. Recommender Algorithms

In this section, we describe the RAs considered in our empirical exploration. Recall that there exists a wide range of RAs. In our exploration, we identify a sample that includes the categories of recommendation strategies presented in Chapter 2.<sup>4</sup> We included all the parameter configurations in Tables A.1 and A.2 of Appendix A.

### 3.2.1. Non-personalized

Non-personalized RAs are common baselines still used in recent studies analyzing RAs [3]. MostPop is still prominent in many platforms utilized when not enough information is available about the target user's preferences [79]. For example, Netflix uses MostPop to jumpstart their RA for new users.<sup>5</sup>

**MostPop** MostPop recommends the most popular item to the users. The popularity of the item is defined by the number of observed interactions in the training data.

**Random** The Random algorithm recommends items at random to users without considering any user preferences or item characteristics.

### 3.2.2. Neighborhood-based

Neighborhood-based models are a collaborative filtering technique where user-item ratings stored in the system are directly used. It follows the principle of recommending items to users with similar tastes.

**ItemkNN** ItemkNN is an item-item collaborative filtering [54] algorithm that recommends the most similar items to the ones the user has previously rated highly. It identifies the  $k$  most similar items to a target item based on their similarity scores. These similar items are then recommended to the user.

**UserkNN** UserkNN is a user-user collaborative filtering [78] algorithm that recommends items based on the preferences of similar users. It identifies the  $k$  most similar users to a target user, based on their past ratings or interactions, and suggests items highly rated by those similar users.

**AttributeItemkNN** Attribute ItemkNN [34] is similar to the regular ItemkNN algorithm, however, it differs when calculating the item-item similarities by also including additional item attributes which can help capture item characteristics and user preferences more accurately.

**AttributeUserkNN** Attribute UserkNN [34] is similar to the regular UserkNN algorithm, however, it differs when calculating the user-user similarities. It takes into account additional user attributes or demographic information which can help capture individual user preferences more accurately.

### 3.2.3. Latent Factor Models

As mentioned before in Section 2.1.2, latent factor models leverage the user-item matrix to create a predictive model by uncovering hidden patterns from user ratings data.

**BPR-MF** The Bayesian Personalized Ranking with Matrix Factorization model is based on the work of Rendle et al. [76]. It was proposed in 2009 and tailored for implicit feedback. It differs from regular MF with the objective function. While MF is trying to minimize the prediction error, BPR-MF aims to maximize the likelihood that a known positive item (user-item interaction occurred) is ranked higher than a negative item (no user-item interaction occurred).

**FunkSVD** Funk Singular Value Decomposition [32] was proposed by Simon Funk in 2006 for the Netflix Prize challenge. It is a variant of SVD where stochastic gradient descent is used to minimize the rating prediction error. FunkSVD is suitable for sparse user-item rating datasets which is commonly encountered in recommendation systems.

<sup>4</sup>Note that Elliot, the recommender framework used in this research work opts for a more nuanced categorization of the RAs. For easier reproducibility, we present the RAs in the same categorization as the Elliot framework in the rest of this manuscript.

<sup>5</sup><https://help.netflix.com/en/node/100639>

**MF** A regularized Matrix Factorization based on [50] creates two lower-rank matrices from the user-item matrix. A regularization term is integrated into the objective function to avoid overfitting. This MF model uses stochastic gradient descent to minimize the prediction error.

**MF2020** Another Matrix Factorization algorithm, but based on the work of Rendle et al. [77]. This version uses more regularization parameters compared to the MF model.

**PMF** Probabilistic Matrix Factorization as described in [63] is a variation of MF that uses a probabilistic linear model with Gaussian observation noise. The defined conditional distribution allows for a probabilistic interpretation of the user-item interactions.

**PureSVD** Pure Singular Value Decomposition based on [16] is an SVD-based implementation. Traditional SVD requires a fully observable matrix for the factorization. PureSVD solves this problem by imputing the missing values with zeroes, which is possible due to the nature of the item ranking task where the focus is not on the rating of the items.

**Slim** Sparse Linear methods [67] was proposed in 2011 as a regression-based method for top-n recommendation tasks. The ElasticNet version of Slim is used based on Levy and Jack [52] since it often leads to more competitive results.

### 3.2.4. Artificial Neural Networks

Deep learning models are neural networks with multiple hidden layers able to capture complex, non-linear relationships in the data.

**ConvMF** Convolutional Matrix Factorization [47] is a deep learning model that combines convolutional neural networks with PMF to learn the context of review documents.

**DeepFM** Deep Factorization Machines [37] combines two components, the factorization machines (FM) component to learn low-order feature interactions and the deep component, which is a multilayer perceptron (MLP), that learns high-order feature interactions.

**NeuMF** Neural Matrix Factorization [43] combines generalized matrix factorization (GMF), which utilizes a weighted similarity function, with MLP to replace the traditional inner product of MF.

### 3.2.5. Adversarial Learning

Adversarial recommenders are designed to maintain their performance even when subjected to attacks that aim to mislead or manipulate the system. Essentially, they are trained to be resilient, ensuring consistent recommendations regardless of whether they are under attack or operating normally.

**AMF** Adversarial Matrix Factorization is introduced by He et al. [42] as a solution to the lack of robustness of many recommender systems. AMF builds further upon BPR-MF which can suffer from adversarial perturbations, that can lead to worse generalization of the model. The difference with BPR-MF is that AMF adds adversarial training to also consider adversarial perturbations when minimizing the objective function.

### 3.2.6. Autoencoders

Autoencoders are neural networks designed to compress data into a latent space and then reconstruct it. Variational Autoencoders, a variant of autoencoders, represent data in this latent space as probability distributions rather than fixed points, offering a more probabilistic and robust approach to encoding [48].

**MultiVAE** MultiVAE [53] was introduced in 2018. It is an extension of variational autoencoders adapted to be used for collaborative filtering with implicit feedback.

### 3.2.7. Content-Based

Content-based RAs generate recommendations by analyzing the attributes of items and comparing them to a user’s profile, which is constructed from their past interactions. Essentially, these models suggest items that align with the user’s observed preferences.

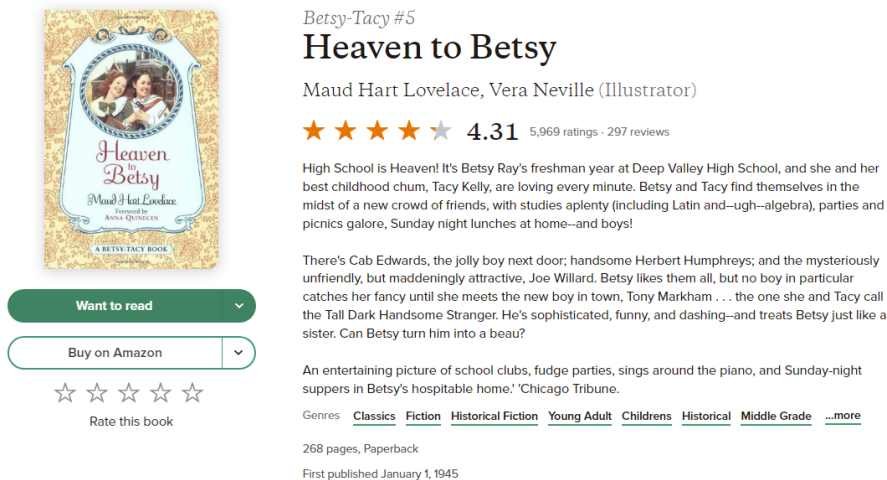
**VSM** The Vector Space Model [24] is a content-based RA. It recommends items to users by computing similarities between item attributes that a user likes. This version extends the classical VSM to include semantic information, making it suitable for dealing with RDF graphs.

## 3.3. Stereotype Detection Models

Although psychological researchers have studied stereotypes for many years, the focus on detecting stereotypes within natural language processing (NLP) research is still in its infancy [31]. Existing strategies for detecting stereotypes often focus on a specific type of stereotype and a particular context, making it difficult to generalize stereotype detection to other domains [31]. Furthermore, some of the research is done at the word level, making it nontrivial for analyzing sentences, which is the case in our work (see Chapter 2).

Recall that our research objective is the exploration of stereotypes and their presence in Top-10 suggestions of RAs. We aim to achieve this by exploring three different strategies each using different SDMs, leveraging their unique capabilities when detecting stereotypes. In this work, we narrow our focus to three types of stereotypes: *gender*, *race*, and *religion* stereotypes. While we acknowledge the existence of other types of stereotypes, we consider these since they are among the most “popular” ones and are frequently encountered in related literature (see Section 2.2) while also having significant impacts on society and potentially harmful effects on children’s lives [59, 60].

In the rest of this section, we will delve deeper into each of these strategies, giving an example of how these SDMs detect gender stereotypes in a book illustrated in Figure 3.1.



*Betsy-Tacy #5*  
**Heaven to Betsy**  
 Maud Hart Lovelace, Vera Neville (Illustrator)  
 ★★★★★ 4.31 5,969 ratings - 297 reviews

High School is Heaven! It's Betsy Ray's freshman year at Deep Valley High School, and she and her best childhood chum, Tacy Kelly, are loving every minute. Betsy and Tacy find themselves in the midst of a new crowd of friends, with studies aplenty (including Latin and—ugh—algebra), parties and picnics galore, Sunday night lunches at home—and boys!

There's Cab Edwards, the jolly boy next door; handsome Herbert Humphreys; and the mysteriously unfriendly, but maddeningly attractive, Joe Willard. Betsy likes them all, but no boy in particular catches her fancy until she meets the new boy in town, Tony Markham . . . the one she and Tacy call the Tall Dark Handsome Stranger. He's sophisticated, funny, and dashing—and treats Betsy just like a sister. Can Betsy turn him into a beau?

An entertaining picture of school clubs, fudge parties, sings around the piano, and Sunday-night suppers in Betsy's hospitable home.' *'Chicago Tribune.*

Genres [Classics](#) [Fiction](#) [Historical Fiction](#) [Young Adult](#) [Childrens](#) [Historical](#) [Middle Grade](#) [...more](#)

268 pages, Paperback  
 First published January 1, 1945

Figure 3.1: Heaven to Betsy, a children’s book, as seen on the Goodreads website.

### 3.3.1. Naive Strategy

We introduce a naive approach, which we refer to as NGIM (Name-based Gender Identification Model), to calculate the number of males or females associated with each item across both datasets. Items are deemed stereotypical if the count of “males” surpasses that of “females”. Retrieving the males and females from both datasets requires a different approach.

For ML we use Python’s IMDb library to process the movies. We query the movie title, retrieving the first five (if available) cast members from the first result. For  $GR_{Ch}$  we use Flair’s Named Entity Recognition tool to extract characters from the book descriptions. We exclude author names from the descriptions because the authors are usually not part of the characters in the books.

For both datasets, we utilize Python’s Gender Guesser library to detect the gender of the cast/characters. If the number of cast members is male-dominated we assign the label 1 (stereotypical) to the movie item, otherwise, we label it 0 (not stereotypical). Then we apply stereotype presence metrics to analyze their occurrence in the recommendation lists of the RAs.

Consider the book description in Figure 3.1, which when processed using NGIM, yields the following entity tags:

#### Entity tags of Flair’s NER tool

High School is Heaven! It’s [Betsy Ray/PER]’s freshman year at [Deep Valley High School/ORG], and she and her best childhood chum, [Tacy Kelly/PER], are loving every minute. [Betsy/PER] and [Tacy/PER] find themselves in the midst of a new crowd of friends, with studies aplenty (including [Latin/MISC] and—ugh—algebra), parties and picnics galore, Sunday night lunches at home—and boys! There’s [Cab Edwards/PER], the jolly boy next door; handsome [Herbert Humphreys/PER]; and the mysteriously unfriendly, but maddeningly attractive, [Joe Willard/PER]. [Betsy/PER] likes them all, but no boy in particular catches her fancy until she meets the new boy in town, [Tony Markham/PER] . . . the one she and [Tacy/PER] call the [Tall Dark Handsome Stranger/MISC]. He’s sophisticated, funny, and dashing—and treats [Betsy/PER] just like a sister. Can [Betsy/PER] turn him into a beau?

From the identified entities, we concentrate on those tagged as persons, using Gender Guesser to extract genders from first names. To prevent counting characters more than once, only unique names are considered. The outcomes from the gender guesser are depicted in Figure 3.2. There are instances where the tool cannot categorize gender, returning “unknown”. In such cases, we omit these results, focusing only on genders that are clearly identified. If the gender guesser suggests a name as ‘most likely male’ or ‘most likely female’, we classify these as male or female, respectively. Using this method, the results indicate the presence of three males and one female in the aforementioned book description, hinting at potential gender stereotyping.

```
Herbert Humphreys -> male, Tony Markham -> male, Joe Willard -> male, Cab Edwards -> unknown, Tacy Kelly -> unknown, Betsy Ray -> female
```

**Figure 3.2:** Output showing Gender Guesser’s assignment of genders based on the first name of the characters extracted from Heaven to Betsy’s description.

### 3.3.2. BiasMeter

BiasMeter is a tool proposed by Gaci et al. [33] to identify stereotypes in sentences or documents. It works by masking words related to a set of predefined social groups, which are gender, race, and religion. The masked sentence is then fed to a language model, in this case, BERT [23], to fill in potential words and compare the probabilities of these words being filled in. The model outputs a bias score for each social group with values above 0 being stereotypical and values below being anti-stereotypical with a range between -1 and 1. We leverage BiasMeter to explore and quantify the presence of gender, race, and religion stereotypes in the suggestions of RAs.<sup>6</sup>

We employ BiasMeter on the item descriptions of both ML and  $GR_{Ch}$ .<sup>7</sup> The item descriptions consist mostly of multiple sentences. Although originally designed to work at the sentence and document level, empirical validations on BiasMeter only demonstrate its applicability at the sentence level [33]. Therefore, we opted to tokenize the descriptions in sentences and feed them to BiasMeter. This results in multiple output labels for each item description.

To derive a conclusive label for each item description, we aggregate the output probabilities from BiasMeter using the Stanford Certainty Factor [56]. The Stanford Certainty Factor is used for aggregating different rules that lead to the same conclusion. In our case, the rules are the sentences, and the conclusions are the aggregated probabilities representing the likelihood of the stereotype labels (i.e., gender, race,

<sup>6</sup>Our exploration is exclusively focused on the presence of stereotypes, therefore not considering anti-stereotypes produced by BiasMeter.

<sup>7</sup>The descriptions of the movies originate from The Movie Database: <https://www.themoviedb.org/>.

and religion labels). The aggregated probabilities are bounded between -1 and 1, with -1 meaning the occurrence is known to be false and 1 meaning the occurrence is known to be true. We have set a threshold of 0.7 for the stereotype labels, considering all values above 0.7 as definitive stereotypes assigned the label of 1, otherwise, assigned the label of 0.<sup>8</sup> We then explore the recommendations of the RAs with the labels gained from BiasMeter to give us an insight as to how stereotypes are present with this strategy.

For illustration, Figure 3.3 displays BiasMeter’s masking technique on a sentence from Heaven to Betsy’s description. Interestingly, in the first masked sentence (2nd line in the example), the probability for “man” stands at 0.0294, whereas “woman” stands at 0.9706. This difference suggests that, in the given context, BERT has an overwhelming bias to align words denoting sophistication, humor, and charm with female attributes rather than male. The second sentence shows similar behavior, however, here it is justifiably biased toward female words since Betsy is the name of the female character and the masked word is linked to her.

```
He's sophisticated, funny, and dashing--and treats Betsy just like a sister.

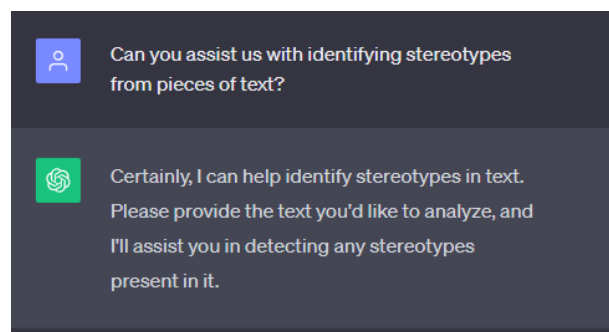
[MASK] 's sophisticated , funny , and dashing -- and treats betsy just like a sister .
{'man': 0.02938519927053183, 'woman': 0.9706148007294682} Subgroup word: man

he 's sophisticated , funny , and dashing -- and treats betsy just like a [MASK] .
{'man': 0.36930293286484556, 'woman': 0.6306970671351545} Subgroup word: woman
```

**Figure 3.3:** Results of BiasMeter masking words it detected in a sentence from Heaven to Betsy’s description. It shows that BERT has a higher probability of using female terms than male terms. The subgroup word denotes what type of word is masked, in the case of gender, either man or woman.

### 3.3.3. ChatGPT

ChatGPT is a state-of-the-art LLM developed based on the generative pre-trained transformer (GPT) architecture [57]. It can be used in a variety of ways, ranging from holding a conversation to solving specific tasks like text summarization. We utilize ChatGPT in a similar manner to a recent study [28] where it assisted with relevance judgments. ChatGPT claims that it can identify stereotypes from pieces of text, as illustrated in Figure 3.4. Therefore, we leverage ChatGPT to identify stereotypes from movie and book descriptions suggested to children.



**Figure 3.4:** Response of ChatGPT when asked for assistance with stereotype detection.

We utilized ChatGPT’s API for detecting stereotypes in the datasets. There are many models available with the API and we chose ChatGPT 3.5 as the model since this is currently the second-most advanced model of OpenAI and using GPT-4 is too expensive.<sup>9</sup> We used OpenAI’s “API Request Parallel Processor” script which handles large amounts of requests without exceeding the limits of the API.<sup>10</sup>

<sup>8</sup>We set the threshold value at 0.7 to ensure we select labels with a high level of confidence.

<sup>9</sup>For the entirety of both datasets the costs would approximately be 50\$ for GPT 3.5 and 1500\$ for GPT 4 in October 2023.

<sup>10</sup>[https://github.com/openai/openai-cookbook/blob/main/examples/api\\_request\\_parallel\\_processor.py](https://github.com/openai/openai-cookbook/blob/main/examples/api_request_parallel_processor.py)



Because the API does not have a chat-like functionality that is aware of previous interactions, we had to first provide an input prompt to ask ChatGPT to label the item description we will provide next as stereotypical or not.

#### Input Prompt

*You are an expert assessor who is tasked with identifying stereotypes in pieces of text. The texts are a few sentences long and if a stereotype is present output the stereotype label and a numbered probability estimation of its presence in that text. If it is not present, return 0. Please adhere to the following output format and do not include anything else if a stereotype is present:  
Stereotype: stereotype label, Probability: the numbered probability  
If the stereotype is not present output: 0*

For the input prompt, we draw inspiration from Faggioli et al. [28]. We kept the input prompt simple and straightforward and did not spend time optimizing the prompt (i.e., prompt engineering). We use this input prompt as the first baseline and optimizing the prompt is left for future work.

In each request, we add the item description after the input prompt and save the response of ChatGPT for further analysis. From these responses, we extracted all items that were identified as either gender, race, or religion stereotypes<sup>11</sup> and again used a threshold of 0.7 for assigning them with stereotypical label 1; otherwise, we assigned the not stereotypical label 0. Figure 3.5 shows the response from ChatGPT 3.5 when asking to detect stereotypes in the description of the book *Heaven to Betsy*. The response shows that ChatGPT 3.5 finds a gender stereotype and romantic stereotype. In our case, we would label this book as stereotypical since the probability is  $\geq 0.7$ .

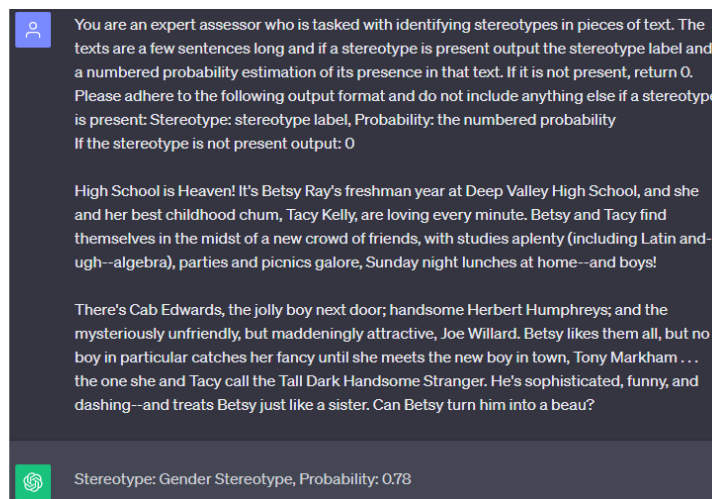


Figure 3.5: ChatGPT's response detecting stereotypes in the description of the book *Heaven to Betsy*.

## 3.4. Metrics

In this section, we describe the metrics used for contextualizing the performance of RAs and the metrics that give us insights into stereotype presence in the Top-N recommendation lists of RAs.

### 3.4.1. Performance

Given that we examine Top-N RAs, we use common assessment metrics to quantify their performance [3]. We use **HIT**, **Mean Average Precision (MAP)**, **Mean Reciprocal Rank (MRR)**, and **Normalized Discounted Cumulative Gain (nDCG)**. We employ the performance metrics on all RAs, not to compare their effectiveness but to contextualize the RAs behavior when exploring the presence of stereotypes within

<sup>11</sup>As the responses from ChatGPT 3.5 were not consistent, we extracted the stereotypes with the help of definitional words used in the work of Gaci et al. [33].

the recommendations.

**HIT** The HIT metric indicates whether a relevant item is present in the Top-N recommendation list for a user. It captures the presence of relevant items by outputting a 1 for a user if at least one item is relevant from the Top-N recommendation list and otherwise 0 if no relevant item is present. The overall HIT score for an RA is averaged over all users.

**MAP** The MAP metric assesses the ability of an RA to rank relevant items at a higher position in the Top-N recommendation list. It does this by averaging the precision scores at each position where a relevant item is found and taking the mean of these average precisions for each user. The overall MAP score averages the MAP scores of each user and it ranges between 0 and 1, with 1 indicating that all the items are relevant and 0 indicating that no item is relevant in the Top-N recommendation list.

**MRR** The MRR metric considers only the rank of the first relevant item in the Top-N recommendation list for each user. The overall MRR score is the average across all users and it ranges between 0 and 1, with 1 meaning the first relevant item is at the top, and 0 meaning the relevant item is not present in the Top-N recommendation list.

**nDCG** The nDCG metric captures the ranking quality of the items in the Top-N recommendation lists for each user. It first calculates the DCG, giving higher importance to relevant items ranked higher in the Top-N recommendation list. Then, it normalizes the DCG by the ideal DCG, where all relevant items are ranked at the top of the Top-N recommendation list. The overall nDCG score is the average across all users and it ranges between 0 and 1, with 1 meaning all relevant items are at the top, and 0 meaning no relevant items are present in the Top-N recommendation list.

### 3.4.2. Stereotype presence

For detecting stereotype presence in recommendations of RAs we use three different metrics:  $HIT_{BAD}$ ,  $MRR_{BAD}$ , and REC-ST. Each of these gives us a different perspective on the stereotype presence in the recommendation lists.

**$HIT_{BAD}$**  The  $HIT_{BAD}$  metric closely resembles the previously mentioned performance metric HIT. They differ in what is classified as a hit. While HIT returns 1 when at least one relevant item is present in the Top-N recommendation list,  $HIT_{BAD}$  returns 1 when at least one item in the Top-N recommendation list contains a stereotype. Recall that we consider stereotypes either to be gender, race, or religion stereotypes and we will denote them as  $HIT_{BAD,Gender}$ ,  $HIT_{BAD,Race}$ , and  $HIT_{BAD,Religion}$  respectively. This metric gives us insights into how often users are presented with stereotypes, helping us explore their presence in the recommendations of the RAs.

**$MRR_{BAD}$**  The  $MRR_{BAD}$  metric behaves much like the performance metric MRR, however, we are interested in the rank of the first item containing a stereotype in the Top-N recommendation list. We consider this metric because recent research of Allen et al. [1] mentioned that most of the time, children only click on the top-2 results and seldom look beyond the top-6 results when presented with a ranked list of items. This metric provides insight into where RAs place the first stereotypical item and the potential interaction of children with this item placed in the Top-N recommendation list.

For each gender, race, and religion stereotype, we will refer to  $MRR_{BAD}$  as  $MRR_{BAD,Gender}$ ,  $MRR_{BAD,Race}$ , and  $MRR_{BAD,Religion}$  respectively.

**REC-ST** We propose adapting the SERP-MS metric introduced by Hussein et al. [44]. Although the metric was originally introduced to account for the amount of misinformation and the ranking of  $N$  results in SERPs, we posit that it can be easily repurposed to quantify stereotypes in the Top-N recommendation list. We refer to the adapted SERP-MS as REC-ST (**RE**Commender **ST**ereotypes). In our case, REC-ST captures the number of stereotypes while taking into account the ranking of the item in the Top-N recommendation list (see Equation 3.1). REC-ST can provide insight into how the stereotypical items are ranked in the RAs recommendation lists, with items ranked higher being more likely to be interacted with by children.



$$REC-ST@N = \frac{\sum_{r=1}^N (x_i * (N - r + 1))}{\frac{N*(N+1)}{2}} \quad (3.1)$$

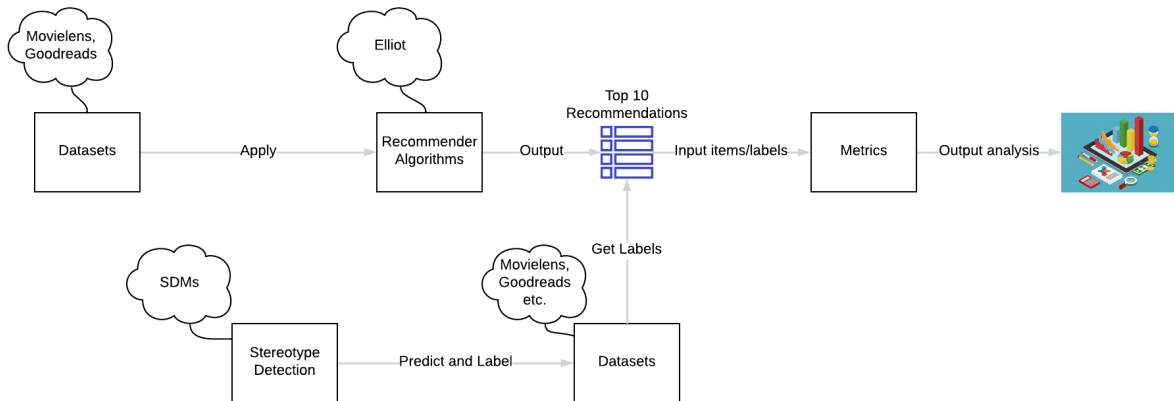
In this equation,  $x_i$  represents an item in the position of the Top-N recommendation list,  $r$  is the item's rank, and  $N$  denotes the number of items in the list. The numerator accumulates the weighted sum of stereotypical items, considering their ranks and the denominator normalizes this sum against the maximum possible score (all items are stereotypical), resulting in a value between 0 and 1. A score of 0 means no stereotypes in the list, and a score of 1 indicates that every item in the list contains a stereotype.

For each gender, race, and religion stereotype, we will refer to REC-ST as  $REC-ST_{Gender}$ ,  $REC-ST_{Race}$ , and  $REC-ST_{Religion}$  respectively.

### 3.5. Experimental Setup

Before diving into the experiments we explain our setup. We used SURF's Research Cloud to gain access to Virtual Machines (VMs) with Ubuntu 20.04 and an NVIDIA A10 GPU.<sup>12</sup> On these VMs, we deploy Elliot [2], a comprehensive recommendation framework for RA research. Although several frameworks offer RA research, we select Elliot due to its extensive range of RAs and its simplicity in conducting experiments. It provides an end-to-end process for RA research, providing parts such as data loading and hyperparameter tuning to execution of significance tests on the output of RAs.

We conduct two experiments: the first experiment explores the performance of the RAs based on the generated recommendation lists, while the second experiment examines the presence of stereotypes in the recommendations obtained from the first experiment.<sup>13</sup> We consider the Top-10 recommendations from the RAs for each experiment. A general overview of the workflow of our proposed empirical exploration is given in Figure 3.6.



**Figure 3.6:** The proposed workflow of our empirical exploration shows how we apply Elliot and the SDMs to perform our analysis.

#### 3.5.1. Experiment 1: Exploring the performance of RAs

In our first experiment, we aim to examine various RAs by generating Top-10 recommendation lists and analyzing them using the performance metrics outlined in Section 3.4. Evaluating these lists allows us to contextualize the RAs and understand their performance nuances on the selected datasets, highlighting notable differences between them. We specifically focus on understanding how these RAs perform when tailoring recommendation lists for children. We detail the process steps below.

<sup>12</sup><https://servicedesk.surf.nl/wiki/display/WIKI/Research+Cloud+Documentation>

<sup>13</sup>Whenever we discuss users of  $GR_{Ch}$  in these experiments, we assume these to be children since all the books considered are tailored to children.

**Datasets and Preprocessing** Recall that the datasets we consider for the performance are  $ML_{Ch}$  and  $GR_{Ch}$ . Given the order-of-magnitude difference in size between ML and  $ML_{Ch}$  (see Table 3.1), we opted to use the much larger ML as our training set and evaluate the  $ML_{Ch}$  set to overcome the limitations of insufficient training data when using  $ML_{Ch}$ .

We follow the footsteps of Anelli et al. [3] who utilized ML with Elliot. Here only a few RAs are considered and the ML dataset is without side information, meaning that RAs that require extra data, e.g. content-based models requiring item descriptions, are not able to run with this configuration. To utilize RAs such as content-based, we leverage Elliot’s knowledge graph for ML [4]. We follow the same preprocessing steps of Anelli et al. [3], however, we incorporate the knowledge graph and refer to this dataset with side information as  $ML_{Ch,S}$ .<sup>14</sup> To the best of our knowledge, no previous research used the  $GR_{Ch}$  dataset with Elliot, hence we applied the same preprocessing steps to  $GR_{Ch}$  that were also applied to ML in previous research.

Elliot provides us with several pre-filtering and data-splitting options. For both ML and  $GR_{Ch}$ , we transform the rating data (explicit feedback) to implicit feedback (see Section 2.1.1) by setting Elliot’s *global\_threshold* to 4 and considering ratings with 4 and 5 as positive unary signals. This means all ratings of 4 and 5 are changed to 1’s in the user-item matrix and all other values are set to 0.

Because real-world datasets are very sparse, a common pre-processing step is to reduce this sparsity by filtering out users and items with less than *k-core* interactions. We achieve this by using Elliot’s *iterative\_k\_core* variable which iteratively filters out users and items with less than *k* interactions until no changes occur anymore in the dataset. Common values for *k* are 5 or 10 and we set  $k = 10$  for ML and  $k = 20$  for  $GR_{Ch}$ . Because  $GR_{Ch}$  is quite large compared to the ML dataset, it takes a lot longer to train and validate the RAs. We chose a less common value for  $GR_{Ch}$  to reduce this time significantly for the RAs because the number of users, items, and ratings is also reduced. The datasets’ characteristics before and after pre-processing are summarized in Table 3.2.

Dataset	k-core	before pre-processing			after pre-processing		
		#users	#items	#interactions	#users	#items	#interactions
$ML_S$	10	6,040	3,883	1,000,209	5,949	2,654	536,647
$GR_{Ch}$	20	542,145	124,082	10,059,349	43,294	14,935	2,483,230

**Table 3.2:** Details on datasets before and after pre-processing.

**Dataset Splitting** After preprocessing comes dataset splitting. We used Elliot’s *random\_subsampling* method to randomly split the datasets in a 5-fold 80-20 train-test split. This results in 5 separate train-test sets where 80% of users’ ratings are used for training the RAs and 20% of users’ ratings are used for testing or evaluating the RAs.<sup>15</sup>

**Hyperparameter Tuning** Elliot integrates an automated process for hyperparameter search and evaluation of RAs. We used the embedded search method in Elliot named *Tree Parzen Estimators*<sup>16</sup> (TPE) [7]. The experiments, including the hyperparameter ranges, can be defined through text-based configuration files. We extensively tune the hyperparameters of the RAs in a similar manner done in previous work [3, 86]. If a considered RA was not present in these works, we turned to the original papers of the RA to identify suitable ranges. When this was not possible we used similar ranges of the RAs present in previous work. All ranges for the models are included in the appendix. Although these ranges are based on previous research, it is important to note that we do not rule out that there exists a better set of hyperparameters

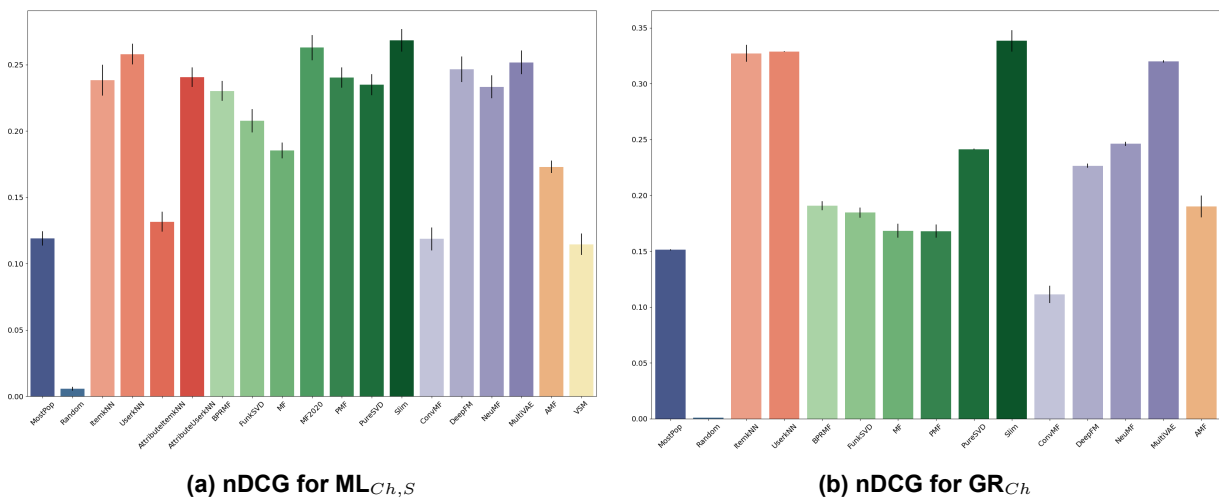
<sup>14</sup>It is important to note that for training purposes, we use the entire ML dataset and we refer to this as  $ML_S$ . Furthermore, we built upon the steps of previous work, however, we include our initial results when reproducing the prior work to show that our results align with previous work in Appendix A.

<sup>15</sup>In our case the test set will only consist of users with age < 18, i.e., the  $ML_{Ch}$  dataset.

<sup>16</sup>TPE is an efficient algorithm for hyperparameter tuning. It does so by building a probabilistic model that focuses on regions in the hyperparameter search space where improvements are more likely to occur.

for the RAs. The RAs need to be “good enough” for our experiments to be able to contextualize them for our exploration.<sup>17</sup> The nDCG was used as a target metric during the hyperparameter optimization. As mentioned before, to the best of our knowledge, no previous work exists that utilizes  $GR_{Ch}$  with Elliot therefore we adopt the same hyperparameter tuning strategy on all datasets.

**Performance Evaluation** After completing the hyperparameter tuning, we assess the variability of the RAs across different cross-validation folds. Figure 3.7 displays the variability for both datasets based on the nDCG. Due to the nature of cross-validation and hyperparameter tuning in Elliot, the optimal model can emerge from any fold. This poses a challenge for conducting valid t-tests, as Elliot’s integrated t-tests require data from the same population. To address this and still utilize the t-tests, we execute each fold independently on separate VMs. Given the observed minimal variability across the folds for all datasets, we select and report results from an arbitrary fold for each dataset. This approach ensures that we can discuss both the statistical significance between pairs of RAs and the consistent behavior of the models on the datasets. The significance tests are based on paired t-tests with  $p < 0.05$  and Bonferroni correction ( $n = 19$  for  $ML_{Ch,S}$  and  $n = 15$  for  $GR_{Ch}$ ) unless stated otherwise. Detailed performance results of the RAs are presented in Chapter 5.



**Figure 3.7:** Mean and std of the nDCG for all RAs on  $ML_{Ch,S}$  and  $GR_{Ch}$ . The black stripes on top of the bar showcase the std, the longer the line the larger the std.

### 3.5.2. Experiment 2: Exploring stereotype presence

In our second experiment, our primary objective is to explore the extent to which stereotypes are present among the Top-10 recommendations generated by RAs. To achieve this, we employ a variety of SDMs detailed in Section 3.3, each designed to detect and analyze stereotypes from a different lens. Using a multi-faceted approach helps us to gain a thorough understanding of the prominence of stereotypes in the recommendations. Below we describe the specifics of how we applied these SDMs to our chosen datasets and how we leveraged the stereotype prominence metrics, as detailed in Section 3.4.2, to guide our analysis.

Each SDM has its unique requirements. We process our datasets to be suitable for each model in order to perform the analysis. Using our naive strategy, NGIM, we identify gender stereotypes based on name-gender associations in movies and books. This approach categorizes items as stereotypical when male entities surpass female entities. For BiasMeter, we tokenize the descriptions into sentences and use the Stanford Certainty Factor to aggregate the probabilities representing the likelihood of the stereotype labels. The specifics of this aggregation, including our chosen threshold, are detailed in Section 3.3.2. With ChatGPT 3.5, we utilize its API, feeding it movie and book descriptions, to identify and categorize

<sup>17</sup>Shehzad and Jannach [86] mention that the RAs should be *good enough* for their experiments, meaning that a set of tuned hyperparameters for an RA should consistently outperform a random set of hyperparameters for all other RAs. In our case, we consider an RA to be *good enough* by validating our approach with previous work.

potential stereotypes. Both BiasMeter and ChatGPT 3.5’s approaches consider gender, race, and religion stereotypes. Once we have all the stereotype labels from each SDM for  $ML_{Ch,S}$  and GR, we apply the stereotype prominence metrics in combination with the Top-10 suggestions for analysis.

In this experiment, our primary aim is to discern patterns of stereotype prominence in suggestions made to children by RAs. Specifically, we compare and contrast the frequency and type of stereotypes, such as gender, race, and religion stereotypes in these recommendations based on the stereotype prominence metrics. We investigate whether certain RAs are more likely to expose children to stereotypes and if some algorithms are particularly prone to specific types of stereotypes. By examining these patterns across different algorithms and different stereotypes, we aim to gain a deeper understanding of how current RAs operate with respect to stereotype exposure. This is crucial not just for understanding the current state of RAs but also for future developments in this field to ensure that RAs do not propagate stereotypes, especially when the target audience is impressionable, such as children.

To statistically validate our findings, we conduct pairwise comparisons between RAs using a paired t-test with  $p < 0.05$  and Bonferroni correction with  $n = 19$  for  $ML_{Ch,S}$  and  $n = 15$  for  $GR_{Ch}$ .

# 4

## Ethical Considerations

### 4.1. Data Management

We only use two publicly available datasets for our empirical exploration, which are ML and GR<sub>Ch</sub>. We only produce one repository containing all code and necessary data for reproducing our results. We do not redistribute the datasets used in our exploration since that violates the user license agreements of these datasets. Therefore, we do not generate any datasets or data that needs to be stored, making our data management plan straightforward.

### 4.2. Ethics

Research should always be carried out carefully, especially when children are involved. In our research, the data is already publicly available and we do not participate in any form of data collection. Since the data is already publicly available and anonymized, we did not require ethical approval as there are no additional risks or vulnerabilities because of our research. This decision was made with the collaboration of TU Delft's Human Research Ethics Committee.

#### 4.2.1. Authorship policy

In the writing of this manuscript, we have utilized tools such as Grammarly and ChatGPT to assist with grammar and spelling corrections. The use of these tools was strictly limited to these aspects of writing and did not influence the generation of ideas, research methodologies, or the academic content of this work, which are entirely our own.

# 5

## Results

In this chapter, we present the results of the empirical explorations conducted using the methodology presented in Chapter 3. We use results from Experiment 1 to contextualize the performance of the RAs. We use results from Experiment 2 to explore the presence of stereotypes. Analysis of emerging findings and associated discussions allows us to answer the research question. Along the way, we present the potential implications emerging from both experiments.

### 5.1. Experiment 1: Performance of RAs

We first examine the performance of RAs from multiple perspectives. Figures 5.1 and 5.2 illustrate the mean performance for all the performance metrics for both  $ML_{Ch,S}$  and  $GR_{Ch}$  respectively. These figures also indicate the pairwise comparison results of the RAs under study, highlighting statistically significant differences confirmed by a paired t-test with  $p < 0.05$  and Bonferroni correction with  $n = 19$  for  $ML_{Ch,S}$  and  $n = 15$  for  $GR_{Ch}$  unless noted otherwise. The exact performance values of these figures are depicted in Tables 5.1 and 5.2 for  $ML_{Ch,S}$  and  $GR_{Ch}$ . Below we highlight observed trends from these results.<sup>1</sup>

**Overall performance of RAs for children** For  $ML_{Ch,S}$ , ConvMF is the lowest-performing personalized RA across all metrics. It only exceeds MostPop slightly based on MRR. These results are significant across all RAs, except for MostPop and VSM, which are not significant based on all the performance metrics, and for nDCG and MAP, ConvMF is not significant compared to AttributeItemkNN.

MF2020 is the best performing RA based on the nDCG ( $= 0.2696$ ), meaning that the relevant items are positioned higher in the Top-10 suggestions compared to its peers. UserkNN yields the highest MRR ( $= 0.4811$ ), meaning that on average, the first relevant item is positioned higher compared to the rest. This is significantly better than the worst RAs: ConvMF ( $= 0.2346$ ), and MostPop ( $= 0.2269$ ). Slim has the highest MAP ( $= 0.2586$ ), meaning that on average it ranks relevant items higher compared to the other RAs. Again, ConvMF has the lowest MAP ( $= 0.1154$ ) value. Both DeepFM and MultiVAE have the highest HR ( $= 0.8211$ ), meaning these RAs recommend at least one item the most to users. ConvMF is yet again the lowest with  $HR = 0.5275$ .

For  $GR_{Ch}$  we see a similar trend in terms of the worst performing RA. Here, ConvMF is significantly worse than all other personalized RAs and it even performs worse than the non-personalized MostPop algorithm, for all the metrics. The best-performing model on  $GR_{Ch}$  is SLIM. It outperforms all other RAs significantly for each performance metric.

**Per-category analysis** For the *non-personalized* RAs, MostPop significantly outperforms the Random model on both datasets as expected.

Among the *neighborhood-based* models, AttributeItemkNN is the worst for  $ML_{Ch,S}$  (Bonferroni  $n = 4$ ). Although not significant, UserkNN performed the best across all the performance metrics and achieved the same HR compared to AttributeUserkNN. In fact, aside from AttributeItemkNN, the models show similar performance for all the metrics on  $ML_{Ch,S}$ . For  $GR_{Ch}$ , the performance of UserkNN and ItemkNN are

<sup>1</sup>It is important to note that for  $ML_{Ch,S}$  these results are based on the 222 extracted users below the age of 18. We added the performance results for all users in Appendix A, Table A.4.

similar as well. However, UserkNN has a statistically significant but slight edge over ItemkNN for the MRR and HR metrics.

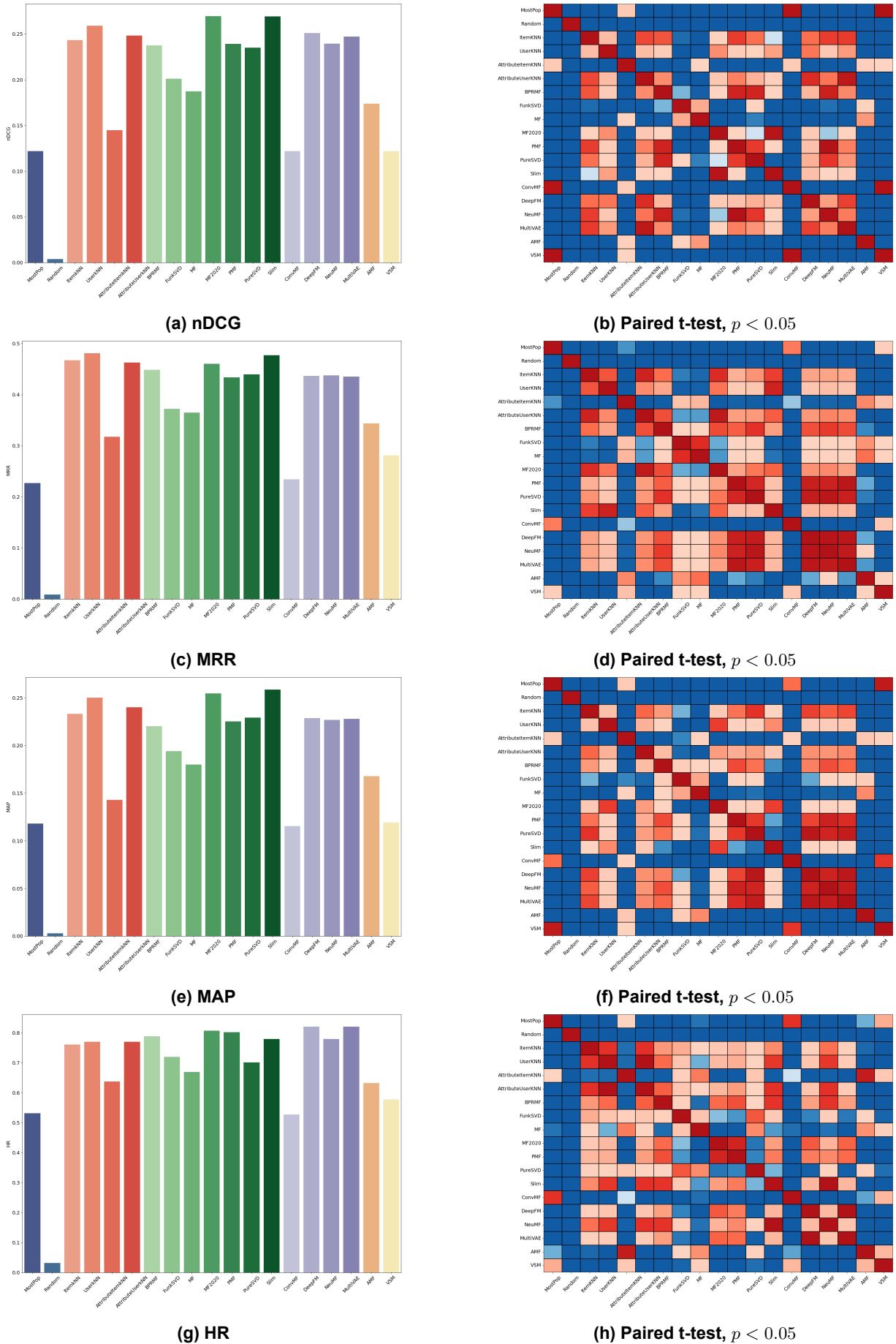
Across all *latent factor models* for  $ML_{Ch,S}$ , MF performed the worst for each performance metric. This result is statistically significant (Bonferroni  $n = 7$ ) compared to the other latent factor RAs for each metric except FunkSVD for each metric, and PureSVD for MRR and HR. The best RA for nDCG and HR is MF2020, and for MAP and MRR Slim performs the best. Slim is statistically significant compared to all other models except MF2020 for nDCG, MRR, and MAP. The HR is close for most of the latent factor models. For  $GR_{Ch}$ , the behavior of the RAs is similar to each other with the exception of Slim and PureSVD, which show significantly higher performances than the rest across all performance metrics. All the latent factor RA pairs are statistically significant for all the metrics with the exception of the pair of MF and PMF for MAP and the pair of BPR-MF and PMF for HR.

For the  $ML_{Ch,S}$  dataset, all *artificial neural network* RAs show comparable performance across the metrics we considered for analysis purposes. The notable exception is ConvMF, which underperforms the other neural models (Bonferroni  $n = 4$ ). For  $GR_{Ch}$ , there are salient differences in the performance of the RAs. Again, ConvMF is the worst neural model. However, we see that MultiVAE outperforms the other neural models. DeepFM and NeuMF exhibit similar trends but NeuMF marginally outperforms DeepFM. All the performance differences between the neural models are statistically significant.

As AMF and VSM are the only representatives in their category, the insights are straightforward.

**Inter-Category Comparison Analysis** In the  $ML_{Ch,S}$  dataset, we observe for the personalized categories that the *neighborhood-based*, *latent factor models*, and *artificial neural network* RAs demonstrate comparable performance, with no significant differences between them. On the other hand, *adversarial* and *content-based* RAs underperform compared to the other categories. We see from Figure 5.1 that the results are statistically significant when RAs outperform AMF and VSM by a large margin. When the performance differences become more marginal, these results show no statistical significance. All categories significantly outperform the *non-personalized* RAs with the exception of AttributeItemkNN, ConvMF, and VSM.

For  $GR_{Ch}$ , the *neighborhood-based* models as a category outperform the others. Within this category, both ItemkNN and UserkNN surpass all other RAs in performance, with the only exceptions being Slim across all metrics and MultiVAE for the HR metric. The comparisons of ItemkNN and UserkNN to all the other RAs are statistically significant. Meanwhile, the *artificial neural network* RAs generally surpass most of the *latent factor models*. Furthermore, despite not being the absolute weakest among individual RAs, the *adversarial* RA category as a whole is underperforming when compared to other personalized categories. Again, all categories outperform the *non-personalized* RAs with all the paired performance comparisons being statistically significant with the exception of the comparison between MostPop and PMF for MRR.

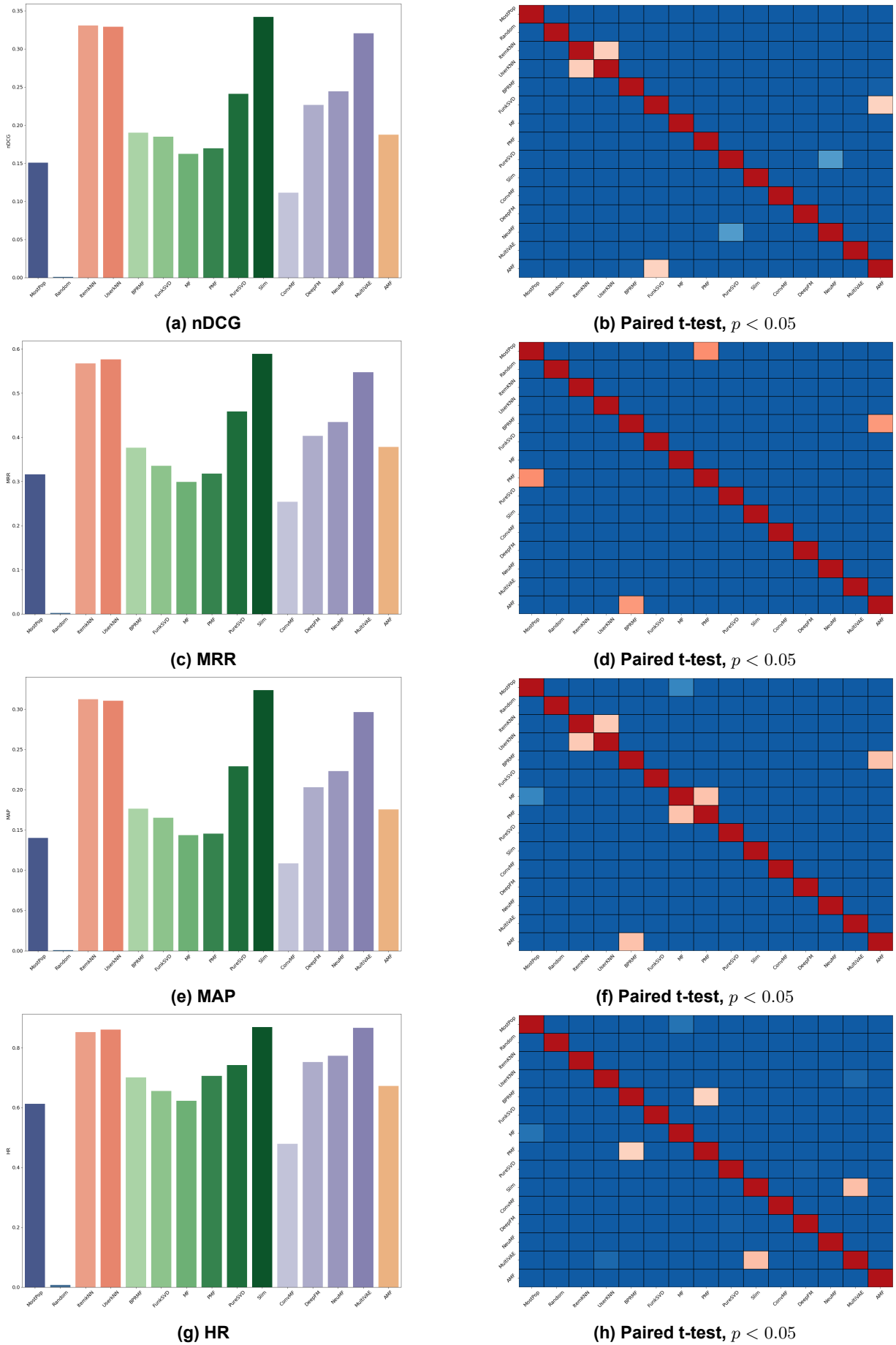


**Figure 5.1:** Mean of the performance metrics and paired t-test results for all RAs on the  $ML_S$  dataset. Shades of colors indicate a category from Section 3.2.



	$ML_S$			
	nDCG	MRR	MAP	HR
<b><i>Non-personalized</i></b>				
MostPop	0.1219	0.2269	0.1181	0.5321
Random	0.0038	0.0091	0.0028	0.0321
<b><i>Neighborhood-based</i></b>				
Item-kNN	0.2434	0.4671	0.2331	0.7615
<i>User-kNN</i>	0.2591	<b>0.4811</b>	0.2500	0.7706
AttributeItem-kNN	0.1449	0.3174	0.1428	0.6376
AttributeUser-kNN	0.2482	0.4627	0.2400	0.7706
<b><i>Latent Factor Models</i></b>				
BPR-MF	0.2375	0.4483	0.2201	0.7890
FunkSVD	0.2012	0.3723	0.1940	0.7202
MF	0.1873	0.3649	0.1799	0.6697
<i>MF2020</i>	<b>0.2696</b>	0.4604	<u>0.2545</u>	<u>0.8073</u>
PMF	0.2393	0.4340	0.2251	0.8028
PureSVD	0.2352	0.4395	0.2292	0.7018
Slim	<u>0.2693</u>	<u>0.4768</u>	<b>0.2586</b>	0.7798
<b><i>Artificial Neural Networks</i></b>				
ConvMF	0.1218	0.2346	0.1154	0.5275
<i>DeepFM</i>	0.2512	0.4367	0.2285	<b>0.8211</b>
NeuMF	0.2394	0.4376	0.2268	0.7798
<b><i>Adversarial Learning</i></b>				
<i>AMF</i>	0.1739	0.3434	0.1677	0.6330
<b><i>Autoencoders</i></b>				
<i>MultiVAE</i>	0.2473	0.4353	0.2279	<b>0.8211</b>
<b><i>Content-Based</i></b>				
<i>VSM</i>	0.1219	0.2811	0.1191	0.5780

**Table 5.1:** Results for the RAs performance on  $ML_{Ch,S}$ . The best model is highlighted in **bold** and the second best model is highlighted with an underline. The best model per RA type based on nDCG is highlighted in *italic*.



**Figure 5.2:** Mean of the performance metrics and paired t-test results for all RAs on the  $GR_{Ch}$  dataset. Shades of colors indicate a category from Section 3.2.

	$GR_{Ch}$			
	nDCG	MRR	MAP	HR
<b>Non-personalized</b>				
<i>MostPop</i>	0.1508	0.3163	0.1400	0.6122
Random	0.0008	0.0022	0.0008	0.0077
<b>Neighborhood-based</b>				
<i>Item-kNN</i>	<u>0.3307</u>	0.5672	<u>0.3123</u>	0.8522
User-kNN	0.3289	<u>0.5766</u>	0.3105	0.8607
AttributeItem-kNN*	-	-	-	-
AttributeUser-kNN*	-	-	-	-
<b>Latent Factor Models</b>				
BPR-MF	0.1900	0.3763	0.1765	0.7006
FunkSVD	0.1847	0.3356	0.1653	0.6552
MF	0.1622	0.2989	0.1436	0.6226
MF2020**	-	-	-	-
PMF	0.1695	0.3177	0.1454	0.7058
PureSVD	0.2410	0.4583	0.2289	0.7422
<i>Slim</i>	<b>0.3420</b>	<b>0.5891</b>	<b>0.3236</b>	<b>0.8693</b>
<b>Artificial Neural Networks</b>				
ConvMF	0.1113	0.2537	0.1084	0.4790
DeepFM	0.2263	0.4032	0.2031	0.7520
<i>NeuMF</i>	0.2444	0.4347	0.2230	0.7734
<b>Adversarial Learning</b>				
<i>AMF</i>	0.1874	0.3780	0.1753	0.6720
<b>Autoencoders</b>				
<i>MultiVAE</i>	0.3203	0.5472	0.2963	<u>0.8666</u>
<b>Content-Based</b>				
VSM*	-	-	-	-

**Table 5.2:** Results for the RAS performance on  $GR_{Ch}$ . The best model is highlighted in **bold** and the second best model is highlighted with an underline. The best model per RA type based on nDCG is highlighted in *italic*. Note that '\*' indicates no results due to unavailability of side information; '\*\*' indicates that the model was excluded because it was too slow on the  $GR_{Ch}$  dataset

## 5.2. Experiment 2: Exploring Stereotype presence with SDMs

In this section, we present the results of Experiment 2 to analyze stereotype prominence, based on the different SDMs introduced in Section 3.3, among Top-N suggestions. We discuss our findings per stereotype, highlighting any trends observed in the recommendations of RAS. The number of items labeled as stereotypical by the SDMs is presented in Table 5.3. Whenever we compare stereotype prominence of RAS, all results are significant unless stated otherwise. We conduct pairwise comparisons between RAS, based on a paired t-test with  $p < 0.05$  and Bonferroni correction,  $n = 19$  ( $ML_{Ch,S}$ ),  $n = 15$  ( $GR_{Ch}$ ).

	$ML_{Ch,S}$			$GR_{Ch}$		
	Gender	Race	Religion	Gender	Race	Religion
<b>NGIM</b>	2866	-	-	3438	-	-
<b>BiasMeter</b>	2423	374	85	8174	931	313
<b>ChatGPT 3.5</b>	192	79	17	356	50	8

**Table 5.3:** Amount of stereotype labels assigned based on each SDM to items of  $ML_{Ch,S}$  (3880 movies) and  $GR_{Ch}$  (14935 books).

### 5.2.1. Gender Stereotypes

For the  $ML_{Ch,S}$  dataset, we present results from each SDM in Figure 5.3 for  $HIT_{BAD,Gender}$ , Figure 5.5 for  $MRR_{BAD,Gender}$ , and Figure 5.7 for  $REC-ST_{Gender}$ . Results for  $GR_{Ch}$  are demonstrated in Figures 5.4, 5.6, and 5.8.

Reported  $HIT_{BAD,Gender}$  scores show that both NGIM and BiasMeter consistently achieve an average  $HIT_{BAD,Gender}$  of 1 for every RA. This means that each algorithm, on average, presents at least one item containing gender stereotypes to all the users. These results show no statistical significance since all RAs exhibit the same behavior of promoting items containing stereotypes.

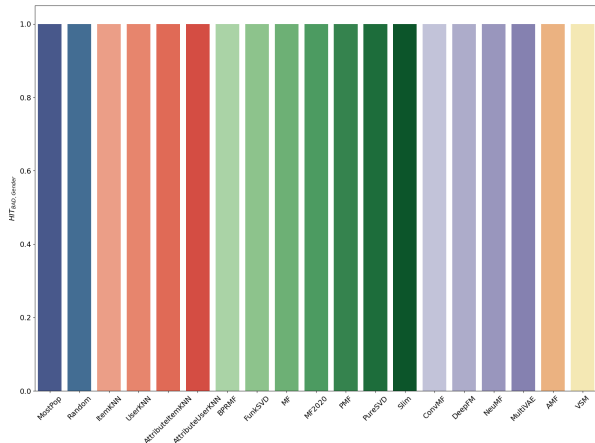
For  $GR_{Ch}$ , we see slight but significant differences for both NGIM and BiasMeter.

The lowest  $HIT_{BAD,Gender}$  based on NGIM is achieved by the Random RA ( $= 0.9281$ ) and ItemkNN ( $= 0.9298$ ). The difference between the  $HIT_{BAD,Gender}$  value of the two is not significant. The RAs with the highest  $HIT_{BAD,Gender}$  being ConvMF ( $= 0.9981$ ), AMF ( $= 0.9948$ ), and MostPop ( $= 0.9967$ ).

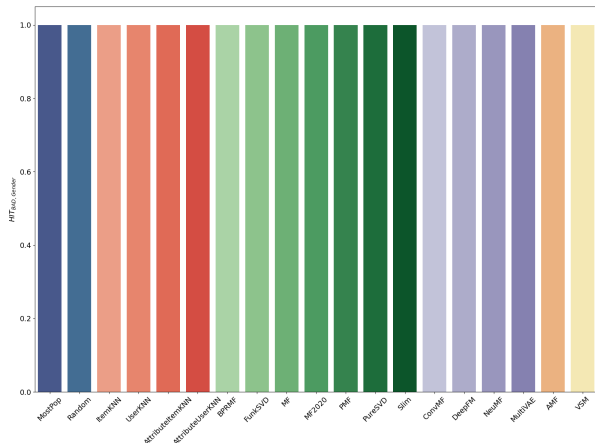
Results from BiasMeter indicate that MostPop and ConvMF are the worst in terms of  $HIT_{BAD,Gender}$  ( $= 1$ ) compared to other RAs. A comparison between the two yields no significant difference. Furthermore, both are not statistically significant compared to PMF. ItemkNN has the lowest  $HIT_{BAD,Gender}$  score ( $= 0.996$ ). This result is statistically non-significant compared to Slim, DeepFM, and NeuMF.

Turning our attention to ChatGPT 3.5, we see varied behaviors among the RA suggestions based on  $ML_{Ch,S}$ . Specifically, MostPop and ConvMF show items containing gender stereotypes to more users compared with other RAs. AMF also stands out, with a higher  $HIT_{BAD,Gender}$  than most other RAs. The exceptions are the AttributeItemkNN and Random RAs, for which comparisons with AMF are statistically non-significant. Meanwhile, ItemkNN is the RA that shows items containing gender stereotypes to the least amount of users. This result is statistically non-significant compared to FunkSVD, MF, PureSVD, Slim, DeepFM, and MultiVAE.

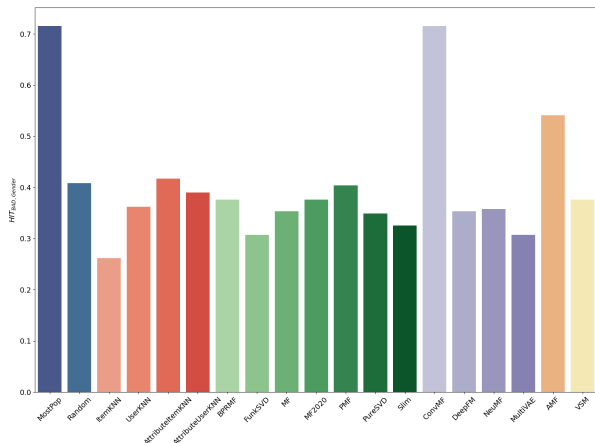
When examining the  $GR_{Ch}$  dataset, another observation worth mentioning arises. The behavior of MostPop, ConvMF, and AMF is completely inverted, going from the highest  $HIT_{BAD,Gender}$  score (MostPop and ConvMF  $= 0.716$ , AMF  $= 0.541$ ) to having significantly the lowest  $HIT_{BAD,Gender}$  score (MostPop  $= 0.0022$ , ConvMF  $= 0.0017$ , AMF  $= 0.137$ ). Only the comparison between MostPop and ConvMF is not significant.



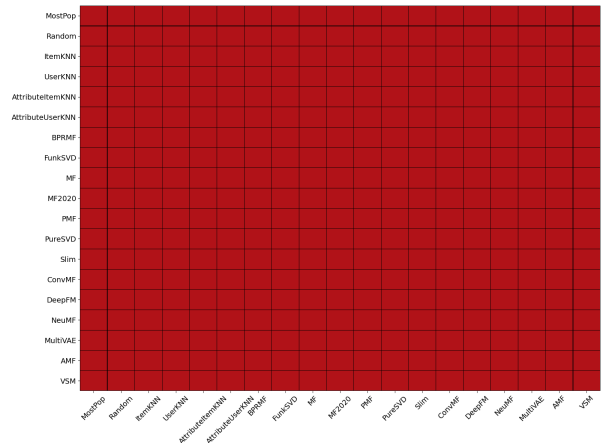
(a)  $HIT_{BAD,Gender}$  with NGIM



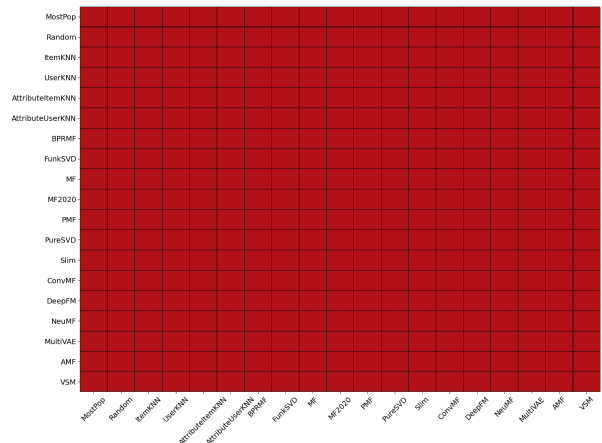
(c)  $HIT_{BAD,Gender}$  with BiasMeter



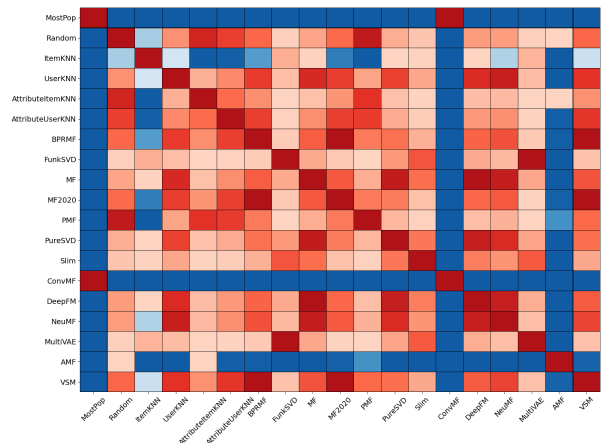
(e)  $HIT_{BAD,Gender}$  with ChatGPT 3.5



(b) Paired t-test,  $p < 0.05$



(d) Paired t-test,  $p < 0.05$



(f) Paired t-test,  $p < 0.05$

Figure 5.3:  $HIT_{BAD,Gender}$  for all SDMs on  $ML_{Ch,S}$ .

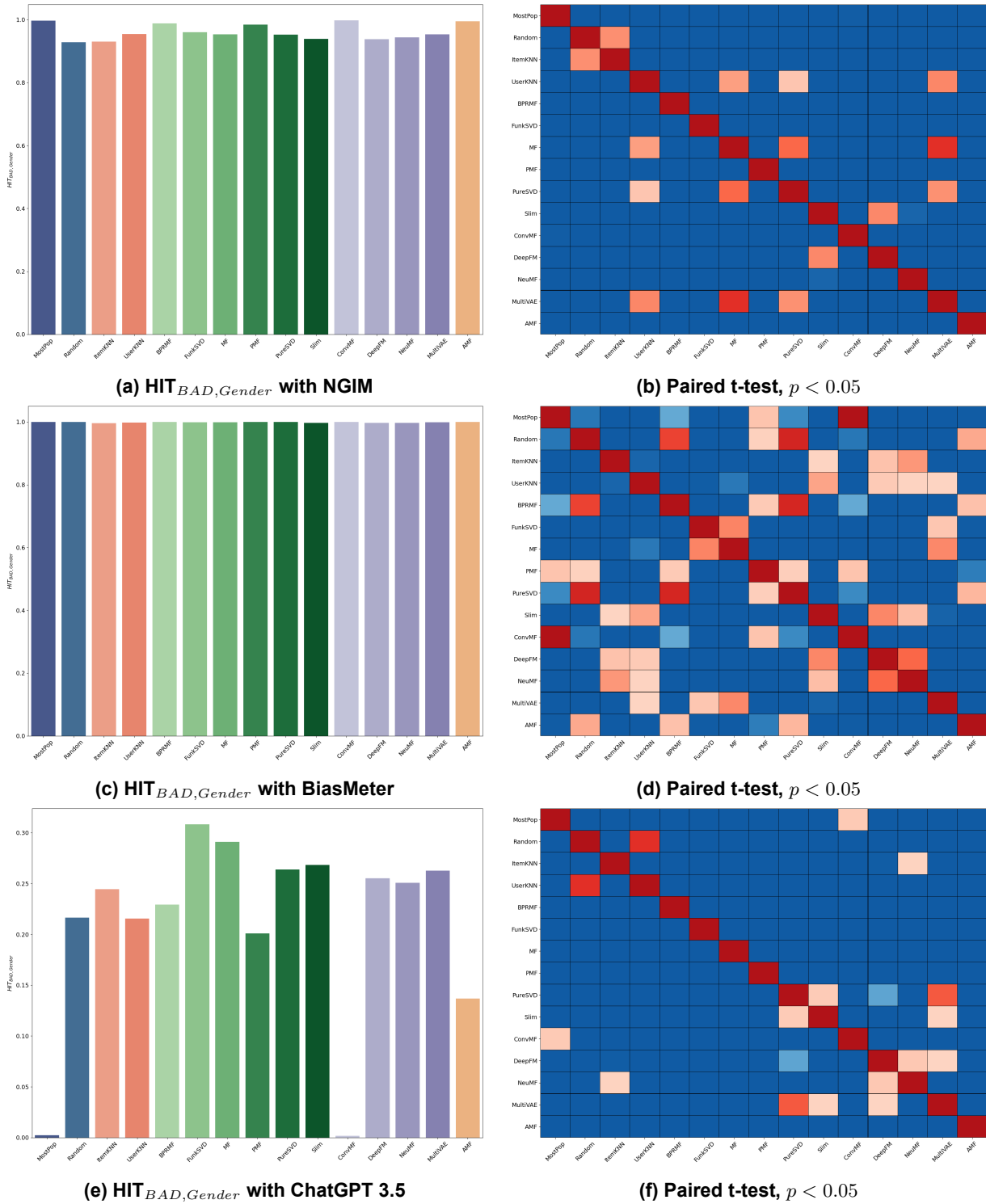


Figure 5.4:  $HIT_{BAD,Gender}$  for all SDMs on  $GR_{Ch}$ .

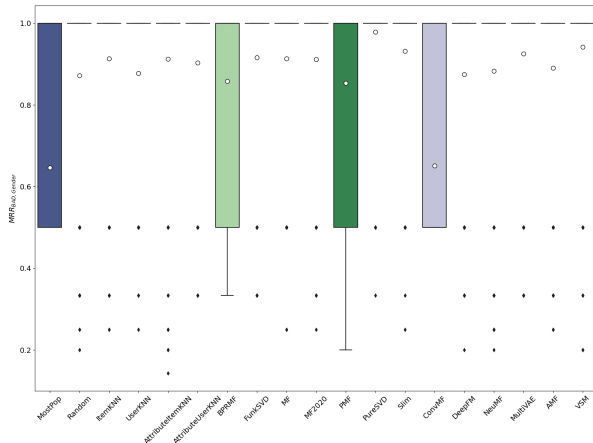
Figure 5.5 shows the  $MRR_{BAD,Gender}$  for the three SDMs applied on  $ML_{Ch,S}$ . The results based on NGIM indicate that most RAs tend to rank the first item containing a gender stereotype higher in the recommendation list. We observe that PureSVD is placing male-dominated movies at the top of the lists the most compared to the other RAs. Only the comparison with VSM is statistically non-significant. Furthermore, we see that MostPop and ConvMF have on average, the lowest  $MRR_{BAD,Gender}$  compared to all the other RAs. However, from the spread of the data, we see that both ConvMF and MostPop consistently have items

at either the top or second position in the list. In contrast, the remainder of the RAs have outliers placing items as low as the last place in the Top-10 list. Just the comparison between MostPop and ConvMF is statistically insignificant.

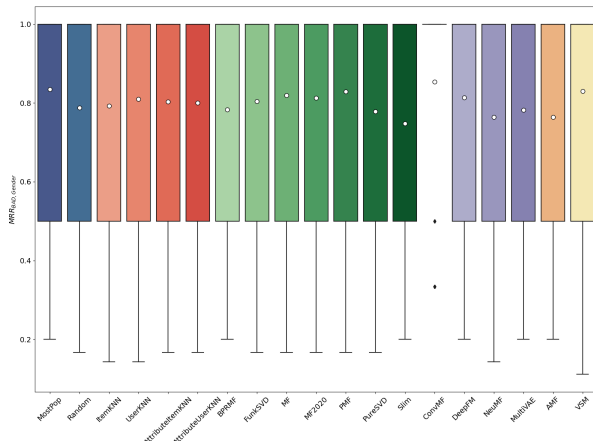
For BiasMeter the results seem similar across all RAs, with most of the items being placed at the Top-2 positions. The only observation worth mentioning is that ConvMF is the only RA whose data spread is mostly fixated at an  $MRR_{BAD,Gender}$  of 1. This means that most users have male-dominated items at the top of their suggestions based on recommendations from ConvMF. However, these comparisons are almost all non-significant.

Observing the results for ChatGPT 3.5 we see that ConvMF has a significantly higher average  $MRR_{BAD,Gender}$  compared to the other RAs. However, most of the gender-stereotypical movies are placed halfway or lower in the Top-10 recommendations.

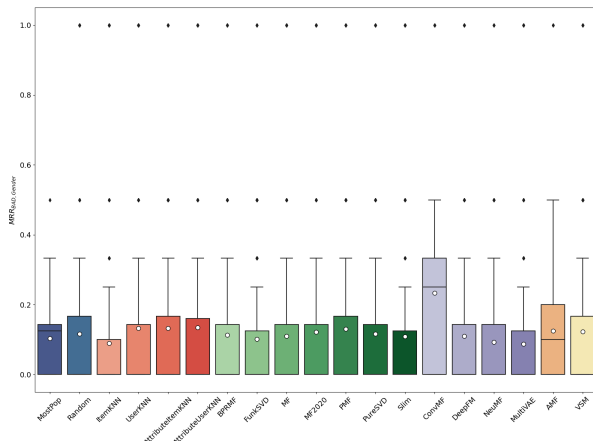
While the  $MRR_{BAD,Gender}$  values are relatively similar among most RAs for the  $ML_{Ch,S}$  dataset, we observe remarkable differences in the  $GR_{Ch}$  dataset (see Fig. 5.6). ConvMF performs well according to NGIM but is the worst according to BiasMeter. It is also worth highlighting that PMF achieves the highest  $MRR_{BAD,Gender}$  ( $= 0.787$ ). Furthermore, based on ChatGPT 3.5, the *latent factor models* and *neural models* have higher  $MRR_{BAD,Gender}$  than their peers. Very few comparisons are statistically non-significant across all RAs.



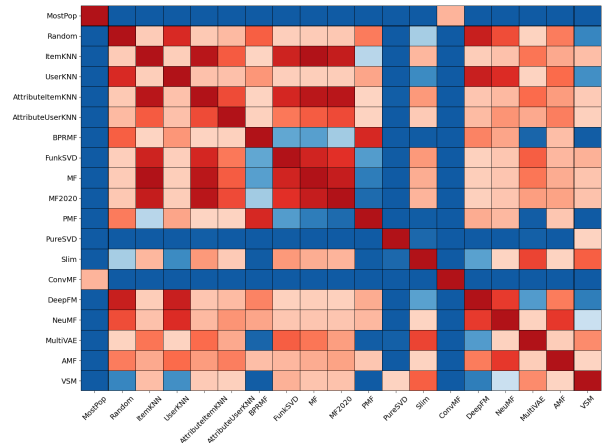
(a)  $MRR_{BAD,Gender}$  with NGIM



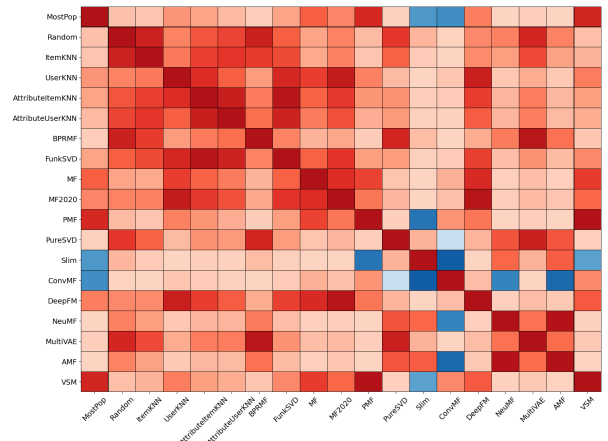
(c)  $MRR_{BAD,Gender}$  with BiasMeter



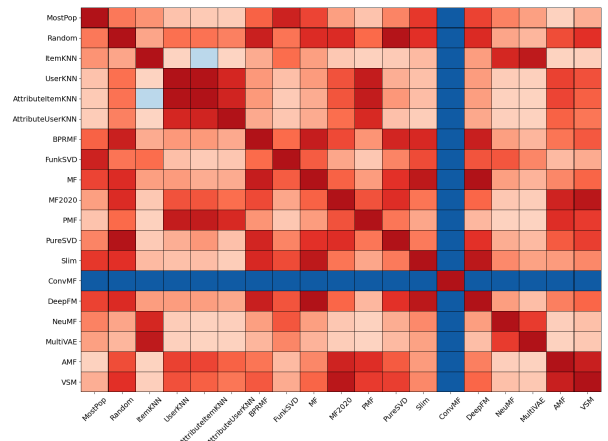
(e)  $MRR_{BAD,Gender}$  with ChatGPT 3.5



(b) Paired t-test,  $p < 0.05$



(d) Paired t-test,  $p < 0.05$



(f) Paired t-test,  $p < 0.05$

Figure 5.5:  $MRR_{BAD,Gender}$  for all SDMs on  $ML_{Ch,S}$ . The white dots indicate the average score.



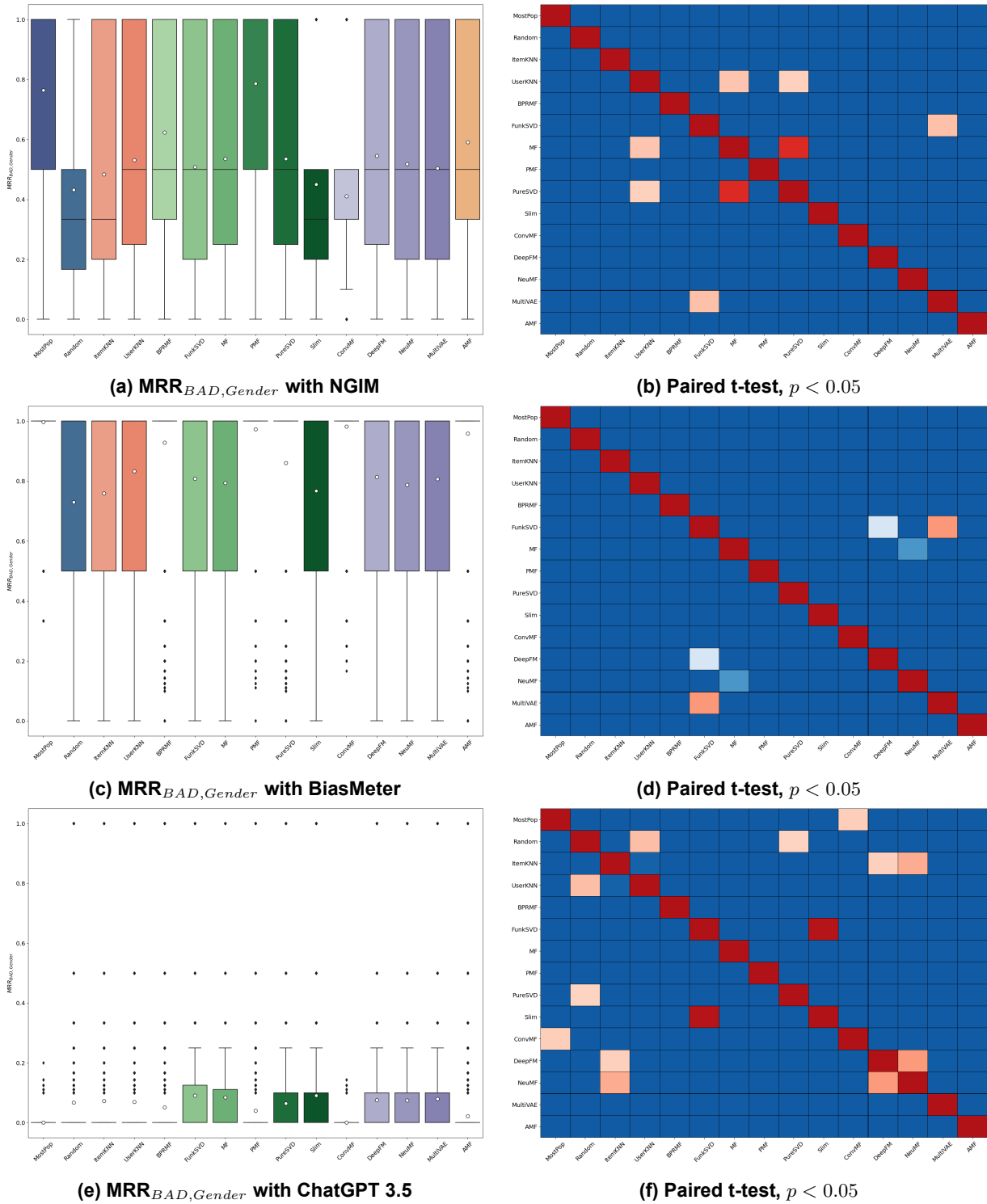


Figure 5.6:  $MRR_{BAD,Gender}$  for all SDMs on  $GR_{Ch}$ . The white dots indicate the average score.

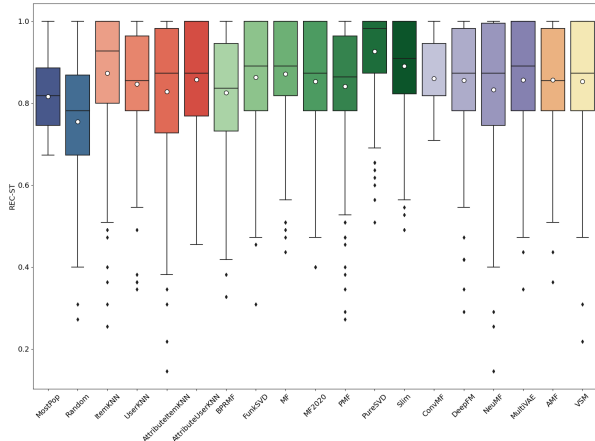
Figure 5.7 presents the  $REC-ST_{Gender}$  for each SDM analyzing the recommendations based on  $ML_{Ch,S}$  for each RA.

We observe that PureSVD emerges as the most problematic RA according to NGIM. It not only suggests a higher number of items containing stereotypes but also prioritizes these items near the top of the Top-10 suggestions list. While the performance appears consistent across most RAs, the Random RA appears to achieve the lowest average  $REC-ST_{Gender}$ , indicating fewer stereotype-driven suggestions.

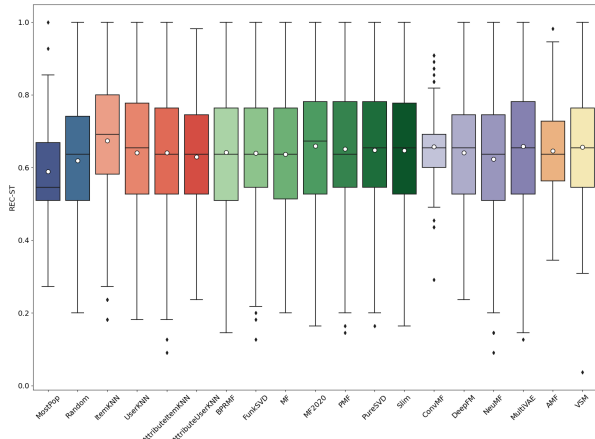
There are no salient observations for BiasMeter. All RAs behave similarly and MostPop has the lowest mean for  $\text{REC-ST}_{Gender}$ . This result is statistically non-significant compared to Random and NeuMF.

According to ChatGPT 3.5, the average  $\text{REC-ST}_{Gender}$  is relatively low. This means that all RAs on average present only a few items and place these at the bottom end of the recommendation lists rather than the top. This is consistent with the  $\text{MRR}_{BAD,Gender}$  of ChatGPT 3.5 in Figure 5.5. Furthermore, ConvMF seems to have the highest  $\text{REC-ST}_{Gender}$  compared to its peers.

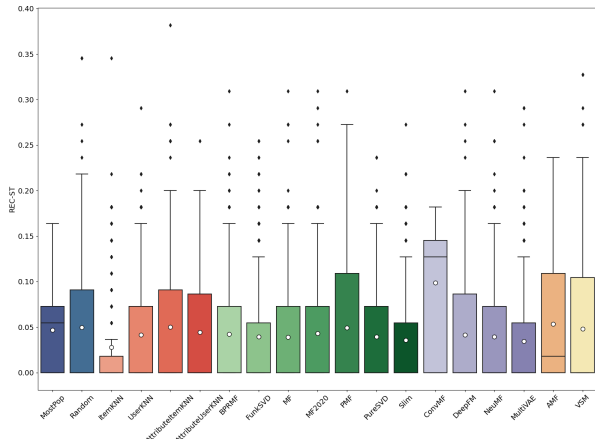
On the  $\text{GR}_{Ch}$  dataset, RAs show the same trends for  $\text{REC-ST}_{Gender}$  compared to its  $\text{MRR}_{BAD,Gender}$  counterpart.



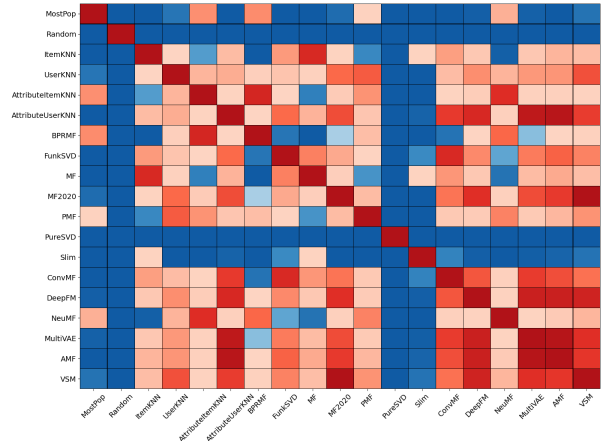
(a) REC-ST<sub>Gender</sub> with NGIM



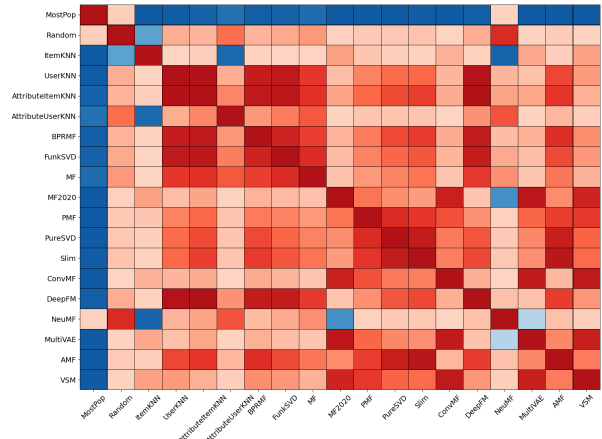
(c) REC-ST<sub>Gender</sub> with BiasMeter



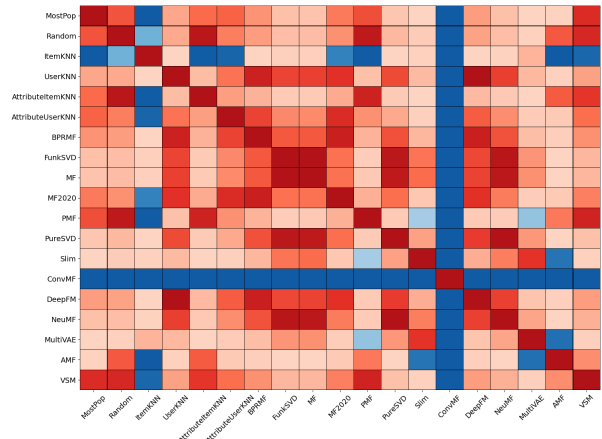
(e) REC-ST<sub>Gender</sub> with ChatGPT 3.5



(b) Paired t-test,  $p < 0.05$



(d) Paired t-test,  $p < 0.05$



(f) Paired t-test,  $p < 0.05$

Figure 5.7: REC-ST<sub>Gender</sub> for all SDMs on  $ML_{Ch,S}$ . The white dots indicate the average score.

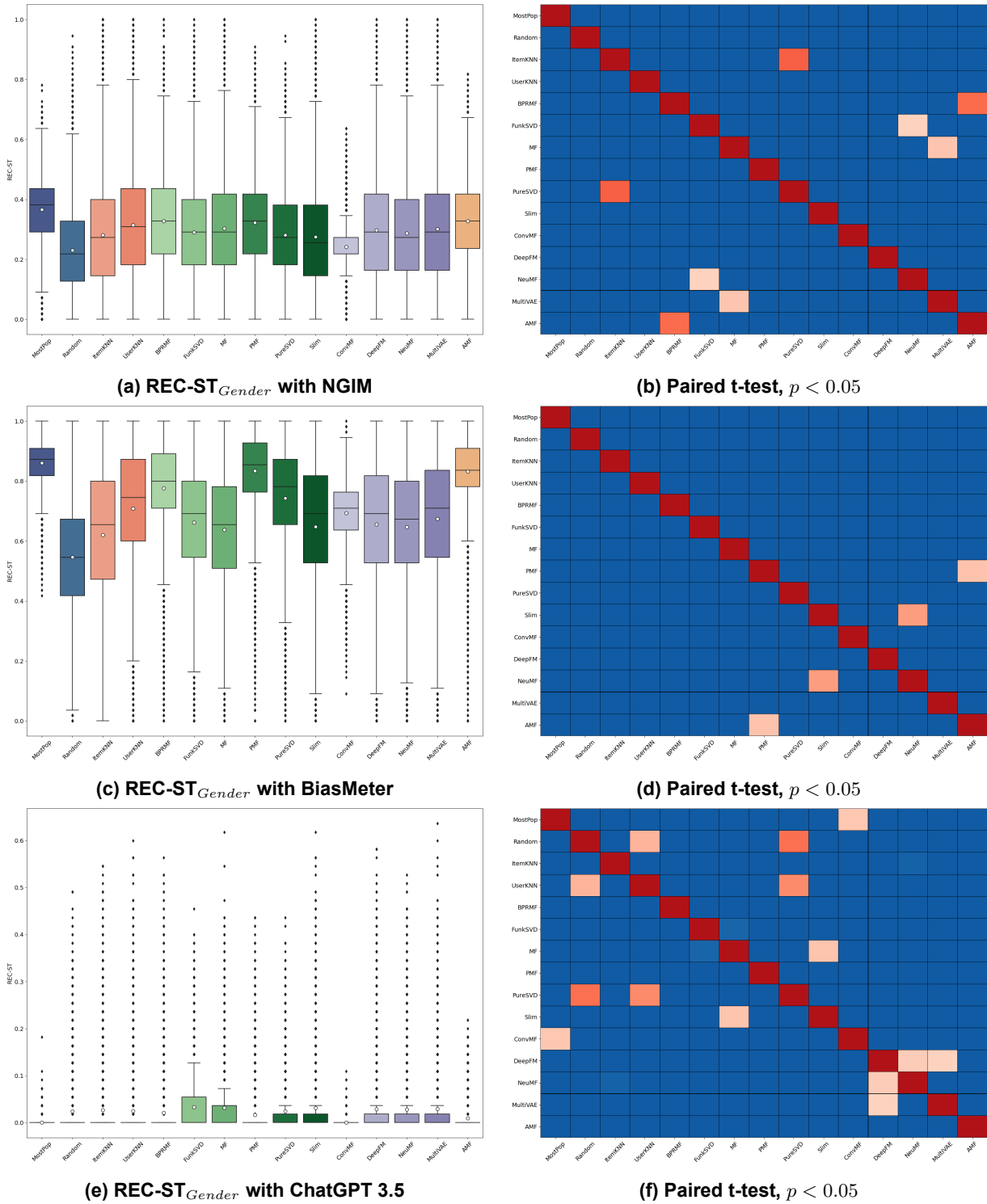


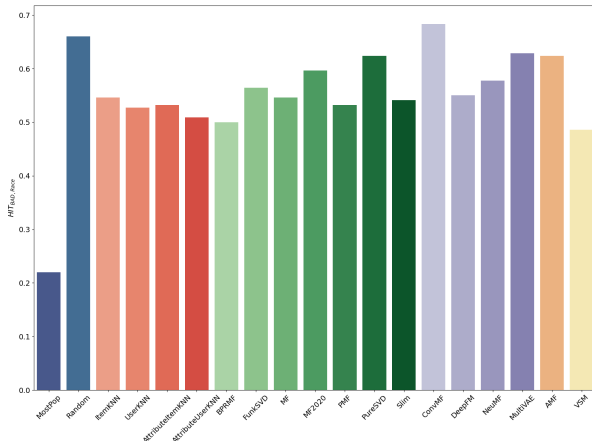
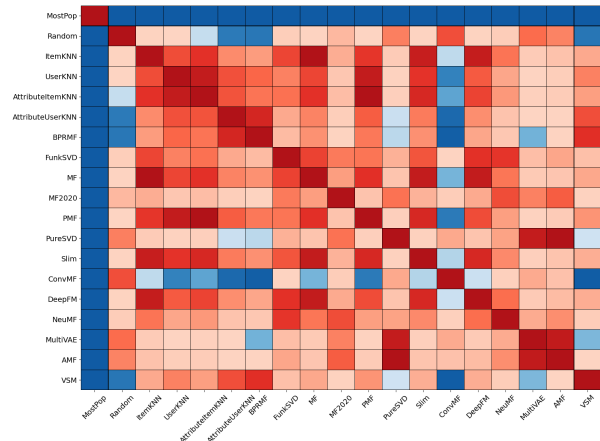
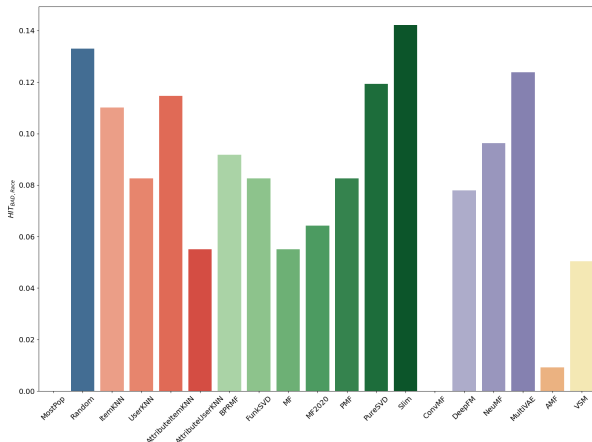
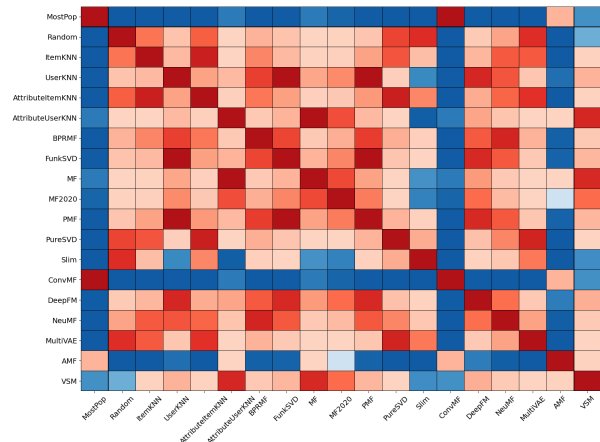
Figure 5.8: REC-ST<sub>Gender</sub> for all SDMs on GR<sub>Ch</sub>. The white dots indicate the average score.

### 5.2.2. Race Stereotypes

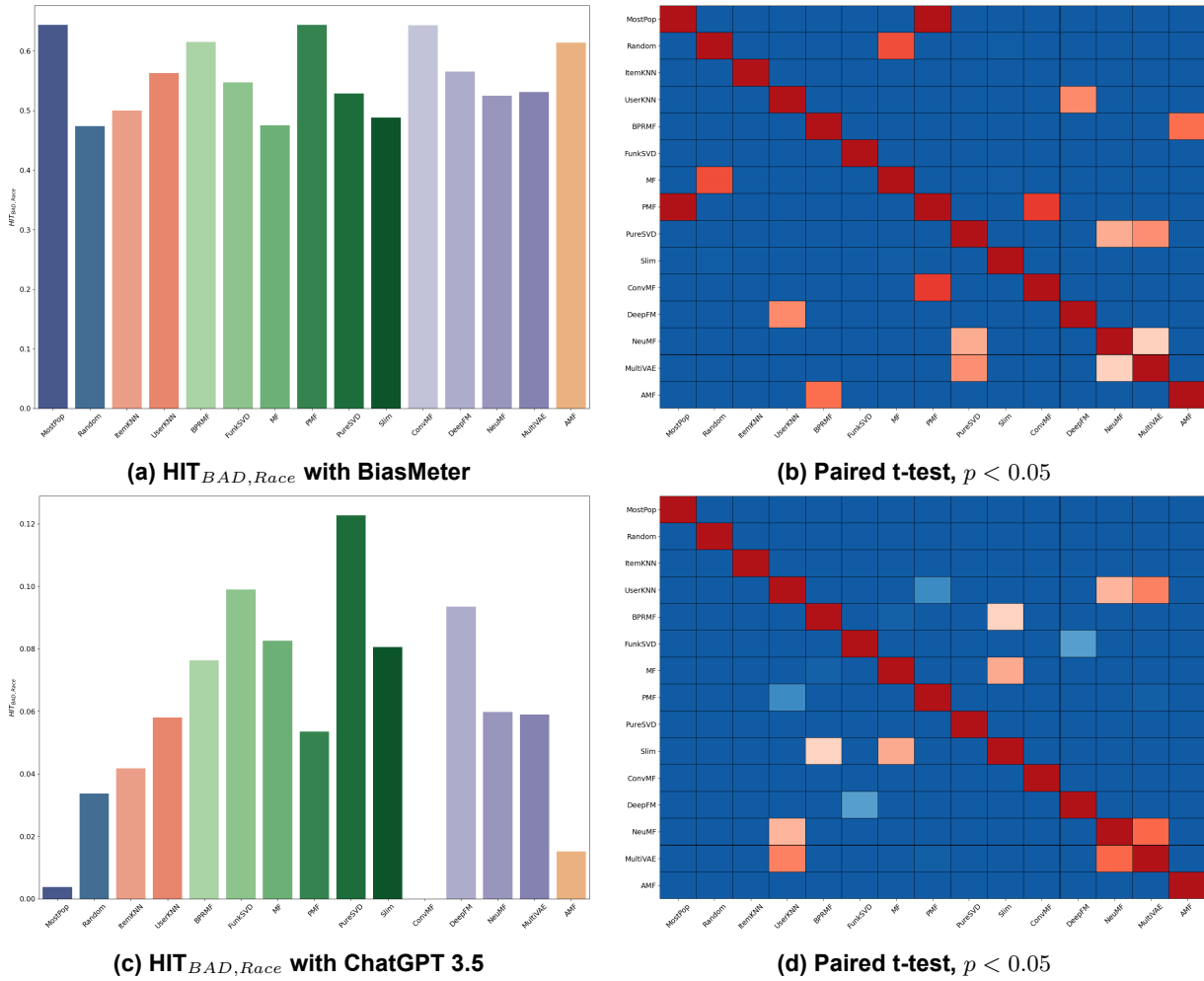
We present HIT<sub>BAD,Race</sub> results for BiasMeter and ChatGPT 3.5 in Figure 5.9 for ML<sub>Ch,S</sub> and Figure 5.10 for GR<sub>Ch</sub>.<sup>2</sup>

<sup>2</sup>Recall that NGIM was specifically designed for gender stereotypes. Therefore, we do not have results for race and religion stereotypes using this approach.

Based on the results from BiasMeter, most of the RAs exhibit similar performance. MostPop achieves the lowest  $HIT_{BAD,Race}$  ( $= 0.22$ ) on  $ML_{Ch,S}$  and the highest  $HIT_{BAD,Race}$  ( $= 0.644$ ) on  $GR_{Ch}$ . The only exception is PMF on  $GR_{Ch}$ , which achieves the same score and is statistically insignificant compared to MostPop and ConvMF. ConvMF has the highest  $HIT_{BAD,Race}$  ( $= 0.683$ ) for  $ML_{Ch,S}$  (with a lot of insignificant comparisons) and is a close second on  $GR_{Ch}$  with  $HIT_{BAD,Race} = 0.643$ . Interestingly, when we compare the RAs based on BiasMeter and ChatGPT 3.5, ConvMF has a  $HIT_{BAD,Race}$  of 0 for both datasets. Although the  $HIT_{BAD,Race}$  differences among RAs are marginal for ChatGPT 3.5, both MostPop and AMF mirror the trend exhibited by ConvMF across the datasets. Lastly, we see that a *latent factor model* is the worst in conveying stereotypical items to users across both datasets with SLIM having  $HIT_{BAD,Race} = 0.142$  on  $ML_{Ch,S}$  and PureSVD having an  $HIT_{BAD,Race} = 0.123$  on  $GR_{Ch}$ . More than half of the RAs compared to Slim are statistically non-significant. PureSVD is non-significant compared to NeuMF and MultiVAE.

(a)  $HIT_{BAD,Race}$  with BiasMeter(b) Paired t-test,  $p < 0.05$ (c)  $HIT_{BAD,Race}$  with ChatGPT 3.5(d) Paired t-test,  $p < 0.05$ 

**Figure 5.9:**  $HIT_{BAD,Race}$  for BiasMeter and ChatGPT 3.5 on  $ML_{Ch,S}$ . The white dots indicate the average score.



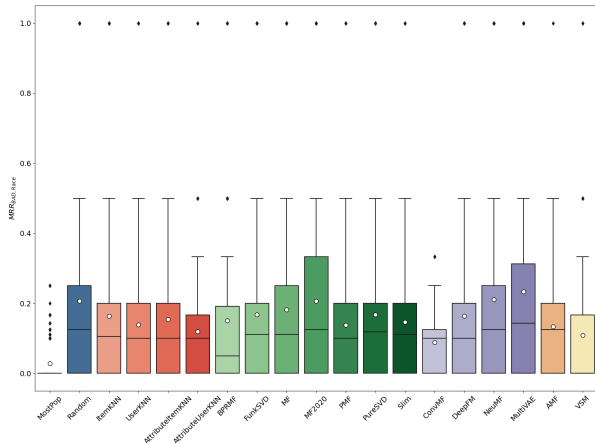
**Figure 5.10:** HIT<sub>BAD,Race</sub> for BiasMeter and ChatGPT 3.5 on GR<sub>Ch</sub>. The white dots indicate the average score.

As shown in The results for MRR<sub>BAD,Race</sub> are depicted in Figures 5.11 and 5.12 for ML<sub>Ch,S</sub> and GR<sub>Ch</sub> respectively.

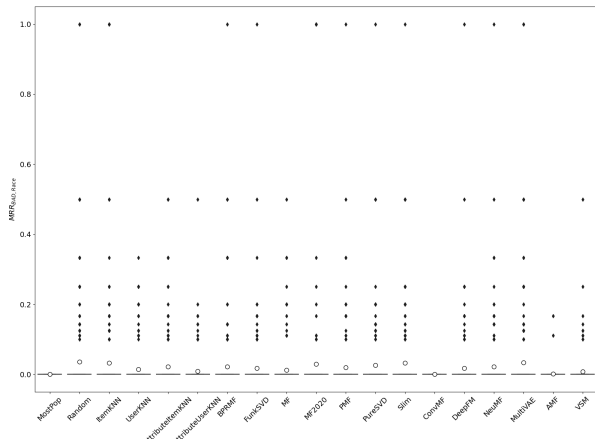
For ML<sub>Ch,S</sub>, most of the RAs achieve similar MRR<sub>BAD,Race</sub> scores based on BiasMeter. The MostPop algorithm has the lowest MRR<sub>BAD,Race</sub> with most of the users not getting stereotypical items at all. MultiVAE shows the highest MRR<sub>BAD,Race</sub> ( $= 0.233$ ). This is non-significant compared to Random, FunkSVD, MF, MF2020, and NeuMF. The outliers indicate that all RAs placed the first item at the top of the list for some users, with the exception of MostPop and ConvMF.

Results for GR<sub>Ch</sub> indicate that MostPop has a potential uniform distribution, due to all the results being contained in the entire box. Other RAs worth mentioning are BPR-MF, ConvMF, and AMF, achieving higher MRR<sub>BAD,Race</sub> than their peers.

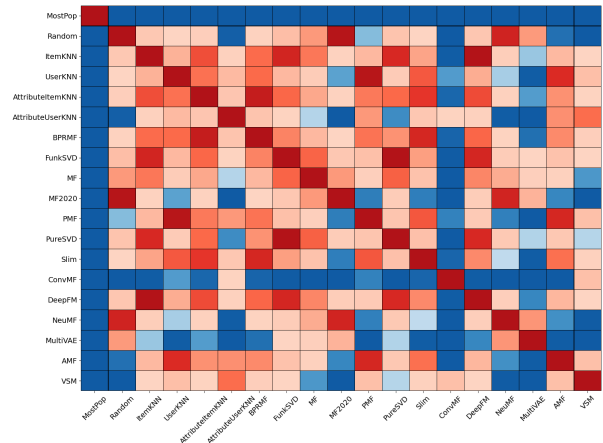
All the RAs have low MRR<sub>BAD,Race</sub> according to ChatGPT 3.5 on both datasets. However, the outliers suggest that still users are presented with items containing stereotypes.



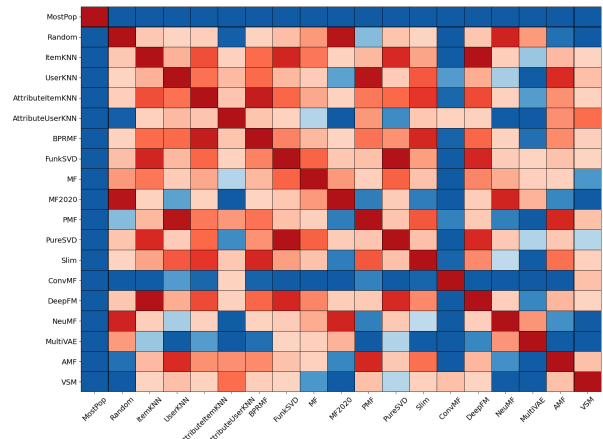
(a)  $MRR_{BAD,Race}$  with BiasMeter



(c)  $MRR_{BAD,Race}$  with ChatGPT 3.5

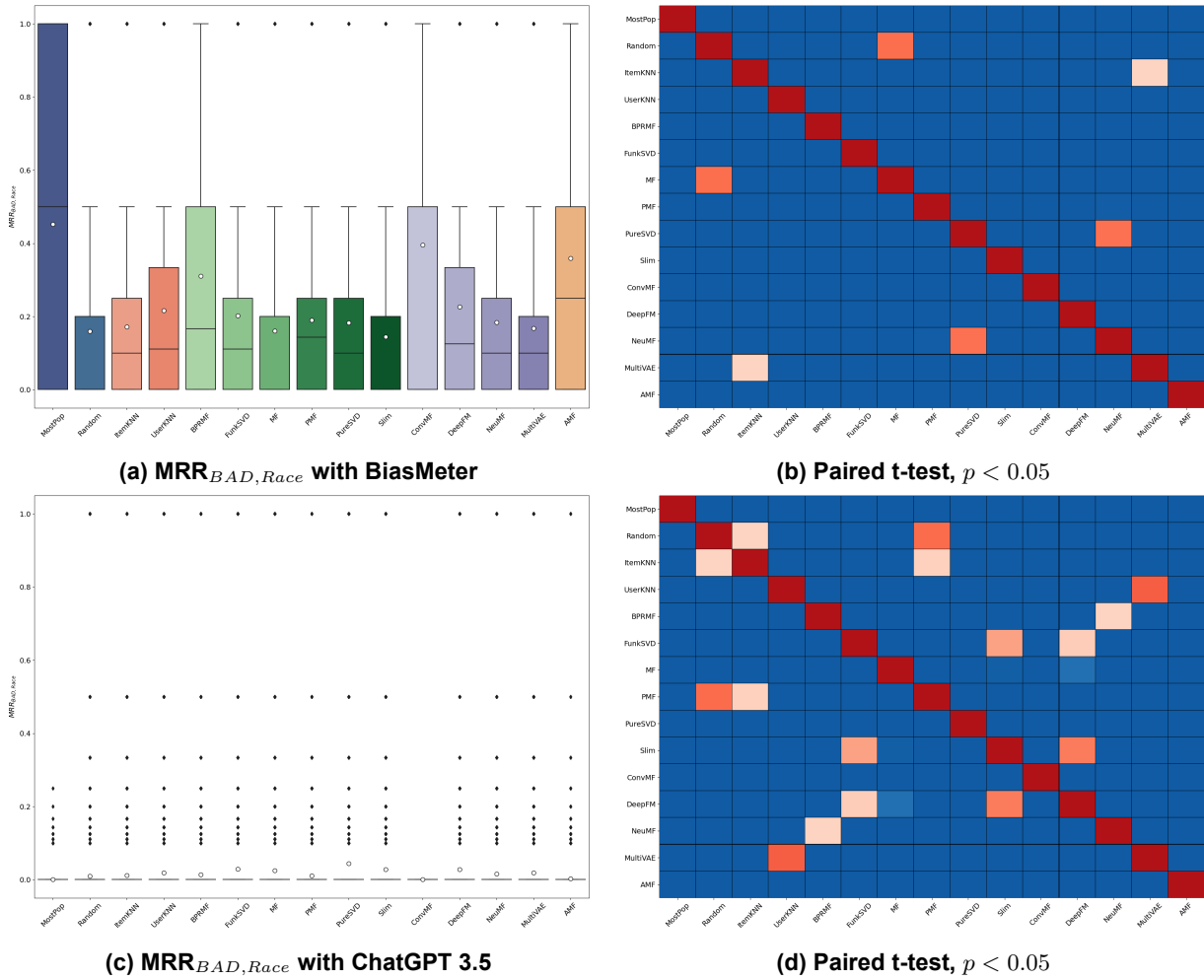


(b) Paired t-test,  $p < 0.05$



(d) Paired t-test,  $p < 0.05$

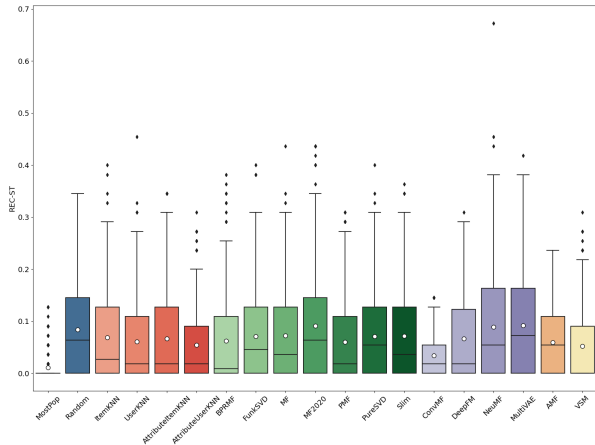
Figure 5.11:  $MRR_{BAD,Race}$  for BiasMeter and ChatGPT 3.5 on  $ML_{Ch,S}$ . The white dots indicate the average score.



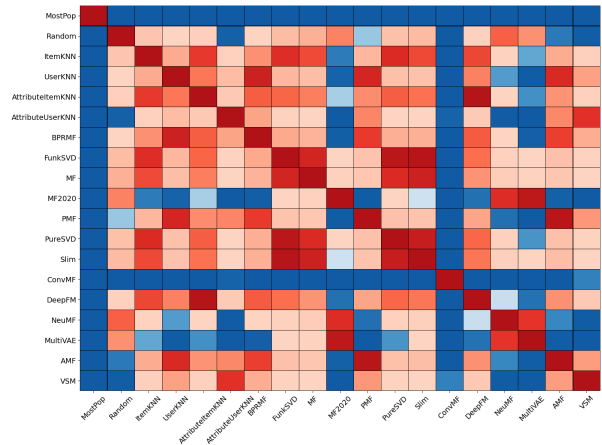
**Figure 5.12:**  $MRR_{BAD,Race}$  for BiasMeter and ChatGPT 3.5 on  $GR_{Ch}$ . The white dots indicate the average score.

The  $REC-ST_{Race}$  results depicted in Figures 5.13 and 5.14 for  $ML_{Ch,S}$  and  $GR_{Ch}$  show similar trends when compared to  $MRR_{BAD,Race}$ . However, the differences in  $REC-ST_{Race}$  scores are more nuanced.

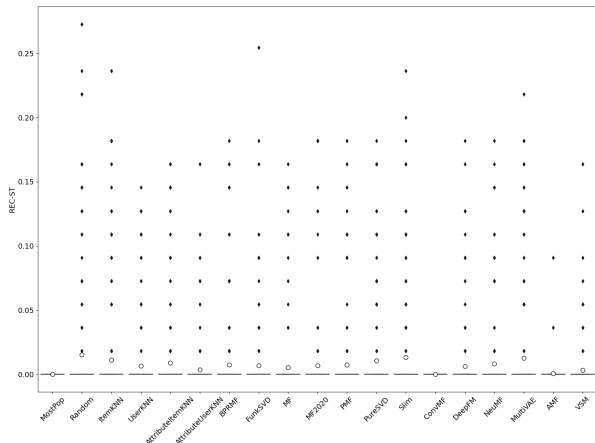




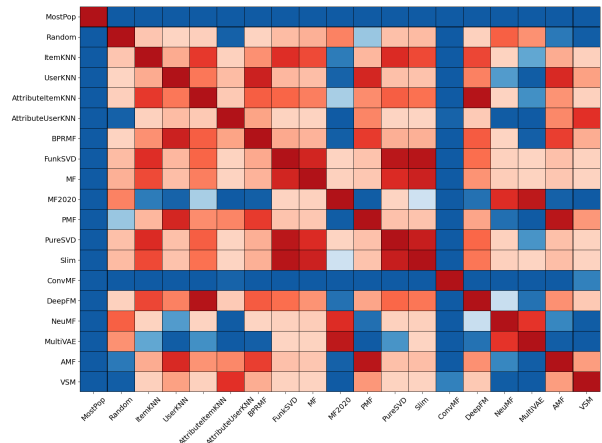
(a) REC-ST<sub>Race</sub> with BiasMeter



(b) Paired t-test,  $p < 0.05$

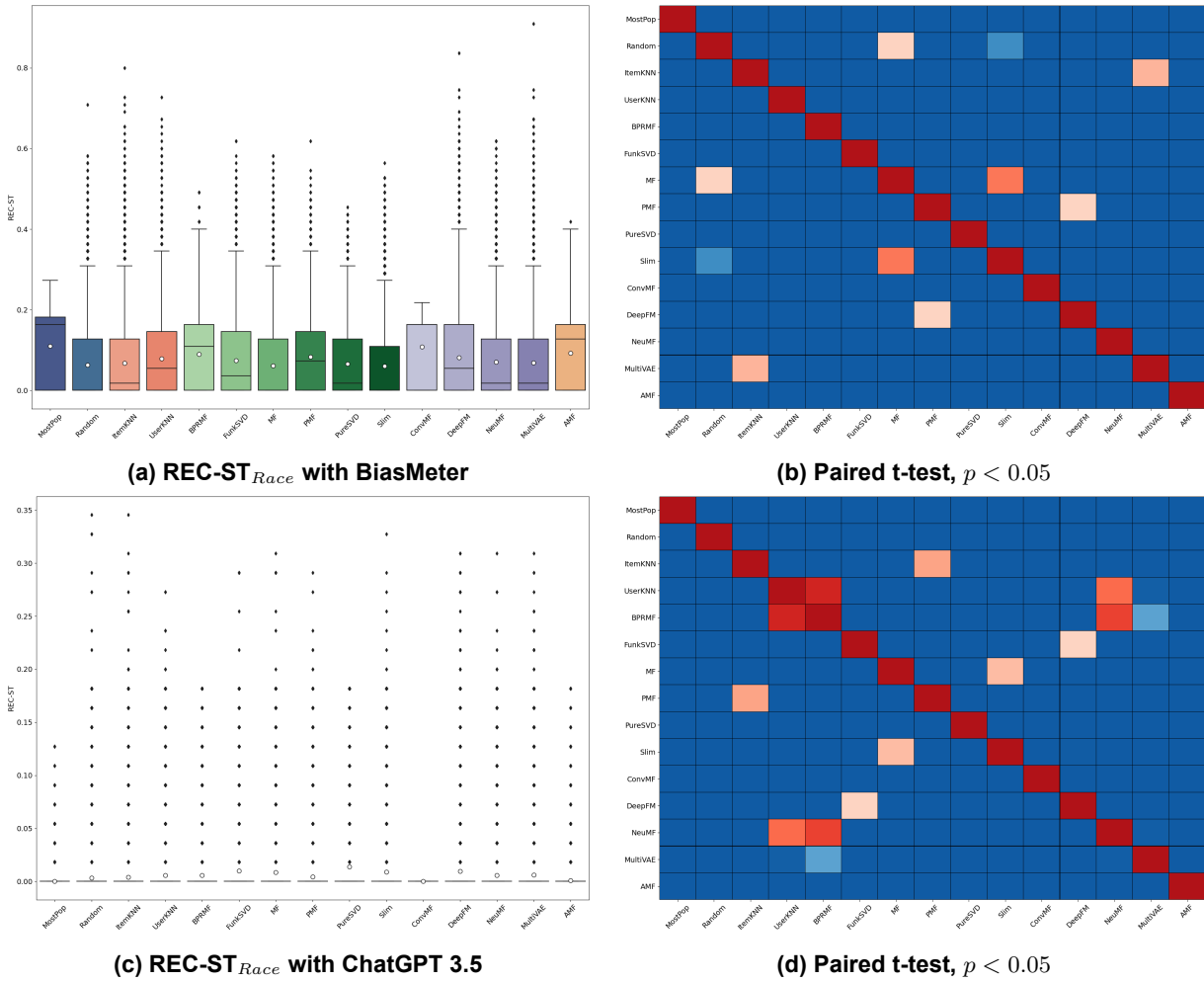


(c) REC-ST<sub>Race</sub> with ChatGPT 3.5



(d) Paired t-test,  $p < 0.05$

Figure 5.13: REC-ST<sub>Race</sub> for BiasMeter and ChatGPT 3.5 on  $ML_{Ch,S}$ . The white dots indicate the average score.

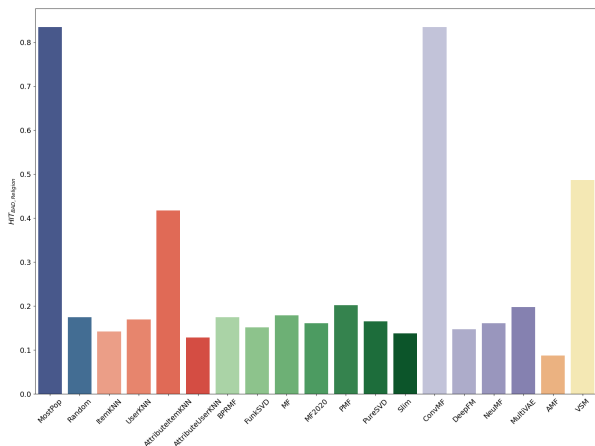


**Figure 5.14:** REC-ST<sub>Race</sub> for BiasMeter and ChatGPT 3.5 on GR<sub>Ch</sub>. The white dots indicate the average score.

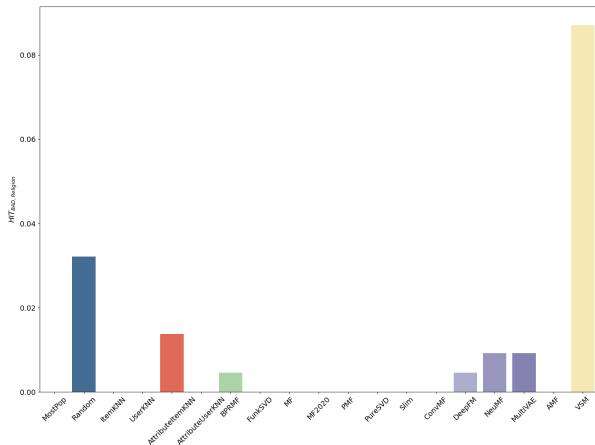
### 5.2.3. Religion Stereotypes

Religion stereotypes are present the least compared to gender and race for both SDMs as depicted in Table 5.3. Nevertheless, we see significant differences across all RAs. In Figure 5.15 we observe that MostPop and ConvMF have the highest  $HIT_{BAD,Religion}$  ( $= 0.835$ ), followed by VSM ( $= 0.486$ ) and AttributeItemKNN ( $= 0.417$ ) for BiasMeter. Differences between other RAs are marginal. Only 7 out of the 19 RAs suggest stereotypical items based on ChatGPT 3.5. VSM has the highest  $HIT_{BAD,Religion}$  ( $= 0.087$ ). This result is only statistically non-significant with the Random RA.

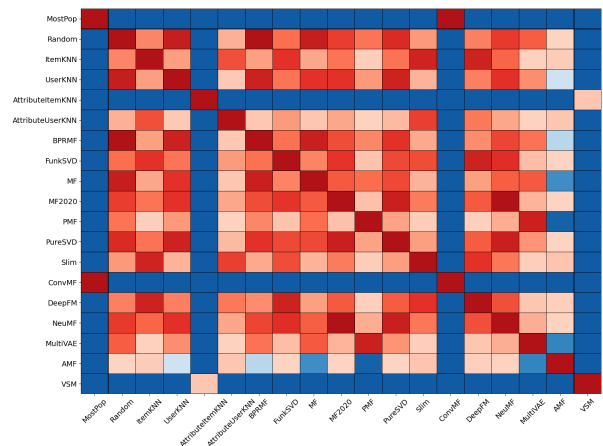
In Figure 5.16, we observe similar trends for BiasMeter results on GR<sub>Ch</sub>. However, MostPop appears now to have the lowest  $HIT_{BAD,Religion}$  ( $= 0.035$ ), and ConvMF is yet again the worst performing RA ( $= 0.885$ ). Furthermore, PMF has the lowest  $HIT_{BAD,Religion}$  ( $= 0.078$ ) as a personalized RA. With ChatGPT 3.5, the differences are very small, with DeepFM being the worst-performing personalized RA ( $HIT_{BAD,Religion} = 0.0015$ ). This result is statistically non-significant compared to MF and MultiVAE.



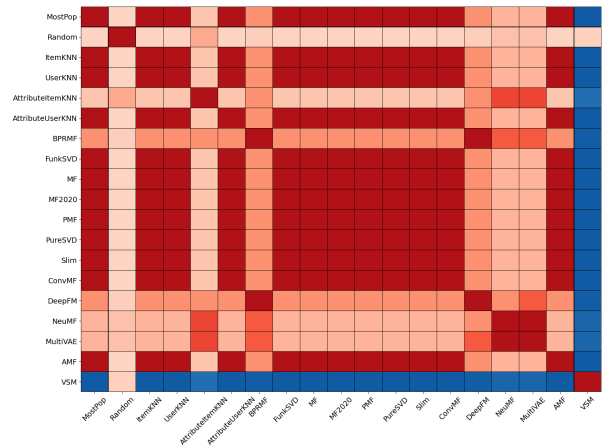
(a)  $HIT_{BAD,Religion}$  with BiasMeter



(c)  $HIT_{BAD,Religion}$  with ChatGPT 3.5

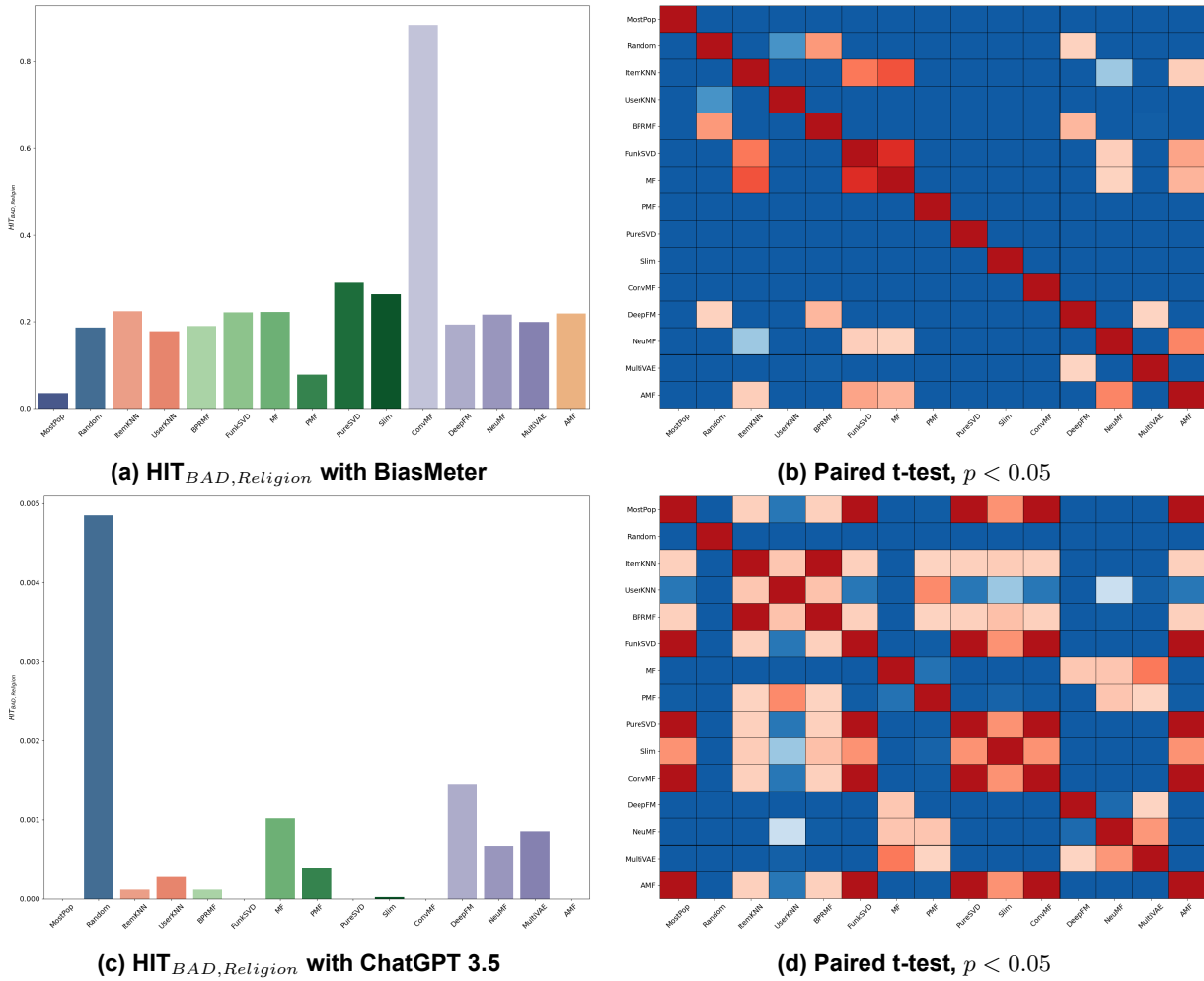


(b) Paired t-test,  $p < 0.05$



(d) Paired t-test,  $p < 0.05$

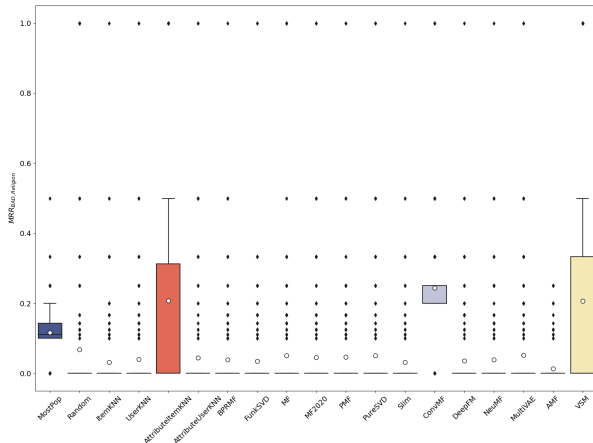
**Figure 5.15:**  $HIT_{BAD,Religion}$  for BiasMeter and ChatGPT 3.5 on  $ML_{Ch,S}$ . The white dots indicate the average score.



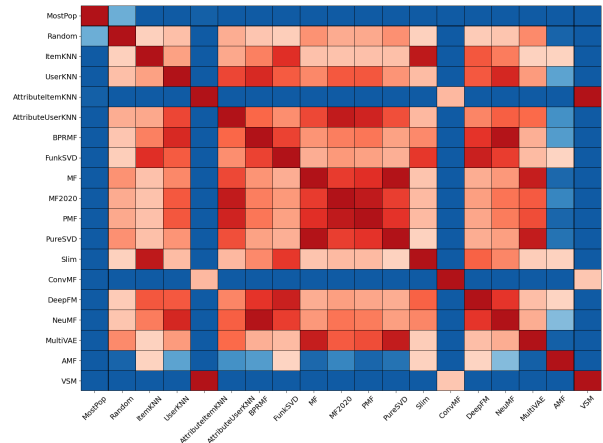
**Figure 5.16:**  $HIT_{BAD,Religion}$  for BiasMeter and ChatGPT 3.5 on  $GR_{Ch}$ . The white dots indicate the average score.

Even with the low values of  $HIT_{BAD,Religion}$ , we see that some users are still seeing items at the top of the recommendations, as illustrated in Figures 5.17 and 5.18. Turning our attention to  $ML_{Ch,S}$ , we observe for BiasMeter that `AttributemkNN`, `VSM`, and `ConvMF` are the personalized RAs that have a higher  $MRR_{BAD,Religion}$  compared to the other RAs. However, the outliers indicate that all RAs have shown movies containing religion stereotypes at the top of the list, with the exception of `MostPop` and `AMF`. Furthermore, for `AMF`, the highest position of a movie with stereotypes is at the 4<sup>th</sup> place in the Top-10 recommendation list. We see similar trends with  $GR_{Ch}$ , however, next to `ConvMF`, the outstanding RAs are `PureSVD` and `Slim`.

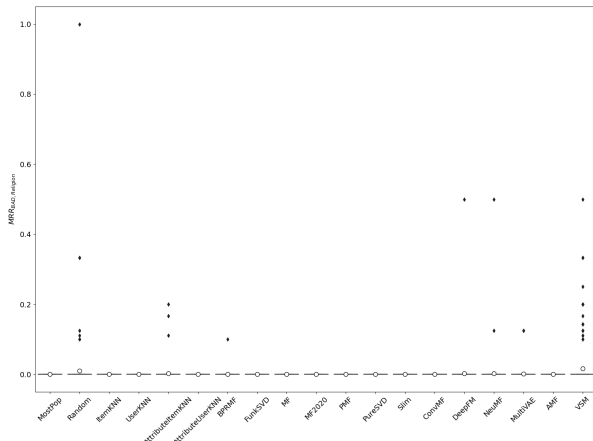
Through ChatGPT 3.5's lens, we see low  $MRR_{BAD,Religion}$  scores. However, with the low amounts of items containing a religion stereotype ( $ML_{Ch,S} = 17$  and  $GR_{Ch} = 8$  depicted in Table 5.3), it is concerning that there are still RAs where the first item that contains a stereotype is positioned at or near the top of the recommendation list.



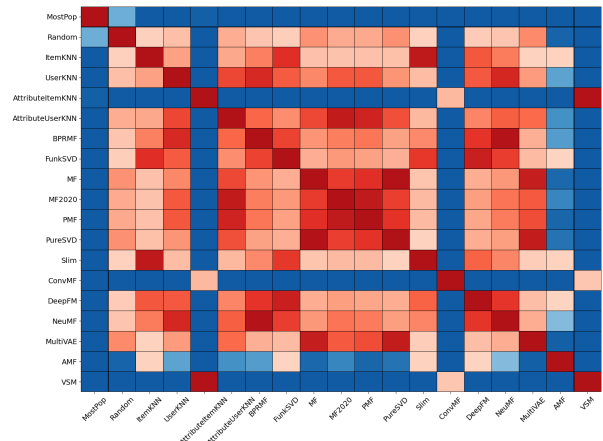
(a)  $MRR_{BAD,Religion}$  with BiasMeter



(b) Paired t-test,  $p < 0.05$

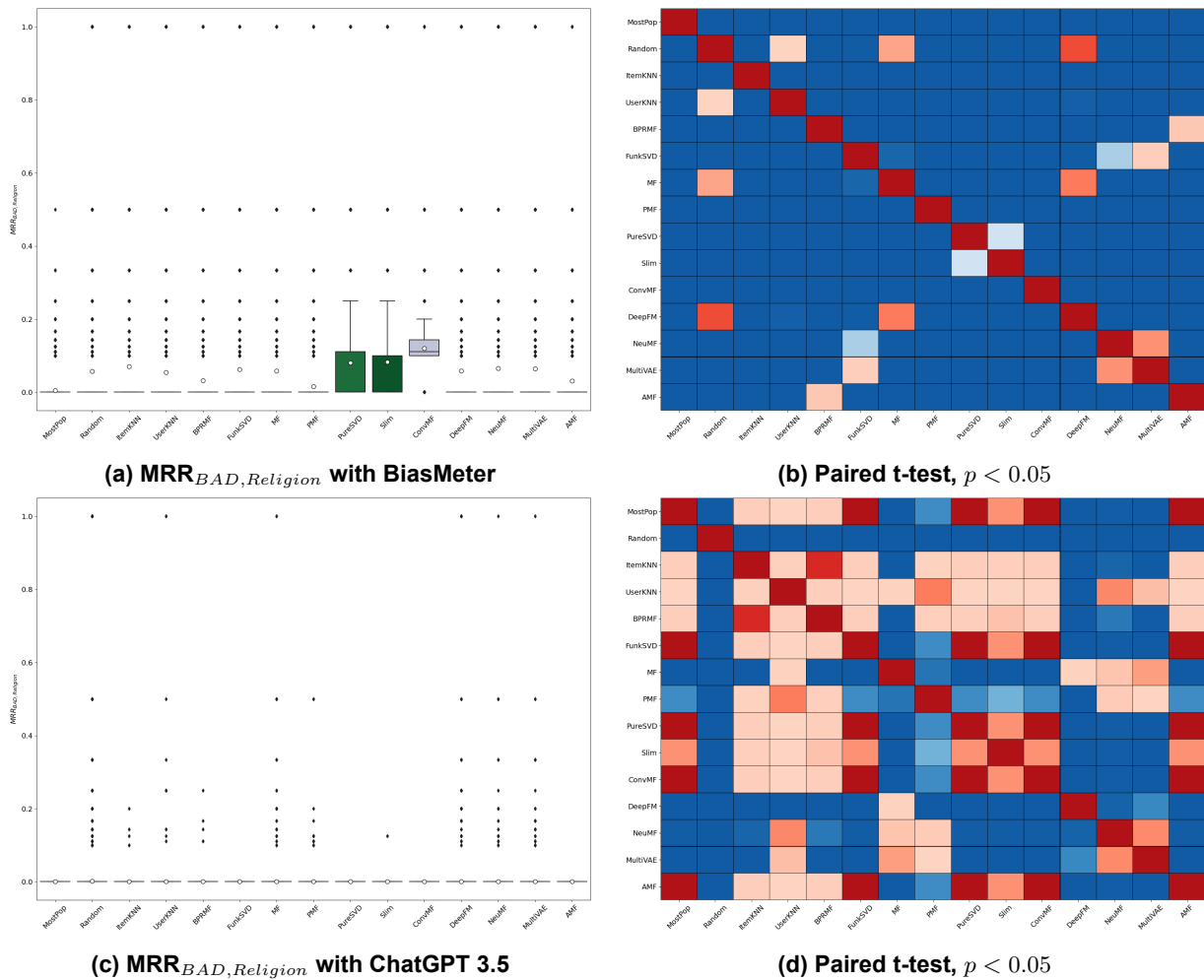


(c)  $MRR_{BAD,Religion}$  with ChatGPT 3.5



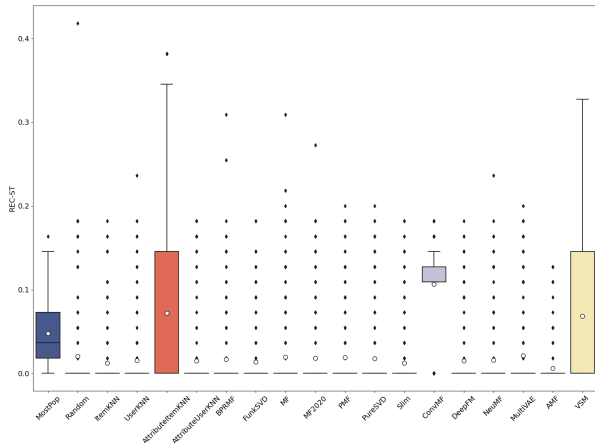
(d) Paired t-test,  $p < 0.05$

Figure 5.17:  $MRR_{BAD,Religion}$  for BiasMeter and ChatGPT 3.5 on  $ML_{Ch,S}$ . The white dots indicate the average score.

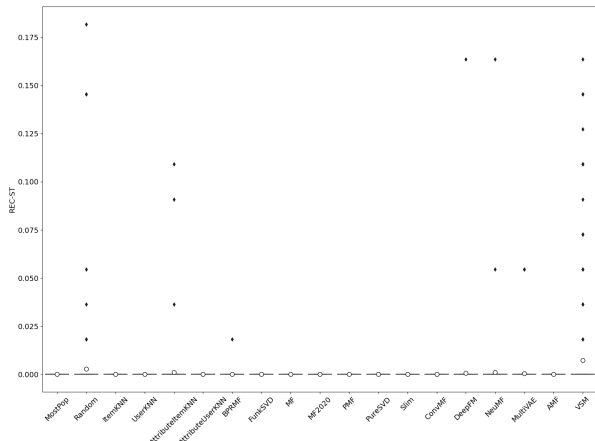


**Figure 5.18:**  $MRR_{BAD,Religion}$  for BiasMeter and ChatGPT 3.5 on  $GR_{Ch}$ . The white dots indicate the average score.

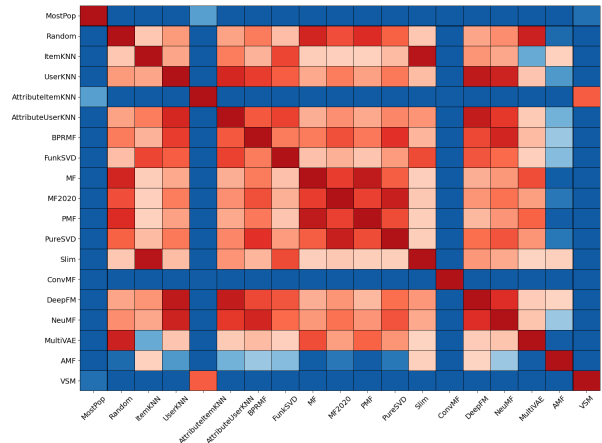
Similar to racial stereotypes, we see a recurring trend for  $REC-ST_{Religion}$  compared with  $MRR_{BAD,Religion}$  for both stereotypes. Interesting to note is the difference of the  $REC-ST_{Religion}$  scores from BiasMeter between  $ML_{Ch,S}$  (Fig. 5.19) and  $GR_{Ch}$  (Fig. 5.20). While the average  $REC-ST_{Religion}$  is similar for both datasets, we observe higher and simultaneously many more outliers for  $GR_{Ch}$  than for  $ML_{Ch,S}$ . One outlier got even close to  $REC-ST_{Religion}$  of 1 for DeepFM, meaning that a user got almost only books containing religious stereotypes in their recommendations.



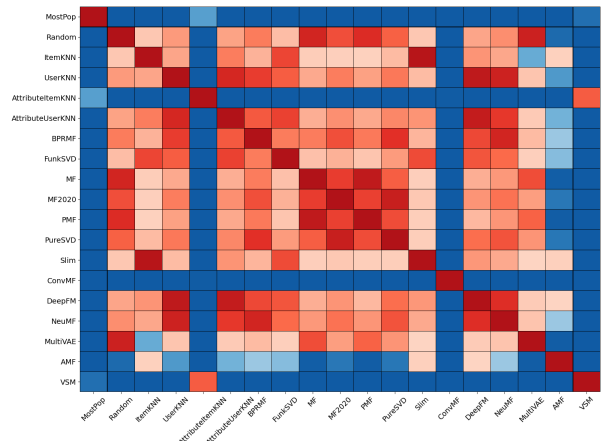
(a) REC-ST<sub>Religion</sub> with BiasMeter



(c) REC-ST<sub>Religion</sub> with ChatGPT 3.5

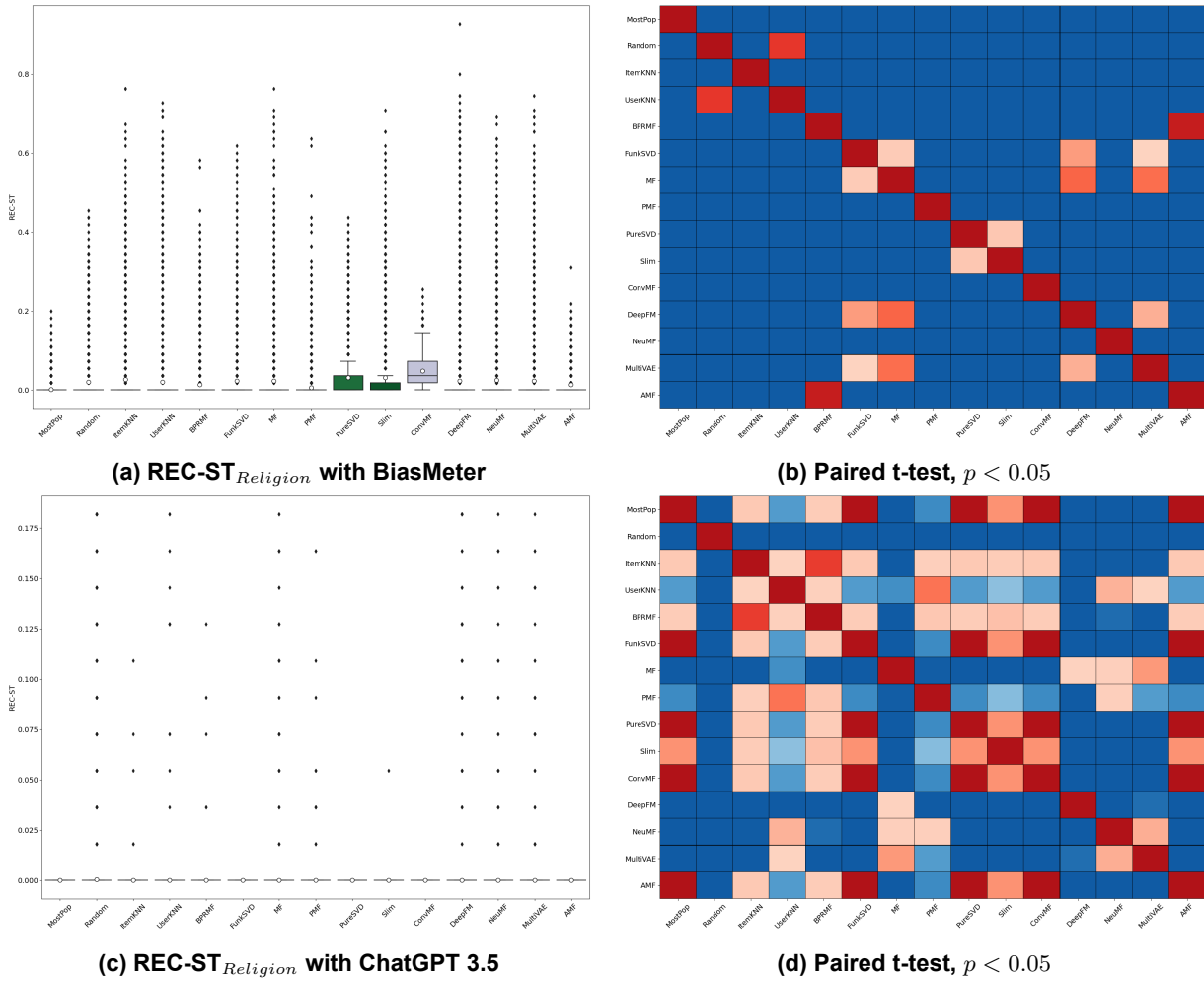


(b) Paired t-test,  $p < 0.05$



(d) Paired t-test,  $p < 0.05$

Figure 5.19: REC-ST<sub>Religion</sub> for BiasMeter and ChatGPT 3.5 on  $ML_{Ch,S}$ . The white dots indicate the average score.



**Figure 5.20:** REC-ST<sub>Religion</sub> for BiasMeter and ChatGPT 3.5 on GR<sub>Ch</sub>. The white dots indicate the average score.

## 5.3. Discussion

With the results from Experiments 1 and 2 (presented in Sections 5.1 and 5.2), we are able to assess the performances of the RAs and juxtapose this with their stereotype prominence and exposure in their suggestions to children. We discuss and highlight salient RAs based on the trade-offs between their performance and stereotype exposure.

We find that ConvMF is among the least effective in terms of performance, with  $nDCG_{ML} = 0.1563$  and  $nDCG_{Gr} = 0.1113$ , while also frequently exposing users to stereotypes the most. Interestingly, when looking at  $HIT_{BAD,Gender}$  of ChatGPT 3.5, we observe that ConvMF exposes users the most to movies with stereotypes, yet the least to books with stereotypes. However, for race and religion stereotypes, ConvMF does not appear to suggest any items containing stereotypes for both datasets. A potential reason for the consistent underperformance of ConvMF could be its implementation in Elliot. This implementation, based on the work by Kim et al. [47], allows ConvMF to extract context from documents. However, this feature seems to be missing in Elliot’s implementation. Additionally, we see the same behavior of ConvMF in MostPop and AMF as demonstrated in Figure 5.10. Furthermore, we also notice that RAs requiring side-information such as AttributItemKNN and VSM are also among the worst-performing models in terms of performance and stereotype prominence. Making these RAs the worst contenders to consider when both high performance and low stereotype prominence are preferred.

Models that have high performance, such as MF2020 ( $nDCG_{ML} = 0.2696$ ) and SLIM ( $nDCG_{ML} = 0.2693$ ), overall behave similarly to their peers. However, we notice that SLIM has the highest  $HIT_{BAD,Race}$



(= 0.142) based on ChatGPT 3.5 when recommending movies. This means that while it is one of the best RAs in terms of recommending relevant items, movies with racial stereotypes reach more children compared to other RAs. Furthermore, we observe that MF2020 has a significantly lower  $HIT_{BAD,Race}$  (= 0.064) than SLIM, making it the most suitable RA when one is concerned with high performance and low stereotype prominence for race. If even lower racial stereotype prominence is preferred, then MostPop, AttributeUserkNN, ConvMF, AMF, and VSM could be considered at the cost of less accurate relevant recommendations.

ItemkNN would be the best option for  $GR_{Ch}$  according to ChatGPT 3.5. It has the lowest  $HIT_{BAD,Race}$  with approximately 4% of the users getting books recommended containing racial stereotypes while being the second best RA in suggesting relevant items ( $nDCG_{GR} = 0.3307$ ). Again, if a lower  $HIT_{BAD,Race}$  is preferred, MostPop, ConvMF, and AMF can be considered at the cost of performance.

Upon analyzing results from BiasMeter, we observe that for *latent factor models*, MF2020 has the second highest  $HIT_{BAD,Race}$  after PureSVD for  $ML_{Ch,S}$ . Despite this, it simultaneously has the highest  $MRR_{BAD,Race}$  and  $REC-ST_{Race}$ , indicating that the items are ranked in positions that are detrimental to children compared to other RAs. MultiVAE follows this same trend for *neural models*.

Turning our attention to religious stereotypes, we observed that VSM is significantly worse than most other RAs for both SDM strategies. The high stereotype prominence paired with bad performance in recommending relevant items makes VSM the worst choice for this specific stereotype.

From the reported results and aforementioned salient trends, we argue that the outcomes are concerning. While RAs are designed to improve user experience by offering relevant content, our analysis suggests that they inadvertently expose children to biased and potentially harmful stereotypes. What especially is concerning is even with the low amount of books labeled as stereotypical by BiasMeter (931) and ChatGPT 3.5 (50) from the total of 14935, we still observe that a lot of users get at least one item containing a stereotype. For example, for ChatGPT 3.5 only 0.33% of all the items of  $GR_{Ch}$  contain racial stereotypes, yet 12.3% of the users have gotten at least one stereotypical item recommended by PureSVD.

Children aged between 3 and 4 years old become aware of ethnicity and gender, begin sorting themselves and others by ethnicity, and develop personal stereotypes [61]. Furthermore, children develop *stereotype consciousness*—the awareness of other people’s stereotypes—between the ages of 5 and 11 [60]. This awareness also increases with the age of children and by early adolescence, most children have developed knowledge of broadly held stereotypes [60]. Moreover, children from academically stigmatized ethnic groups (African Americans and Latinos) are more likely to be aware of broadly held stereotypes at all ages compared to children from academically nonstigmatized ethnic groups (Whites and Asians) [61].

Stereotype consciousness can influence a child’s academic performance through negative teacher expectancy. In this scenario, a child may worry that their test performance will be evaluated by their teacher based on prevailing stereotypes related to their race or ethnicity [60, 61]. This occurs frequently for children from stigmatized ethnic groups.

The link between stereotype consciousness and academic performance is not only limited to racial and ethnic stereotypes but can also be observed with gender stereotypes. For example, traditional gender stereotypes have been linked to less interest in STEM (Science, Technology, Engineering, and Mathematics) fields for girls [59, 94]. Furthermore, media has a role in shaping children’s careers. For instance, children who played a digital learning game about science had stronger interest and motivation for STEM fields, depending on the age of the children and whether the character they played was female or male [94]. Findings underscore the importance of representation in media content, for example, when children see a character they can identify with—such as a female scientist in a game—this can positively influence the child’s perception and interest in those fields [94].

Stereotypes about religion also carry negative consequences. Brown et al. [13] find that American children between the ages of 6 and 11 hold stereotypes against Arab Muslim males, perceiving them as anti-American and hostile, and Arab Muslim females, perceiving them as more oppressed compared to others. Only in 5% of films are Muslims depicted as “average” people, and on television, 81% of Muslims are depicted as terrorists. These stereotypes held by the children are consistent with the portrayal of Arab Muslims in the media [13].

Considering the influence of stereotypes on children’s cognitive and social development, it is crucial to analyze how RAs might be contributing to this phenomenon. Children who are in the developmental stages

of their lives, may not question stereotypes exposed to them by recommendations and start to hold these beliefs. If items containing stereotypes start to get interactions, they may be exacerbated by RAs due to their nature of recommending items based on interactions—especially with MostPop, which is still used in large platforms such as Netflix.

For example, when a girl is constantly recommended content that promotes traditional gender stereotypes, she may develop a skewed perspective of her potential career paths. Consider the book *Heaven to Betsy*, detected as a stereotypical item by all three SDM's, and its description: *"High School is Heaven! It's Betsy Ray's freshman year at Deep Valley High School, and she and her best childhood chum, Tacy Kelly, are loving every minute. Betsy and Tacy find themselves in the midst of a new crowd of friends, with studies aplenty (including Latin and—ugh—algebra), parties and picnics galore, Sunday night lunches at home—and boys! There's Cab Edwards, the jolly boy next door; handsome Herbert Humphreys; and the mysteriously unfriendly, but maddeningly attractive, Joe Willard. Betsy likes them all, but no boy, in particular, catches her fancy until she meets the new boy in town, Tony Markham . . . the one she and Tacy call the Tall Dark Handsome Stranger. He's sophisticated, funny, and dashing—and treats Betsy just like a sister. Can Betsy turn him into a beau?"* While describing Betsy's experiences, the passage's mention of "Latin and—ugh—algebra" might suggest a stereotype that girls find certain academic subjects, like mathematics, tiresome or challenging. When books with descriptions like these are suggested to children, especially girls, this might impact them negatively over time by endorsing such stereotypes.

Such content can inadvertently limit a child's imagination and ambition, restricting them to predefined societal gender roles instead of encouraging them to explore a range of possibilities. Children who do not conform to traditional gender roles might experience anxiety, depression, or feelings of isolation [94]. Therefore, it is important for RAs to be designed and tuned with sensitivity to these potential stereotypes, to ensure that they promote a diverse and enriching experience for users of all ages.

#### **Answer to the Research Question**

Based on the results presented in Section 5.1, 5.2 and the discussion above, RAs in general do suggest stereotypes to children. Table 5.3 showed that gender stereotypes are the most prominent in both datasets, followed by race and a very low value for religion stereotypes. What is interesting is the fact that even with very low amounts of items containing stereotypes, they can be quite prominent in recommendations of RAs and sometimes even at the top of the Top-10 list.

Given these findings, we issue a call to action for researchers and practitioners to critically evaluate their RAs, examining the prominence of stereotypes within their systems. Such reflection is essential to ensure that the content that is recommended is both relevant and not harmful, especially when catering to impressionable audiences like children. Furthermore, from our analysis, there is no specific RA or category of RAs that does not recommend stereotypes, making this a challenging problem to solve, since there is no existing RA that can serve as a reference for mitigating stereotypes from suggestions.

Large platforms such as Netflix, YouTube, and TikTok are also encouraged to assess their RAs, especially since they have content tailored to children. For example, a recent study highlighted concerns regarding the promotion of sexualized content to children on TikTok, where hypersexualized behaviors were observed in a majority of videos from popular accounts [88]. Such exposure can have harmful effects on children's well-being and academic performance. Specifically, exposure to traditional media content emphasizing women's appearance or sexual appeal has been associated with decreased academic performance or weaker interest in pursuing certain careers [94].

# Part III

## Closure

# 6

## Conclusion

In this manuscript, we discuss the outcomes of the empirical analysis we conducted to explore stereotype prominence in Top-10 recommendations made by RAs to children. We used performance metrics to contextualize an extensive suite of RAs and three different SDMs to explore stereotype prominence through different lenses. Specifically, we used a naive approach, BiasMeter, and ChatGPT 3.5, leveraging three different stereotype metrics to gain different insights into stereotype prominence.

Results from our analysis indicate that well-known and widely used RAs do suggest content containing stereotypes to children. The extent of stereotypes in these recommendations can vary significantly across different RAs, yet none have demonstrated resilience to suggesting stereotypical content. However, there are scenarios where some RAs are preferred to use in terms of performance and stereotype prominence. For example, ItemkNN is the second best model in terms of nDCG on  $GR_{Ch}$  and has relatively the lowest stereotype prominence for  $HIT_{BAD, Race}$ .

To the best of our knowledge, our work presents a novel analysis in an area largely unexplored in existing literature. Apart from recent, and very preliminary explorations [74, 75], no other research has delved into this subject. We are the first to examine a wide variety of RAs and analyze their recommendations for the presence of stereotypes, specifically in the context of content suggested to children, given the potential harm stereotypes can cause to this audience.

As with any research study, we identify some limitations. Specifically, we encountered issues with the reproducibility of BiasMeter, as the provided code did not produce results or function as described in the original research paper. We addressed this issue by making necessary adjustments to the code. Additionally, we undertook manual checks to ensure that our findings were consistent with the expectations based on the original BiasMeter research paper. These steps were necessary to ensure that our adaptations of the tool did not deviate from its intended functionality and that the integrity of our analysis was maintained.

Another limitation is related to the RAs used in our study that require side information. Models such as VSM and AttributeItemkNN are not utilized to their full potential. This is because Elliot supports only categorical item descriptions, like genre or author, as side information. It does not allow for more detailed and varied data, such as movie or book plot summaries, which are often used to enhance recommendations.

An additional limitation is with regard to our choice of metric for hyperparameter tuning. In this study, we use nDCG to optimize the RAs. While nDCG is a widely accepted measure, different results might have been observed had we selected and optimized based on another metric. Thus, the findings presented here should be interpreted with this consideration in mind.

In this work, we consider only two datasets, ML and GR, that are from different domains and among the most common datasets used in the recommender systems literature. Doing so, allows us to explore stereotype presence in items with different metadata in two domains popular among children: books and movies [29, 85].

Furthermore, we consider a vast amount of RAs from different categories. These RAs span from common baselines to state-of-the-art models in the literature. There exist many more RAs in Elliot's suite that potentially offer more insights. However, these are excluded because they require metadata that is beyond the scope of this work.

The results regarding stereotype prominence should be interpreted while keeping the limitations and biases of the SDMs used in mind. For instance, BiasMeter tends to be highly sensitive to false positives when detecting gender stereotypes, which could potentially exaggerate the prominence of gender stereotypes in the suggestions made by RAs. To overcome such limitations to an extent, we employ different strategies, leveraging the strengths of different state-of-the-art models to help us identify the stereotype presence in descriptions of movies and books.

Nevertheless, these aforementioned choices enable us to make a strong foundation for future work, creating many new research paths to explore. Our work can be expanded by considering different perspectives of movies and books. For instance, instead of solely focusing on textual elements, we can also examine visual aspects. This could involve analyzing book covers from  $GR_{Ch}$  to explore potential stereotypes depicted in the images. Additionally, we could investigate these book covers with Visual Recommenders, which make recommendations based on images, and see how books are recommended compared to the RAs in this work. Focusing on visual aspects, such as book covers, is particularly relevant as children are often attracted to and influenced by visual attributes [8].

Other types of multimedia recommenders, such as music recommenders, could also be considered. Music has been shown to significantly impact the lives and development of children [39]. The LastFM dataset [82] might help explore stereotypes in music. It contains user listening histories and demographic information, such as age, from the music streaming service Last.fm. This dataset could serve as a starting point to explore potential stereotypes in music tailored to children by analyzing the lyrics of the songs.

Further research could build upon our findings to uncover hidden trends related to the prevalence of stereotypes in these suggestions. For example, one could explore whether certain genres of movies display more stereotypes than others. Additionally, it would be interesting to investigate if stereotypes are suggested more frequently to girls as compared to boys, and vice versa. Examining whether there are noticeable differences in stereotype exposure among children from diverse ethnic or racial backgrounds could also provide valuable insights. Moreover, it is interesting to explore how stereotypes considered in this work or different ones are portrayed with different SDMs. For example, for ChatGPT 3.5 we only considered gender, race, and religion stereotypes. However, ChatGPT 3.5 was also able to detect age, beauty, and sexual orientation stereotypes.

Findings from our work may also motivate researchers and practitioners to seek opinions from experts specialized in stereotypes. These experts could assess the presence of stereotypes in movies from ML and books from  $GR_{Ch}$ . This could potentially lead to the creation of a dataset that can be used in the future to evaluate the prominence of stereotypes in the suggestions of RAs, taking a step closer to mitigating stereotypes from suggestions.

The empirical analysis we conducted has potential applications beyond recommendation systems and could be extended to the field of Information Retrieval (IR). Specifically, Search Engine Results Pages tailored to children could be analyzed to determine whether they inadvertently expose children to stereotypes.

Our work also impacts other domains where recommenders are used, such as education. Kollmayer et al. [49] mention the presence of gender stereotypes in school textbooks, focusing on the implicit stereotypes portrayed in textbook imagery. Are such stereotypes prominent in suggestions made by book recommenders for school teaching materials?

On top of that, findings from our work call for the need for multidisciplinary efforts that bring together insights from computer science, sociology, psychology, and Human-Computer Interaction to discuss the next steps for designing RAs more robust to stereotypes. The aim should be to expand on inclusion, ensuring that recommendations cater to everyone and that all users, regardless of their background, feel represented. These discussions should lean towards a human-centered technology design, such as principles like Human-Centered AI and Human-Centered IR, which prioritize the creation of systems that are beneficial, respectful, and tailored to human needs. Embracing such principles may lead to the birth of RAs where children are placed at the center and treated with ethical considerations, rather than being seen merely as userIDs interacting with items.

# Bibliography

- [1] Allen, G., Peterson, B.L., Ratakonda, D.k., Sakib, M.N., Fails, J.A., Kennington, C., Wright, K.L., Pera, M.S.: Engage!: Co-designing Search Engine Result Pages to Foster Interactions. In: Proceedings of the 20th Annual ACM Interaction Design and Children Conference, pp. 583–587, IDC '21, Association for Computing Machinery, New York, NY, USA (Jun 2021), ISBN 978-1-4503-8452-0, doi:10.1145/3459990.3465183, URL <https://dl.acm.org/doi/10.1145/3459990.3465183>
- [2] Anelli, V.W., Bellogin, A., Ferrara, A., Malitesta, D., Merra, F.A., Pomo, C., Donini, F.M., Di Noia, T.: Elliot: A Comprehensive and Rigorous Framework for Reproducible Recommender Systems Evaluation. In: Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 2405–2414, SIGIR '21, Association for Computing Machinery, New York, NY, USA (Jul 2021), ISBN 978-1-4503-8037-9, doi:10.1145/3404835.3463245, URL <https://dl.acm.org/doi/10.1145/3404835.3463245>
- [3] Anelli, V.W., Bellogin, A., Di Noia, T., Jannach, D., Pomo, C.: Top-N Recommendation Algorithms: A Quest for the State-of-the-Art. In: Proceedings of the 30th ACM Conference on User Modeling, Adaptation and Personalization, pp. 121–131, UMAP '22, Association for Computing Machinery, New York, NY, USA (Jul 2022), ISBN 978-1-4503-9207-5, doi:10.1145/3503252.3531292, URL <https://dl.acm.org/doi/10.1145/3503252.3531292>
- [4] Anelli, V.W., Di Noia, T., Di Sciascio, E., Ferrara, A., Mancino, A.C.M.: Sparse Feature Factorization for Recommender Systems with Knowledge Graphs. In: Proceedings of the 15th ACM Conference on Recommender Systems, pp. 154–165, RecSys '21, Association for Computing Machinery, New York, NY, USA (Sep 2021), ISBN 978-1-4503-8458-2, doi:10.1145/3460231.3474243, URL <https://dl.acm.org/doi/10.1145/3460231.3474243>
- [5] Aronson, J., Good, C.: The development and consequences of stereotype vulnerability in adolescents. In: Adolescence and education, Information Age Publishing (2002)
- [6] Backstrom, L., Leskovec, J.: Supervised random walks: predicting and recommending links in social networks. In: Proceedings of the fourth ACM international conference on Web search and data mining, pp. 635–644, WSDM '11, Association for Computing Machinery, New York, NY, USA (Feb 2011), ISBN 978-1-4503-0493-1, doi:10.1145/1935826.1935914, URL <https://dl.acm.org/doi/10.1145/1935826.1935914>
- [7] Bergstra, J., Bardenet, R., Bengio, Y., Kégl, B.: Algorithms for Hyper-Parameter Optimization. In: Advances in Neural Information Processing Systems, vol. 24, Curran Associates, Inc. (2011), URL [https://proceedings.neurips.cc/paper\\_files/paper/2011/hash/86e8f7ab32cfd12577bc2619bc635690-Abstract.html](https://proceedings.neurips.cc/paper_files/paper/2011/hash/86e8f7ab32cfd12577bc2619bc635690-Abstract.html)
- [8] Beyhan, Y., Pera, M.S.: Covering Covers: Characterization Of Visual Elements Regarding Sleeves. In: Adjunct Proceedings of the 31st ACM Conference on User Modeling, Adaptation and Personalization, pp. 28–33, UMAP '23 Adjunct, Association for Computing Machinery, New York, NY, USA (Jun 2023), ISBN 978-1-4503-9891-6, doi:10.1145/3563359.3597404, URL <https://dl.acm.org/doi/10.1145/3563359.3597404>
- [9] Bigler, R.S., Liben, L.S.: A developmental intergroup theory of social stereotypes and prejudice. In: Kail, R.V. (ed.) Advances in Child Development and Behavior, vol. 34, pp. 39–89, JAI (Jan 2006), doi:10.1016/S0065-2407(06)80004-2, URL <https://www.sciencedirect.com/science/article/pii/S0065240706800042>
- [10] Bolukbasi, T., Chang, K.W., Zou, J.Y., Saligrama, V., Kalai, A.T.: Man is to Computer Programmer as Woman is to Homemaker? Debiasing Word Embeddings. In: Lee, D., Sugiyama, M., Luxburg, U., Guyon, I., Garnett, R. (eds.) Advances in Neural Information Processing Systems, vol. 29, Curran Associates, Inc. (2016), URL [https://proceedings.neurips.cc/paper\\_files/paper/2016/file/a486cd07e4ac3d270571622f4f316ec5-Paper.pdf](https://proceedings.neurips.cc/paper_files/paper/2016/file/a486cd07e4ac3d270571622f4f316ec5-Paper.pdf)



- [11] Bordalo, P., Coffman, K., Gennaioli, N., Shleifer, A.: Stereotypes\*. *The Quarterly Journal of Economics* **131**(4), 1753–1794 (Nov 2016), ISSN 0033-5533, doi:10.1093/qje/qjw029, URL <https://doi.org/10.1093/qje/qjw029>
- [12] Breiffeller, L., Ahn, E., Jurgens, D., Tsvetkov, Y.: Finding Microaggressions in the Wild: A Case for Locating Elusive Phenomena in Social Media Posts. In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 1664–1674, Association for Computational Linguistics, Hong Kong, China (Nov 2019), doi:10.18653/v1/D19-1176, URL <https://aclanthology.org/D19-1176>
- [13] Brown, C.S., Ali, H., Stone, E.A., Jewell, J.A.: U.S. Children’s Stereotypes and Prejudicial Attitudes toward Arab Muslims. *Analyses of Social Issues and Public Policy* **17**(1), 60–83 (2017), ISSN 1530-2415, doi:10.1111/asap.12129, URL <https://onlinelibrary.wiley.com/doi/abs/10.1111/asap.12129>, \_eprint: <https://onlinelibrary.wiley.com/doi/pdf/10.1111/asap.12129>
- [14] Charlesworth, T.E.S., Yang, V., Mann, T.C., Kurdi, B., Banaji, M.R.: Gender Stereotypes in Natural Language: Word Embeddings Show Robust Consistency Across Child and Adult Language Corpora of More Than 65 Million Words. *Psychological Science* **32**(2), 218–240 (Feb 2021), ISSN 0956-7976, doi:10.1177/0956797620963619, URL <https://doi.org/10.1177/0956797620963619>, publisher: SAGE Publications Inc
- [15] Chen, J., Dong, H., Wang, X., Feng, F., Wang, M., He, X.: Bias and Debias in Recommender System: A Survey and Future Directions. *ACM Transactions on Information Systems* **41**(3), 67:1–67:39 (Feb 2023), ISSN 1046-8188, doi:10.1145/3564284, URL <https://dl.acm.org/doi/10.1145/3564284>
- [16] Cremonesi, P., Koren, Y., Turrin, R.: Performance of recommender algorithms on top-n recommendation tasks. In: *Proceedings of the fourth ACM conference on Recommender systems*, pp. 39–46, ACM, Barcelona Spain (Sep 2010), ISBN 978-1-60558-906-0, doi:10.1145/1864708.1864721, URL <https://dl.acm.org/doi/10.1145/1864708.1864721>
- [17] Crocker, J., Major, B.: Social stigma and self-esteem: The self-protective properties of stigma. *Psychological Review* **96**, 608–630 (1989), ISSN 1939-1471, doi:10.1037/0033-295X.96.4.608, place: US Publisher: American Psychological Association
- [18] Cryan, J., Tang, S., Zhang, X., Metzger, M., Zheng, H., Zhao, B.Y.: Detecting Gender Stereotypes: Lexicon vs. Supervised Learning Methods. In: *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, pp. 1–11, CHI ’20, Association for Computing Machinery, New York, NY, USA (Apr 2020), ISBN 978-1-4503-6708-0, doi:10.1145/3313831.3376488, URL <https://dl.acm.org/doi/10.1145/3313831.3376488>
- [19] Cvencek, D., Meltzoff, A.N., Greenwald, A.G.: Math–Gender Stereotypes in Elementary School Children. *Child Development* **82**(3), 766–779 (2011), ISSN 1467-8624, doi:10.1111/j.1467-8624.2010.01529.x, URL <https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1467-8624.2010.01529.x>, \_eprint: <https://srcd.onlinelibrary.wiley.com/doi/pdf/10.1111/j.1467-8624.2010.01529.x>
- [20] De Pessemier, T., Dooms, S., Martens, L.: Comparison of group recommendation algorithms. *Multimedia Tools and Applications* **72**(3), 2497–2541 (Oct 2014), ISSN 1573-7721, doi:10.1007/s11042-013-1563-0, URL <https://doi.org/10.1007/s11042-013-1563-0>
- [21] Deldjoo, Y., Frà, C., Valla, M., Paladini, A., Anghileri, D., Tuncil, M.A., Garzotta, F., Cremonesi, P., others: Enhancing children’s experience with recommendation systems. In: *CEUR Workshop Proceedings*, pp. N–A (2017)
- [22] Deldjoo, Y., Noia, T.D., Merra, F.A.: A Survey on Adversarial Recommender Systems: From Attack/Defense Strategies to Generative Adversarial Networks. *ACM Computing Surveys* **54**(2), 35:1–35:38 (2021), ISSN 0360-0300, doi:10.1145/3439729, URL <https://dl.acm.org/doi/10.1145/3439729>
- [23] Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In: *Proceedings of NAACL-HLT*, pp. 4171–4186 (2019), doi:10.48550/arXiv.1810.04805, URL <http://arxiv.org/abs/1810.04805>
- [24] Di Noia, T., Mirizzi, R., Ostuni, V.C., Romito, D., Zanker, M.: Linked open data to support content-

- based recommender systems. In: Proceedings of the 8th International Conference on Semantic Systems, pp. 1–8, I-SEMANTICS '12, Association for Computing Machinery, New York, NY, USA (Sep 2012), ISBN 978-1-4503-1112-0, doi:10.1145/2362499.2362501, URL <https://dl.acm.org/doi/10.1145/2362499.2362501>
- [25] Ekstrand, M.: Challenges in evaluating recommendations for children. In: International Workshop on Children & Recommender Systems. Available at: [shorturl.at/osFV9](http://shorturl.at/osFV9) (2017)
- [26] Ekstrand, M.D., Das, A., Burke, R., Diaz, F.: Fairness in Recommender Systems. In: Ricci, F., Rokach, L., Shapira, B. (eds.) Recommender Systems Handbook, pp. 679–707, Springer US, New York, NY (2022), ISBN 978-1-07-162197-4, doi:10.1007/978-1-0716-2197-4\_18, URL [https://doi.org/10.1007/978-1-0716-2197-4\\_18](https://doi.org/10.1007/978-1-0716-2197-4_18)
- [27] Ekstrand, M.D., Tian, M., Azpiazu, I.M., Ekstrand, J.D., Anuyah, O., McNeill, D., Pera, M.S.: All The Cool Kids, How Do They Fit In?: Popularity and Demographic Biases in Recommender Evaluation and Effectiveness. In: Proceedings of the 1st Conference on Fairness, Accountability and Transparency, pp. 172–186, PMLR (Jan 2018), URL <https://proceedings.mlr.press/v81/ekstrand18b.html>, iSSN: 2640-3498
- [28] Faggioli, G., Dietz, L., Clarke, C.L.A., Demartini, G., Hagen, M., Hauff, C., Kando, N., Kanoulas, E., Potthast, M., Stein, B., Wachsmuth, H.: Perspectives on Large Language Models for Relevance Judgment. In: Proceedings of the 2023 ACM SIGIR International Conference on Theory of Information Retrieval, pp. 39–50, ICTIR '23, Association for Computing Machinery, New York, NY, USA (Aug 2023), doi:10.1145/3578337.3605136, URL <https://dl.acm.org/doi/10.1145/3578337.3605136>
- [29] Flood, A.: Children read more challenging books in lockdowns, data reveals. The Guardian (Apr 2021), ISSN 0261-3077, URL <https://www.theguardian.com/books/2021/apr/29/children-read-longer-more-challenging-books-in-lockdown>
- [30] Fokkens, A., Ruigrok, N., Beukeboom, C., Sarah, G., Van Atteveldt, W.: Studying muslim stereotyping through microportrait extraction. In: Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018) (2018)
- [31] Fraser, K.C., Kiritchenko, S., Nejadgholi, I.: Computational Modeling of Stereotype Content in Text. *Frontiers in Artificial Intelligence* **5** (2022), ISSN 2624-8212, URL <https://www.frontiersin.org/articles/10.3389/frai.2022.826207>
- [32] Funk, S.: Netflix Update: Try This at Home (2006), URL <https://sifter.org/~simon/journal/20061211.html>
- [33] Gaci, Y., Benatallah, B., Casati, F., Benabdeslem, K.: Masked Language Models as Stereotype Detectors? In: EDBT 2022, Edinburgh, United Kingdom (Mar 2022), URL <https://hal.science/hal-03626753>
- [34] Gantner, Z., Rendle, S., Freudenthaler, C., Schmidt-Thieme, L.: MyMediaLite: a free recommender system library. In: Proceedings of the fifth ACM conference on Recommender systems, pp. 305–308, RecSys '11, Association for Computing Machinery, New York, NY, USA (2011), ISBN 978-1-4503-0683-6, doi:10.1145/2043932.2043989, URL <https://dl.acm.org/doi/10.1145/2043932.2043989>
- [35] Gomez-Urbe, C.A., Hunt, N.: The Netflix Recommender System: Algorithms, Business Value, and Innovation. *ACM Transactions on Management Information Systems* **6**(4), 1–19 (Jan 2016), ISSN 2158-656X, 2158-6578, doi:10.1145/2843948, URL <https://dl.acm.org/doi/10.1145/2843948>
- [36] Gretzel, U., Fesenmaier, D.R.: Persuasion in Recommender Systems. *International Journal of Electronic Commerce* **11**(2), 81–100 (Dec 2006), ISSN 1086-4415, 1557-9301, doi:10.2753/JEC1086-4415110204, URL <https://www.tandfonline.com/doi/full/10.2753/JEC1086-4415110204>
- [37] Guo, H., Ruiming, T., Ye, Y., Li, Z., He, X.: DeepFM: A Factorization-Machine based Neural Network for CTR Prediction. In: Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, International Joint Conferences on Artificial Intelligence Organization (2017), URL <http://arxiv.org/abs/1703.04247>
- [38] Gómez Gutiérrez, E., Charisi, V., Chaudron, S.: Evaluating recommender systems with and for children: towards a multi-perspective framework. In: CEUR Workshop Proceedings. 2021; 2955., CEUR Workshop Proceedings (2021), URL <http://repositori.upf.edu/handle/10230/55665>, accepted: 2023-02-07T13:21:55Z Publisher: CEUR Workshop Proceedings



- [39] Hallam, S.: The power of music: Its impact on the intellectual, social and personal development of children and young people. *International Journal of Music Education* **28**(3), 269–289 (Aug 2010), ISSN 0255-7614, doi:10.1177/0255761410370658, URL <https://doi.org/10.1177/0255761410370658>, publisher: SAGE Publications Ltd
- [40] Hannon, J., Bennett, M., Smyth, B.: Recommending twitter users to follow using content and collaborative filtering approaches. In: *Proceedings of the fourth ACM conference on Recommender systems*, pp. 199–206, RecSys '10, Association for Computing Machinery, New York, NY, USA (Sep 2010), ISBN 978-1-60558-906-0, doi:10.1145/1864708.1864746, URL <https://dl.acm.org/doi/10.1145/1864708.1864746>
- [41] Harper, F.M., Konstan, J.A.: The MovieLens Datasets: History and Context. *ACM Transactions on Interactive Intelligent Systems* **5**(4), 19:1–19:19 (Dec 2015), ISSN 2160-6455, doi:10.1145/2827872, URL <https://doi.org/10.1145/2827872>
- [42] He, X., He, Z., Du, X., Chua, T.S.: Adversarial Personalized Ranking for Recommendation. In: *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, pp. 355–364 (Jun 2018), doi:10.1145/3209978.3209981, URL <http://arxiv.org/abs/1808.03908>, arXiv:1808.03908 [cs, stat]
- [43] He, X., Liao, L., Zhang, H., Nie, L., Hu, X., Chua, T.S.: Neural collaborative filtering. In: *Proceedings of the 26th international conference on world wide web*, pp. 173–182 (2017), URL <http://arxiv.org/abs/1708.05031>
- [44] Hussein, E., Juneja, P., Mitra, T.: Measuring Misinformation in Video Search Platforms: An Audit Study on YouTube. *Proceedings of the ACM on Human-Computer Interaction* **4**(CSCW1), 48:1–48:27 (2020), doi:10.1145/3392854, URL <https://dl.acm.org/doi/10.1145/3392854>
- [45] Joseph, K., Wei, W., Carley, K.M.: Girls Rule, Boys Drool: Extracting Semantic and Affective Stereotypes from Twitter. In: *Proceedings of the 2017 ACM Conference on Computer Supported Cooperative Work and Social Computing*, pp. 1362–1374, CSCW '17, Association for Computing Machinery, New York, NY, USA (Feb 2017), ISBN 978-1-4503-4335-0, doi:10.1145/2998181.2998187, URL <https://dl.acm.org/doi/10.1145/2998181.2998187>
- [46] Khan, M.M., Ibrahim, R., Ghani, I.: Cross Domain Recommender Systems: A Systematic Literature Review. *ACM Computing Surveys* **50**(3), 36:1–36:34 (Jun 2017), ISSN 0360-0300, doi:10.1145/3073565, URL <https://dl.acm.org/doi/10.1145/3073565>
- [47] Kim, D., Park, C., Oh, J., Lee, S., Yu, H.: Convolutional Matrix Factorization for Document Context-Aware Recommendation. In: *Proceedings of the 10th ACM Conference on Recommender Systems*, pp. 233–240, RecSys '16, Association for Computing Machinery, New York, NY, USA (Sep 2016), ISBN 978-1-4503-4035-9, doi:10.1145/2959100.2959165, URL <https://doi.org/10.1145/2959100.2959165>
- [48] Kingma, D.P., Welling, M.: An Introduction to Variational Autoencoders. *Foundations and Trends® in Machine Learning* **12**(4), 307–392 (Nov 2019), ISSN 1935-8237, 1935-8245, doi:10.1561/22000000056, URL <https://www.nowpublishers.com/article/Details/MAL-056>, publisher: Now Publishers, Inc.
- [49] Kollmayer, M., Schober, B., Spiel, C.: Gender stereotypes in education: Development, consequences, and interventions. *European Journal of Developmental Psychology* **15**(4), 361–377 (Jul 2018), ISSN 1740-5629, doi:10.1080/17405629.2016.1193483, URL <https://doi.org/10.1080/17405629.2016.1193483>, publisher: Routledge\_eprint: <https://doi.org/10.1080/17405629.2016.1193483>
- [50] Koren, Y., Bell, R., Volinsky, C.: Matrix Factorization Techniques for Recommender Systems. *Computer* **42**(8), 30–37 (Aug 2009), ISSN 1558-0814, doi:10.1109/MC.2009.263, conference Name: Computer
- [51] Kucirkova, N.: The Learning Value of Personalization in Children's Reading Recommendation Systems: What Can We Learn From Constructionism? *International Journal of Mobile and Blended Learning (IJMBL)* **11**(4), 80–95 (Oct 2019), ISSN 1941-8647, doi:10.4018/IJMBL.2019100106, URL <https://www.igi-global.com/article/the-learning-value-of-personalization-in-childrens-reading-recommendation-systems/www.igi-global.com/article/the-learning-value-of-personalization-in-childrens-reading-recommendation-systems/226976>, publisher: IGI Global

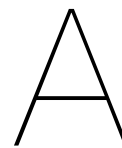
- [52] Levy, M., Jack, K.: Efficient top-n recommendation by linear regression. In: RecSys large scale recommender systems workshop (2013)
- [53] Liang, D., Krishnan, R.G., Hoffman, M.D., Jebara, T.: Variational Autoencoders for Collaborative Filtering. In: Proceedings of the 2018 World Wide Web Conference, pp. 689–698, WWW '18, International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, CHE (Apr 2018), ISBN 978-1-4503-5639-8, doi:10.1145/3178876.3186150, URL <https://dl.acm.org/doi/10.1145/3178876.3186150>
- [54] Linden, G., Smith, B., York, J.: Amazon.com recommendations: item-to-item collaborative filtering. *IEEE Internet Computing* **7**(1), 76–80 (Jan 2003), ISSN 1089-7801, doi:10.1109/MIC.2003.1167344, URL <http://ieeexplore.ieee.org/document/1167344/>
- [55] Livingstone, S.: Risk and harm on the internet. *Media and the well-being of children and adolescents* pp. 129–146 (2014)
- [56] Luger, G.F.: *Artificial Intelligence: Structures and Strategies for Complex Problem Solving*. Pearson Education (2005), ISBN 978-0-321-26318-6, google-Books-ID: QcTuJb7Hi40C
- [57] Lund, B.D., Wang, T.: Chatting about ChatGPT: how may AI and GPT impact academia and libraries? *Library Hi Tech News* **40**(3), 26–29 (Jan 2023), ISSN 0741-9058, doi:10.1108/LHTN-01-2023-0009, URL <https://doi.org/10.1108/LHTN-01-2023-0009>, publisher: Emerald Publishing Limited
- [58] Mansoury, M., Abdollahpouri, H., Pechenizkiy, M., Mobasher, B., Burke, R.: Feedback Loop and Bias Amplification in Recommender Systems. In: Proceedings of the 29th ACM International Conference on Information & Knowledge Management, pp. 2145–2148, CIKM '20, Association for Computing Machinery, New York, NY, USA (2020), ISBN 978-1-4503-6859-9, doi:10.1145/3340531.3412152, URL <https://dl.acm.org/doi/10.1145/3340531.3412152>
- [59] Master, A.: Gender Stereotypes Influence Children's STEM Motivation. *Child Development Perspectives* **15**(3), 203–210 (2021), ISSN 1750-8606, doi:10.1111/cdep.12424, URL <https://onlinelibrary.wiley.com/doi/abs/10.1111/cdep.12424>, \_eprint: <https://onlinelibrary.wiley.com/doi/pdf/10.1111/cdep.12424>
- [60] McKown, C., Strambler, M.J.: Developmental Antecedents and Social and Academic Consequences of Stereotype-Consciousness in Middle Childhood. *Child Development* **80**(6), 1643–1659 (2009), ISSN 1467-8624, doi:10.1111/j.1467-8624.2009.01359.x, URL <https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1467-8624.2009.01359.x>, \_eprint: <https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.1467-8624.2009.01359.x>
- [61] McKown, C., Weinstein, R.S.: The Development and Consequences of Stereotype Consciousness in Middle Childhood. *Child Development* **74**(2), 498–515 (2003), ISSN 0009-3920, URL <https://www.jstor.org/stable/3696327>, publisher: [Wiley, Society for Research in Child Development]
- [62] Millecamp, M., Htun, N.N., Jin, Y., Verbert, K.: Controlling Spotify Recommendations: Effects of Personal Characteristics on Music Recommender User Interfaces. In: Proceedings of the 26th Conference on User Modeling, Adaptation and Personalization, pp. 101–109, UMAP '18, Association for Computing Machinery, New York, NY, USA (Jul 2018), ISBN 978-1-4503-5589-6, doi:10.1145/3209219.3209223, URL <https://dl.acm.org/doi/10.1145/3209219.3209223>
- [63] Mnih, A., Salakhutdinov, R.R.: Probabilistic Matrix Factorization. In: *Advances in Neural Information Processing Systems*, vol. 20, Curran Associates, Inc. (2007), URL <https://papers.nips.cc/paper/2007/hash/d7322ed717dedf1eb4e6e52a37ea7bcd-Abstract.html>
- [64] Murgia, E., Landoni, M., Huibers, T., Fails, J., Pera, M.: The Seven Layers of Complexity of Recommender Systems for Children in Educational Contexts. *CEUR Workshop Proceedings* (Jan 2019), URL [https://scholarworks.boisestate.edu/cs\\_facpubs/203](https://scholarworks.boisestate.edu/cs_facpubs/203)
- [65] Nadeem, M., Bethke, A., Reddy, S.: StereoSet: Measuring stereotypical bias in pretrained language models. In: Zong, C., Xia, F., Li, W., Navigli, R. (eds.) *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 5356–5371, Association for Computational Linguistics, Online (Aug 2021), doi:10.18653/v1/2021.acl-long.416, URL <https://aclanthology.org/2021.acl-long.416>

- [66] Nangia, N., Vania, C., Bhalerao, R., Bowman, S.R.: CrowS-Pairs: A challenge dataset for measuring social biases in masked language models. In: 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, pp. 1953–1967, Association for Computational Linguistics (ACL) (2020), URL <http://arxiv.org/abs/2010.00133>
- [67] Ning, X., Karypis, G.: SLIM: Sparse Linear Methods for Top-N Recommender Systems. In: 2011 IEEE 11th International Conference on Data Mining, pp. 497–506 (Dec 2011), doi:10.1109/ICDM.2011.134, ISSN: 2374-8486
- [68] Papadamou, K., Papasavva, A., Zannettou, S., Blackburn, J., Kourtellis, N., Leontiadis, I., Stringhini, G., Sirivianos, M.: Disturbed YouTube for Kids: Characterizing and Detecting Inappropriate Videos Targeting Young Children. Proceedings of the International AAAI Conference on Web and Social Media **14**, 522–533 (May 2020), ISSN 2334-0770, doi:10.1609/icwsm.v14i1.7320, URL <https://ojs.aaai.org/index.php/ICWSM/article/view/7320>
- [69] Pera, M., Murgia, E., Landoni, M., Huibers, T.: With a Little Help from My Friends: Use of Recommendations at School. Proceedings of ACM RecSys 2019 Late-Breaking Results: Co-Located with the 13th ACM Conference on Recommender Systems (RecSys 2019) (Jan 2019), URL [https://scholarworks.boisestate.edu/cs\\_facpubs/207](https://scholarworks.boisestate.edu/cs_facpubs/207)
- [70] Pera, M.S., Fails, J.A., Gelsomini, M., Garzotto, F.: Building Community: Report on KidRec Workshop on Children and Recommender Systems at RecSys 2017. ACM SIGIR Forum **52**(1), 153–161 (Aug 2018), ISSN 0163-5840, doi:10.1145/3274784.3274803, URL <https://doi.org/10.1145/3274784.3274803>
- [71] Pera, M.S., Ng, Y.K.: What to read next? making personalized book recommendations for K-12 users. In: Proceedings of the 7th ACM conference on Recommender systems, pp. 113–120, RecSys '13, Association for Computing Machinery, New York, NY, USA (2013), ISBN 978-1-4503-2409-0, doi:10.1145/2507157.2507181, URL <https://dl.acm.org/doi/10.1145/2507157.2507181>
- [72] Pera, M.S., Ng, Y.K.: Automating readers' advisory to make book recommendations for K-12 readers. In: Proceedings of the 8th ACM Conference on Recommender systems, pp. 9–16, ACM, Foster City, Silicon Valley California USA (Oct 2014), ISBN 978-1-4503-2668-1, doi:10.1145/2645710.2645721, URL <https://dl.acm.org/doi/10.1145/2645710.2645721>
- [73] Pizzato, L., Rej, T., Akehurst, J., Koprinska, I., Yacef, K., Kay, J.: Recommending people to people: the nature of reciprocal recommenders with a case study in online dating. User Modeling and User-Adapted Interaction **23**(5), 447–488 (Nov 2013), ISSN 1573-1391, doi:10.1007/s11257-012-9125-0, URL <https://doi.org/10.1007/s11257-012-9125-0>
- [74] Raj, A., Ekstrand, M.D.: Fire Dragon and Unicorn Princess: Gender Stereotypes and Children's Products in Search Engine Responses. In: Proceedings of ACM SIGIR Workshop on eCommerce, ACM (2022), URL [https://sigir-ecom.github.io/ecom22Papers/paper\\_4323.pdf](https://sigir-ecom.github.io/ecom22Papers/paper_4323.pdf)
- [75] Raj, A., Milton, A., Ekstrand, M.D.: Pink for Princesses, Blue for Superheroes: The Need to Examine Gender Stereotypes in Kid's Products in Search and Recommendations. KidRec '21: 5th International and Interdisciplinary Perspectives on Children & Recommender and Information Retrieval Systems (KidRec) Search and Recommendation Technology through the Lens of a Teacher- Co-located with ACM IDC 2021 (May 2021), doi:10.48550/arXiv.2105.09296, URL <https://arxiv.org/pdf/2105.09296.pdf>
- [76] Rendle, S., Freudenthaler, C., Gantner, Z., Schmidt-Thieme, L.: BPR: Bayesian personalized ranking from implicit feedback. In: Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence, pp. 452–461 (2009), URL <http://arxiv.org/abs/1205.2618>
- [77] Rendle, S., Krichene, W., Zhang, L., Anderson, J.: Neural collaborative filtering vs. matrix factorization revisited. In: Proceedings of the 14th ACM Conference on Recommender Systems, pp. 240–248 (2020), URL <http://arxiv.org/abs/2005.09683>
- [78] Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., Riedl, J.: GroupLens: an open architecture for collaborative filtering of netnews. In: Proceedings of the 1994 ACM conference on Computer supported cooperative work, pp. 175–186, CSCW '94, Association for Computing Machinery, New York, NY, USA (1994), ISBN 978-0-89791-689-9, doi:10.1145/192844.192905, URL <https://dl.acm.org/doi/10.1145/192844.192905>

- [79] Ricci, F., Rokach, L., Shapira, B. (eds.): *Recommender Systems Handbook*. Springer US, New York, NY (2022), ISBN 978-1-07-162196-7, doi:10.1007/978-1-0716-2197-4, URL <https://link.springer.com/10.1007/978-1-0716-2197-4>
- [80] Rudinger, R., May, C., Van Durme, B.: Social Bias in Elicited Natural Language Inferences. In: *Proceedings of the First ACL Workshop on Ethics in Natural Language Processing*, pp. 74–79, Association for Computational Linguistics, Valencia, Spain (Apr 2017), doi:10.18653/v1/W17-1609, URL <https://aclanthology.org/W17-1609>
- [81] Sap, M., Gabriel, S., Qin, L., Jurafsky, D., Smith, N.A., Choi, Y.: Social Bias Frames: Reasoning about Social and Power Implications of Language. In: *Association for Computational Linguistics (2020)*, URL <http://arxiv.org/abs/1911.03891>
- [82] Schedl, M.: The LFM-1b Dataset for Music Retrieval and Recommendation. In: *Proceedings of the 2016 ACM on International Conference on Multimedia Retrieval*, pp. 103–110, ICMR '16, Association for Computing Machinery, New York, NY, USA (Jun 2016), ISBN 978-1-4503-4359-6, doi:10.1145/2911996.2912004, URL <https://dl.acm.org/doi/10.1145/2911996.2912004>
- [83] Schwartz, B.: The Paradox of Choice. In: *Positive Psychology in Practice*, pp. 121–138, John Wiley & Sons, Ltd (2015), ISBN 978-1-118-99687-4, doi:10.1002/9781118996874.ch8, URL <https://onlinelibrary.wiley.com/doi/abs/10.1002/9781118996874.ch8>, section: 8 \_eprint: <https://onlinelibrary.wiley.com/doi/pdf/10.1002/9781118996874.ch8>
- [84] Seaver, N.: Captivating algorithms: Recommender systems as traps. *Journal of Material Culture* **24**(4), 421–436 (Dec 2019), ISSN 1359-1835, doi:10.1177/1359183518820366, URL <https://doi.org/10.1177/1359183518820366>, publisher: SAGE Publications Ltd
- [85] Seidelin, L.J., Hansen, M.K., Jensen, L.H.: How children consume film and series (Dec 2020), URL <https://www.dfi.dk/en/english/news/how-children-consume-film-and-series>
- [86] Shehzad, F., Jannach, D.: Everyone’s a Winner! On Hyperparameter Tuning of Recommendation Models. In: *Proceedings of the 17th ACM Conference on Recommender Systems*, pp. 652–657, RecSys '23, Association for Computing Machinery, New York, NY, USA (Sep 2023), doi:10.1145/3604915.3609488, URL <https://dl.acm.org/doi/10.1145/3604915.3609488>
- [87] Smith, B., Linden, G.: Two Decades of Recommender Systems at Amazon.com. *IEEE Internet Computing* **21**(3), 12–18 (May 2017), ISSN 1941-0131, doi:10.1109/MIC.2017.72, conference Name: IEEE Internet Computing
- [88] Soriano-Ayala, E., Bonillo Díaz, M., Cala, V.C.: TikTok and Child Hypersexualization: Analysis of Videos and Narratives of Minors. *American Journal of Sexuality Education* **18**(2), 210–230 (Apr 2023), ISSN 1554-6128, doi:10.1080/15546128.2022.2096734, URL <https://doi.org/10.1080/15546128.2022.2096734>, publisher: Taylor & Francis \_eprint: <https://doi.org/10.1080/15546128.2022.2096734>
- [89] Spear, L., Milton, A., Allen, G., Raj, A., Green, M., Ekstrand, M.D., Pera, M.S.: Baby Shark to Barracuda: Analyzing Children’s Music Listening Behavior. In: *Proceedings of the 15th ACM Conference on Recommender Systems*, pp. 639–644, RecSys '21, Association for Computing Machinery, New York, NY, USA (Sep 2021), ISBN 978-1-4503-8458-2, doi:10.1145/3460231.3478856, URL <https://dl.acm.org/doi/10.1145/3460231.3478856>
- [90] Tang, T.Y., Winoto, P.: I should not recommend it to you even if you will like it: the ethics of recommender systems. *New Review of Hypermedia and Multimedia* **22**(1-2), 111–138 (Jan 2016), ISSN 1361-4568, doi:10.1080/13614568.2015.1052099, URL <https://doi.org/10.1080/13614568.2015.1052099>, publisher: Taylor & Francis \_eprint: <https://doi.org/10.1080/13614568.2015.1052099>
- [91] Wan, M., McAuley, J.: Item recommendation on monotonic behavior chains. In: *Proceedings of the 12th ACM Conference on Recommender Systems*, pp. 86–94, RecSys '18, Association for Computing Machinery, New York, NY, USA (Sep 2018), ISBN 978-1-4503-5901-6, doi:10.1145/3240323.3240369, URL <https://dl.acm.org/doi/10.1145/3240323.3240369>
- [92] Wan, M., Misra, R., Nakashole, N., McAuley, J.: Fine-Grained Spoiler Detection from Large-Scale Review Corpora. In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 2605–2610, Association for Computational Linguistics, Florence, Italy (Jul 2019), doi:10.18653/v1/P19-1248, URL <https://aclanthology.org/P19-1248>

- [93] Wang, S., Cao, L., Wang, Y., Sheng, Q.Z., Orgun, M.A., Lian, D.: A Survey on Session-based Recommender Systems. *ACM Computing Surveys* **54**(7), 154:1–154:38 (Jul 2021), ISSN 0360-0300, doi:10.1145/3465401, URL <https://dl.acm.org/doi/10.1145/3465401>
- [94] Ward, L.M., Grower, P.: Media and the Development of Gender Role Stereotypes. *Annual Review of Developmental Psychology* **2**(1), 177–199 (2020), doi:10.1146/annurev-devpsych-051120-010630, URL <https://doi.org/10.1146/annurev-devpsych-051120-010630>, \_eprint: <https://doi.org/10.1146/annurev-devpsych-051120-010630>





# Supplementary Results and Figures

## A.1. Hyperparameter Optimization

Below we present the ranges we considered for the hyperparameters of each RA in Table A.1 and the resulting hyperparameter values are presented in Table A.2.

Algorithm	Hyperparameter	Range	Type	Distribution
<b>UserkNN, ItemkNN</b>	topK	5 - 1000	Integer	uniform
	similarity	cosine, jaccard, dice, pearson, euclidean	Categorical	
<b>AttributeUserkNN, AttributeItemkNN</b>	topK	5 - 1000	Integer	uniform
	similarity	cosine, dot, braycurtis, euclidean, chebyshev	Categorical	
<b>BPRMF</b>	num factors	8, 16, 32, 64, 128, 256	Integer	
	learning rate	0.00001 - 1	Real	log-uniform
	batch size	128, 256, 512	Integer	
	reg user	0.00001 - 0.1	Real	log-uniform
	reg positive item	0.00001 - 0.1	Real	log-uniform
	reg negative item	0.00001 - 0.1	Real	log-uniform
<b>FunkSVD</b>	num factors	10 - 100	Integer	uniform
	batch size	256, 512, 1024	Integer	
	learning rate	0.00001 - 1	Real	log-uniform
<b>MF</b>	num factors	8, 64, 128	Integer	
	batch size	256, 512, 1024	Integer	
	learning rate	0.00001 - 1	Real	log-uniform
	reg	0.00001 - 0.1	Real	log-uniform
<b>MF2020</b>	num factors	8, 16, 32, 64, 128	Integer	
	learning rate	0.00001 - 1	Real	log-uniform
	reg	0.00001 - 0.1	Real	log-uniform
	negative sample	4,6,8	Integer	

Table A.1 continued from previous page

Algorithm	Hyperparameter	Range	Type	Distribution
<b>PMF</b>	num factors	10 - 100	Integer	uniform
	batch size	256, 512, 1024	Integer	
	learning rate	0.00001 - 1	Real	log-uniform
<b>PureSVD</b>	num factors	10 - 100	Integer	uniform
<b>SLIM</b>	topK	5 - 1000	Integer	uniform
	l1 ratio	0.00001 - 1	Real	log-uniform
	alpha	0.01 - 1	Real	uniform
<b>ConvMF</b>	batch size	64, 128, 256	Integer	
	l_w	0.0001, 0.001, 0.0001, 0.00001	Real	
	l_b	0.01, 0.001, 0.0001, 0.00001	Real	
<b>DeepFM</b>	num factors	10 - 100	Integer	uniform
	batch size	256, 512, 1024	Integer	
	l_w	0.0001, 0.001, 0.0005, 0.1	Real	
<b>NeuMF</b>	num factors	8, 16, 32, 64, 128, 256	Integer	
	learning rate	0.00001 - 1	Real	log-uniform
	batch size	128, 256, 512	Integer	
	negative sample	4, 6, 8	Integer	
<b>MultiVAE</b>	learning rate	0.00001 - 1	Real	log-uniform
	batch size	64, 128, 256, 512	Integer	
	reg	0.00001 - 1	Real	log-uniform
<b>AMF</b>	num factors	8 - 32	Integer	uniform
	learning rate	0.01, 0.001, 0.0001	Real	
	eps	0.1 - 0.5	Real	uniform
<b>VSM</b>	similarity	cosine, correlation	Categorical	
	user profile	tfidf, binary	Categorical	
	item profile	tfidf, binary	Categorical	

Table A.1: Hyperparameter ranges used for RAs on both ML and Goodreads

Algorithm	Hyperparameter	MovieLens	GoodReads
<b>UserkNN</b>	topK	291	663
	similarity	cosine	cosine
<b>ItemkNN</b>	topK	200	9
	similarity	cosine	cosine
<b>AttributeUserkNN</b>	topK	529	-
	similarity	braycurtis	

Table A.2 continued from previous page

Algorithm	Hyperparameter	MovieLens	GoodReads
<b>AttributItemkNN</b>	topK	618	
	similarity	dot	-
<b>BPRMF</b>	num factors	256	32
	learning rate	0.0378936256	0.0210161
	batch size	256	256
	reg bias	0	0
	reg user	0.0157839	0.0014956
	reg positive item	0.0005651	6.7450451e-05
	reg negative item	0.0012779	1.6731978e-05
<b>FunkSVD</b>	num factors	91	80
	batch size	1024	1024
	learning rate	0.0001213	4.3556975e-05
	epochs	50	50
	reg_w	0.1	0.0143631
	reg_b	0.001	4.1867191e-05
<b>MF</b>	num factors	32	64
	batch size	1024	1024
	epochs	50	50
	learning rate	0.0003105	0.0003489
	reg	0.0065537	0.0021551
<b>MF2020</b>	num factors	32	
	epochs	256	
	learning rate	0.002	-
	reg	0.005	
	negative sample	8	
<b>PMF</b>	num factors	98	57
	batch size	256	256
	learning rate	0.0003531	0.0015493
	epochs	50	50
	reg	0.0025	0.0157839
	gaussian variance	0.1	0.1
<b>PureSVD</b>	num factors	22	10
<b>SLIM</b>	topK	542	203
	l1 ratio	0.0017297	1.4963653e-05
	alpha	0.2334318	0.1363217



Table A.2 continued from previous page

Algorithm	Hyperparameter	MovieLens	GoodReads
<b>ConvMF</b>	batch size	256	64
	epochs	50	25
	embedding size	64	64
	learning rate	0.001	0.001
	l_w	0.0001	0.005
	l_b	0.001	0.0005
	cnn_channels	(1, 32, 32)	(1, 32, 32)
	cnn_kernels	(2, 2)	(2, 2)
	cnn_strides	(2, 2)	(2, 2)
	dropout prob	0.3	0.3
<b>DeepFM</b>	num factors	100	50
	batch size	1024	512
	epochs	50	50
	learning rate	0.001	0.001
	l_w	0.001	0.001
	hidden neurons	(64, 32)	(64, 32)
	hidden activations	(relu, relu)	(relu, relu)
<b>NeuMF</b>	num factors	16	32
	learning rate	0.001	0.001
	batch size	256	1024
	epochs	20	20
	dropout	0	0
	negative sample	4	4
<b>MultiVAE</b>	intermediate dim	600	600
	latent dim	200	200
	learning rate	0.001	0.001
	batch size	128	512
	epochs	15	15
	dropout pkeep	0.5	0.5
	reg	0	0
<b>AMF</b>	num factors	15	22
	learning rate	0.001	0.0001
	l_w	0.0001	0.0001
	l_b	0.0001	0.0001
	l_adv	1	1
	eps	0.5	0.5
	epochs	20	30
	adversarial epochs	20	30
	eps_iter	0.00001	0.00001
	nb_iter	20	20
<b>VSM</b>	similarity	cosine	
	user profile	binary	-
	item profile	binary	

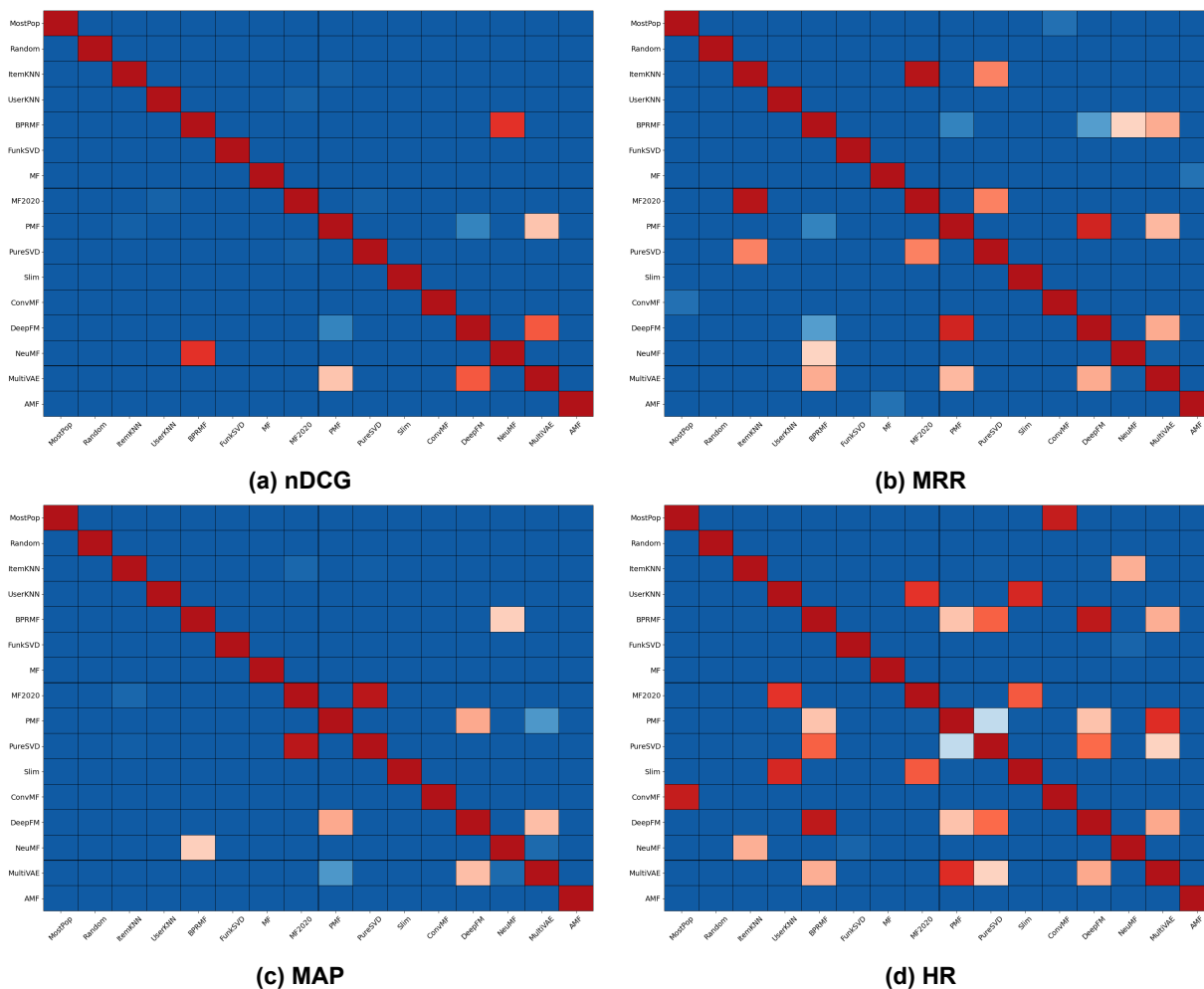
Table A.2: Hyperparameters of RAs used on ML and GR

## A.2. Supplementary Results RA Performance

We show in Table A.3 the performance results of  $ML_{\bar{S}}$  to highlight the similar results we got compared to the work of Anelli et al. [3]. The statistical significance between the RAs is illustrated in Figure A.1.

	$ML_{\bar{S}}$			
	nDCG	MRR	MAP	HR
<b>Non-personalized</b>				
<i>MostPop</i>	0.1622	0.3185	0.1614	0.6236
Random	0.0079	0.0213	0.0073	0.0703
<b>Neighborhood-based</b>				
Item-kNN	0.2993	0.5406	0.3023	0.8364
<i>User-kNN</i>	<u>0.3209</u>	<u>0.5652</u>	<u>0.3202</u>	<u>0.8764</u>
<b>Latent Factor Models</b>				
BPR-MF	0.2765	0.5042	0.2717	0.8539
FunkSVD	0.2583	0.4624	0.2542	0.8151
MF	0.2388	0.4302	0.2359	0.7837
MF2020	0.3144	0.5405	0.3091	0.8751
PMF	0.2927	0.5171	0.2882	0.8606
PureSVD	0.3078	0.5445	0.3093	0.8512
<i>Slim</i>	<b>0.3389</b>	<b>0.5814</b>	<b>0.3390</b>	<b>0.8773</b>
<b>Artificial Neural Networks</b>				
ConvMF	0.1563	0.3112	0.1565	0.6240
<i>DeepFM</i>	0.2885	0.5163	0.2860	0.8543
NeuMF	0.2772	0.4939	0.2763	0.8306
<b>Adversarial Learning</b>				
<i>AMF</i>	0.2105	0.4146	0.2127	0.7141
<b>Autoencoders</b>				
<i>MultiVAE</i>	0.2896	0.5105	0.2828	0.8595

**Table A.3:** Results for the RAs performance on  $ML_{\bar{S}}$ . The best model is highlighted in **bold** and the second best model is highlighted with an underline. The best model per RA type based on nDCG is highlighted in *italic*. Statistical significance for these results can be found in Figure A.1

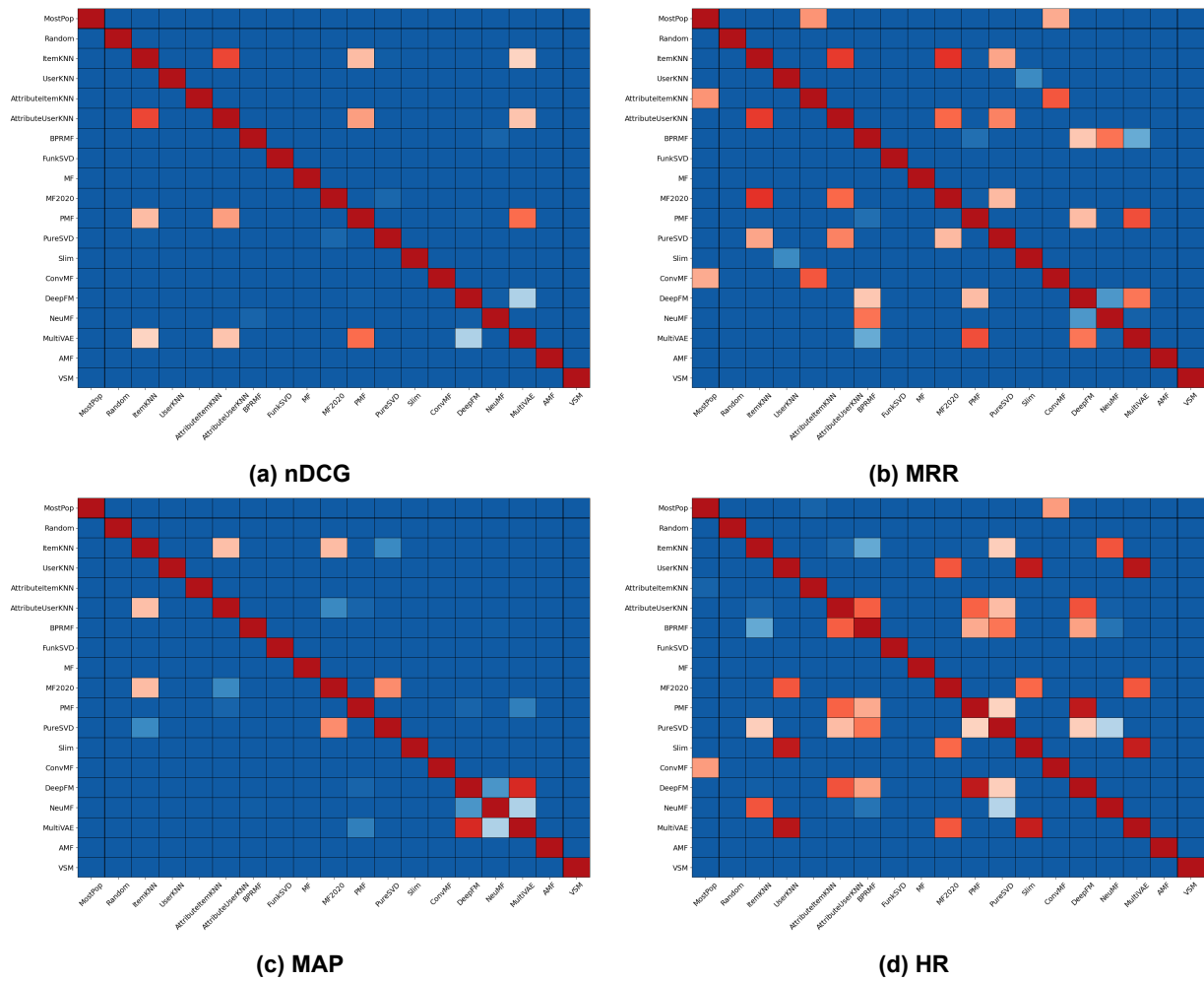


**Figure A.1:** Paired t-test for  $ML_{\bar{S}}$  with  $p < 0.05$ . The colors with blue tints mean statistically significant and colors with red tints mean statistically non-significant.

In Table A.4 we show the performance results of  $ML_{\bar{S}}$ . Figure A.2 shows the statistical significance between the RAs.

	$ML_S$			
	nDCG	MRR	MAP	HR
<b>Non-personalized</b>				
<i>MostPop</i>	0.1636	0.3186	0.1613	0.6239
Random	0.0082	0.0226	0.0079	0.0699
<b>Neighborhood-based</b>				
Item-kNN	0.2954	0.5296	0.2960	0.8289
<i>User-kNN</i>	<u>0.3179</u>	<u>0.5580</u>	<u>0.3140</u>	0.8640
AttributeItem-kNN	0.1427	0.3133	0.1438	0.6024
AttributeUser-kNN	0.2946	0.5312	0.2932	0.8425
<b>Latent Factor Models</b>				
BPR-MF	0.2731	0.4976	0.2672	0.8398
FunkSVD	0.2564	0.4593	0.2517	0.8067
MF	0.2393	0.4366	0.2352	0.7799
MF2020	0.3144	0.5405	0.3091	<b>0.8751</b>
PMF	0.2925	0.5115	0.2866	0.8452
PureSVD	0.3022	0.5349	0.3012	0.8364
<i>Slim</i>	<b>0.3324</b>	<b>0.5683</b>	<b>0.3293</b>	<u>0.8643</u>
<b>Artificial Neural Networks</b>				
ConvMF	0.1592	0.3162	0.1554	0.6279
<i>DeepFM</i>	0.2868	0.5056	0.2812	0.8448
NeuMF	0.2799	0.4938	0.2758	0.8265
<b>Adversarial Learning</b>				
<i>AMF</i>	0.2119	0.4093	0.2116	0.7194
<b>Autoencoders</b>				
<i>MultiVAE</i>	0.2911	0.5092	0.2806	0.8638
<b>Content-Based</b>				
<i>VSM</i>	0.1210	0.2752	0.1206	0.5699

**Table A.4:** Results for the RAs performance on  $ML_S$ . The best model is highlighted in **bold** and the second best model is highlighted with an underline. The best model per RA type based on nDCG is highlighted in *italic*. Statistical significance for these results can be found in Figure [A.2](#)



**Figure A.2:** Paired t-test for  $ML_S$  with  $p < 0.05$  and Bonferroni correction. The colors with blue tints mean statistically significant and colors with red tints mean statistically non-significant.

### A.3. NGIM Stereotype Prominence Results

	$HIT_{BAD,Gender}$		$MRR_{BAD,Gender}$		$REC-ST_{Gender}$	
	mean	std	mean	std	mean	std
MostPop	1.0	0.0	0.646789	0.228224	0.816597	0.087601
Random	1.0	0.0	0.871713	0.233075	0.754379	0.149649
ItemKNN	1.0	0.0	0.912844	0.202356	0.872977	0.153570
UserKNN	1.0	0.0	0.877294	0.221156	0.846872	0.135483
AttributeItemKNN	1.0	0.0	0.911894	0.210258	0.828440	0.172336
AttributeUserKNN	1.0	0.0	0.902905	0.203343	0.857298	0.135162
BPRMF	1.0	0.0	0.857798	0.236056	0.825605	0.139579
FunkSVD	1.0	0.0	0.915902	0.194159	0.863219	0.144900
MF	1.0	0.0	0.912844	0.194617	0.870809	0.128734
MF2020	1.0	0.0	0.910933	0.199202	0.853044	0.140481
PMF	1.0	0.0	0.853517	0.245461	0.841034	0.150891
PureSVD	1.0	0.0	0.978593	0.103944	0.926022	0.097467
Slim	1.0	0.0	0.931575	0.176189	0.890325	0.115671
ConvMF	1.0	0.0	0.651376	0.230253	0.860634	0.079861
DeepFM	1.0	0.0	0.874771	0.226554	0.855296	0.140558
NeuMF	1.0	0.0	0.882416	0.232017	0.832527	0.170723
MultiVAE	1.0	0.0	0.925076	0.185890	0.856547	0.144326
AMF	1.0	0.0	0.889908	0.218915	0.856881	0.120658
VSM	1.0	0.0	0.941437	0.173871	0.853044	0.143761

**Table A.5:** All stereotype prominence metrics for **gender** stereotypes in  $ML_{Ch,S}$  based on **NGIM**.

	$HIT_{BAD,Gender}$		$MRR_{BAD,Gender}$		$REC-ST_{Gender}$	
	mean	std	mean	std	mean	std
MostPop	0.996674	0.057577	0.764427	0.313749	0.365702	0.106749
Random	0.928096	0.258331	0.432273	0.337724	0.230684	0.150825
ItemKNN	0.929806	0.255477	0.483623	0.356049	0.281551	0.183888
UserKNN	0.954543	0.208306	0.531033	0.349373	0.314624	0.184927
BPRMF	0.988243	0.107791	0.623045	0.347412	0.327864	0.147124
FunkSVD	0.960410	0.194996	0.507514	0.345817	0.289708	0.161471
MF	0.953065	0.211502	0.534947	0.358577	0.303139	0.175613
PMF	0.983855	0.126036	0.786845	0.331063	0.322700	0.140468
PureSVD	0.952141	0.213470	0.535558	0.354138	0.281014	0.152763
Slim	0.939045	0.239251	0.450520	0.334936	0.275124	0.175169
ConvMF	0.998083	0.043743	0.410492	0.198702	0.241084	0.072784
DeepFM	0.937728	0.241652	0.545086	0.365348	0.296988	0.177637
NeuMF	0.944219	0.229502	0.517796	0.356502	0.287412	0.174202
MultiVAE	0.953481	0.210609	0.504057	0.344849	0.301369	0.177937
AMF	0.994757	0.072221	0.590618	0.326251	0.327428	0.127326

**Table A.6:** All stereotype prominence metrics for **gender** stereotypes in  $GR_{Ch}$  based on **NGIM**.

## A.4. BiasMeter Stereotype Prominence Results

	$HIT_{BAD,Gender}$		$MRR_{BAD,Gender}$		$REC-ST_{Gender}$	
	mean	std	mean	std	mean	std
MostPop	1.0	0.0	0.835015	0.284863	0.588574	0.118005
Random	1.0	0.0	0.787691	0.287378	0.618682	0.170166
ItemKNN	1.0	0.0	0.792169	0.284403	0.673394	0.175698
UserKNN	1.0	0.0	0.809830	0.278337	0.640701	0.170353
AttributeItemKNN	1.0	0.0	0.803058	0.273109	0.640951	0.176914
AttributeUserKNN	1.0	0.0	0.799847	0.280384	0.629441	0.155720
BPRMF	1.0	0.0	0.783104	0.284028	0.642202	0.168464
FunkSVD	1.0	0.0	0.804358	0.283187	0.639700	0.175676
MF	1.0	0.0	0.819495	0.272274	0.636364	0.169062
MF2020	1.0	0.0	0.812462	0.280299	0.659216	0.173981
PMF	1.0	0.0	0.829281	0.270962	0.651126	0.171297
PureSVD	1.0	0.0	0.777905	0.288750	0.647790	0.174118
Slim	1.0	0.0	0.747706	0.298412	0.646789	0.184180
ConvMF	1.0	0.0	0.853211	0.268432	0.657298	0.089192
DeepFM	1.0	0.0	0.813379	0.281037	0.640951	0.169526
NeuMF	1.0	0.0	0.763958	0.298007	0.622769	0.173970
MultiVAE	1.0	0.0	0.781651	0.285424	0.658549	0.178407
AMF	1.0	0.0	0.763532	0.289108	0.646038	0.128187
VSM	1.0	0.0	0.829562	0.264867	0.656464	0.157486

**Table A.7:** All stereotype prominence metrics for **gender** stereotypes in  $ML_{Ch,S}$  based on **BiasMeter**.

	$HIT_{BAD,Gender}$		$MRR_{BAD,Gender}$		$REC-ST_{Gender}$	
	mean	std	mean	std	mean	std
MostPop	1.000000	0.000000	0.997648	0.034956	0.860339	0.067985
Random	0.999723	0.016646	0.729363	0.308458	0.546845	0.178308
ItemKNN	0.996235	0.061244	0.759091	0.318483	0.620663	0.228465
UserKNN	0.997529	0.049653	0.833664	0.279426	0.708231	0.213202
BPRMF	0.999769	0.015196	0.928571	0.192139	0.776158	0.161314
FunkSVD	0.998730	0.035620	0.808227	0.284833	0.661731	0.194669
MF	0.998522	0.038420	0.793653	0.290060	0.637186	0.197117
PMF	0.999931	0.008324	0.973478	0.119222	0.833810	0.127503
PureSVD	0.999746	0.015938	0.860410	0.259705	0.743037	0.186058
Slim	0.997182	0.053010	0.767303	0.311131	0.648028	0.219279
ConvMF	1.000000	0.000000	0.982438	0.100202	0.692963	0.093936
DeepFM	0.996859	0.055960	0.813439	0.287174	0.656189	0.212788
NeuMF	0.996628	0.057974	0.787614	0.297782	0.646867	0.213489
MultiVAE	0.998291	0.041308	0.806364	0.292340	0.674368	0.214641
AMF	0.999561	0.020945	0.958446	0.144631	0.832589	0.106537

**Table A.8:** All stereotype prominence metrics for **gender** stereotypes in  $GR_{Ch}$  based on **BiasMeter**.

	$HIT_{BAD,Race}$		$MRR_{BAD,Race}$		$REC-ST_{Race}$	
	mean	std	mean	std	mean	std
MostPop	0.220183	0.415324	0.028411	0.056677	0.010926	0.025662
Random	0.660550	0.474612	0.206027	0.269945	0.083736	0.087520
ItemKNN	0.545872	0.499037	0.163725	0.245521	0.069224	0.088986
UserKNN	0.527523	0.500391	0.138445	0.206979	0.061301	0.079998
AttributeItemKNN	0.532110	0.500116	0.153419	0.236333	0.066639	0.082820
AttributeUserKNN	0.509174	0.501066	0.120105	0.177250	0.054545	0.072033
BPRMF	0.500000	0.501151	0.150706	0.236449	0.062385	0.086447
FunkSVD	0.564220	0.497000	0.167908	0.247658	0.071226	0.085321
MF	0.545872	0.499037	0.181342	0.267267	0.072644	0.091510
MF2020	0.596330	0.491762	0.206744	0.275199	0.091243	0.107274
PMF	0.532110	0.500116	0.137720	0.204727	0.060050	0.080028
PureSVD	0.623853	0.485532	0.168119	0.229471	0.070809	0.082489
Slim	0.541284	0.499440	0.146805	0.206416	0.071476	0.089957
ConvMF	0.683486	0.466186	0.088971	0.071602	0.033945	0.035540
DeepFM	0.550459	0.498592	0.163470	0.253111	0.066472	0.082587
NeuMF	0.577982	0.495018	0.210725	0.291948	0.089074	0.108515
MultiVAE	0.628440	0.484334	0.233437	0.307468	0.091743	0.101116
AMF	0.623853	0.485532	0.133983	0.145186	0.059716	0.057949
VSM	0.486239	0.500961	0.108763	0.164578	0.052544	0.072783

**Table A.9:** All stereotype prominence metrics for **race** stereotypes in  $ML_{Ch,S}$  based on **BiasMeter**.

	$HIT_{BAD,Race}$		$MRR_{BAD,Race}$		$REC-ST_{Race}$	
	mean	std	mean	std	mean	std
MostPop	0.643946	0.478837	0.452545	0.390278	0.110042	0.082227
Random	0.473553	0.499306	0.159239	0.259588	0.062555	0.086734
ItemKNN	0.500139	0.500006	0.171633	0.266774	0.068040	0.090841
UserKNN	0.563080	0.496011	0.216616	0.302287	0.078508	0.090560
BPRMF	0.614843	0.486638	0.310091	0.360225	0.090021	0.082718
FunkSVD	0.547166	0.497776	0.201925	0.291661	0.074324	0.089363
MF	0.475239	0.499392	0.160500	0.261599	0.061020	0.083337
PMF	0.643946	0.478837	0.190128	0.217730	0.083114	0.080498
PureSVD	0.528480	0.499194	0.182524	0.275211	0.065483	0.079164
Slim	0.487827	0.499858	0.143776	0.227676	0.060652	0.082076
ConvMF	0.643207	0.479059	0.395256	0.339223	0.107926	0.080622
DeepFM	0.565506	0.495696	0.226219	0.308374	0.081653	0.094234
NeuMF	0.524923	0.499384	0.183589	0.272426	0.070613	0.088345
MultiVAE	0.531159	0.499034	0.167716	0.253893	0.068673	0.088849
AMF	0.613619	0.486925	0.358898	0.391936	0.092684	0.079904

**Table A.10:** All stereotype prominence metrics for **race** stereotypes in  $GR_{Ch}$  based on **BiasMeter**.



	$HIT_{BAD,Religion}$		$MRR_{BAD,Religion}$		$REC-ST_{Religion}$	
	mean	std	mean	std	mean	std
MostPop	0.834862	0.372159	0.116195	0.070257	0.047957	0.038725
Random	0.174312	0.380251	0.067950	0.205123	0.020601	0.054770
ItemKNN	0.142202	0.350061	0.031668	0.103403	0.012344	0.035506
UserKNN	0.169725	0.376255	0.040693	0.126136	0.015430	0.041270
AttributeItemKNN	0.417431	0.494270	0.207099	0.333423	0.072143	0.104416
AttributeUserKNN	0.128440	0.335350	0.044743	0.150992	0.014846	0.043649
BPRMF	0.174312	0.380251	0.038942	0.112854	0.016597	0.045840
FunkSVD	0.151376	0.359240	0.034624	0.115126	0.013678	0.036793
MF	0.178899	0.384150	0.050448	0.153729	0.019600	0.049427
MF2020	0.160550	0.367961	0.045919	0.136858	0.018098	0.047113
PMF	0.201835	0.402293	0.046754	0.128327	0.019266	0.044506
PureSVD	0.165138	0.372159	0.050610	0.152605	0.017598	0.045363
Slim	0.137615	0.345288	0.031082	0.103102	0.012260	0.036487
ConvMF	0.834862	0.372159	0.243731	0.182627	0.106589	0.051187
DeepFM	0.146789	0.354710	0.036002	0.110027	0.015179	0.039805
NeuMF	0.160550	0.367961	0.038749	0.114023	0.015930	0.042284
MultiVAE	0.197248	0.398837	0.052017	0.138389	0.021435	0.048906
AMF	0.087156	0.282713	0.013499	0.046378	0.006172	0.022721
VSM	0.486239	0.500961	0.206997	0.307293	0.068307	0.085460

**Table A.11:** All stereotype prominence metrics for **religion** stereotypes in  $ML_{Ch,S}$  based on **BiasMeter**.

	$HIT_{BAD,Religion}$		$MRR_{BAD,Religion}$		$REC-ST_{Religion}$	
	mean	std	mean	std	mean	std
MostPop	0.034855	0.183414	0.004061	0.021991	0.001347	0.008477
Random	0.186585	0.389582	0.057295	0.167175	0.020583	0.050651
ItemKNN	0.223588	0.416654	0.070674	0.183815	0.026549	0.060768
UserKNN	0.178015	0.382530	0.054118	0.163035	0.020453	0.056567
BPRMF	0.189426	0.391851	0.031874	0.086458	0.013188	0.034991
FunkSVD	0.221601	0.415329	0.062112	0.165785	0.022503	0.050778
MF	0.222271	0.415777	0.058623	0.154739	0.023086	0.053899
PMF	0.077678	0.267668	0.015617	0.071620	0.006400	0.027186
PureSVD	0.289417	0.453497	0.080247	0.173473	0.031796	0.059109
Slim	0.263570	0.440574	0.082500	0.192603	0.031339	0.064091
ConvMF	0.884672	0.319421	0.120234	0.064961	0.048673	0.036430
DeepFM	0.192960	0.394626	0.057852	0.170596	0.022877	0.063336
NeuMF	0.216196	0.411654	0.065348	0.176342	0.024944	0.059389
MultiVAE	0.199011	0.399261	0.064334	0.182696	0.023302	0.058953
AMF	0.218483	0.413222	0.030976	0.064059	0.013163	0.029627

**Table A.12:** All stereotype prominence metrics for **religion** stereotypes in  $GR_{Ch}$  based on **BiasMeter**.

## A.5. ChatGPT 3.5 Stereotype Prominence Results

	$HIT_{BAD,Gender}$		$MRR_{BAD,Gender}$		$REC-ST_{Gender}$	
	mean	std	mean	std	mean	std
MostPop	0.715596	0.452168	0.103644	0.078700	0.046789	0.038069
Random	0.408257	0.492642	0.115857	0.210794	0.049875	0.074626
ItemKNN	0.261468	0.440446	0.089340	0.223326	0.027940	0.058419
UserKNN	0.362385	0.481796	0.132243	0.264670	0.041451	0.066214
AttributeItemKNN	0.417431	0.494270	0.132006	0.227525	0.050292	0.075513
AttributeUserKNN	0.389908	0.488852	0.135115	0.255191	0.044204	0.066093
BPRMF	0.376147	0.485532	0.112653	0.216487	0.042202	0.066850
FunkSVD	0.307339	0.462453	0.101438	0.200663	0.039283	0.068113
MF	0.353211	0.479068	0.110210	0.223222	0.039199	0.065900
MF2020	0.376147	0.485532	0.121611	0.237636	0.042952	0.068858
PMF	0.403670	0.491762	0.130091	0.224649	0.049291	0.070761
PureSVD	0.348624	0.477631	0.116304	0.233302	0.039616	0.063201
Slim	0.325688	0.469710	0.108907	0.231374	0.035780	0.062190
ConvMF	0.715596	0.452168	0.233180	0.199936	0.098666	0.063529
DeepFM	0.353211	0.479068	0.110128	0.216663	0.041451	0.066237
NeuMF	0.357798	0.480456	0.093030	0.177150	0.039533	0.064222
MultiVAE	0.307339	0.462453	0.087860	0.191906	0.034279	0.062621
AMF	0.541284	0.499440	0.125031	0.165577	0.053628	0.059456
VSM	0.376147	0.485532	0.122726	0.218623	0.048123	0.074382

**Table A.13:** All stereotype prominence metrics for **gender** stereotypes in  $ML_{Ch,S}$  based on **ChatGPT 3.5**.

	$HIT_{BAD,Gender}$		$MRR_{BAD,Gender}$		$REC-ST_{Gender}$	
	mean	std	mean	std	mean	std
MostPop	0.002171	0.046546	0.000241	0.005229	0.000073	0.001958
Random	0.216427	0.411814	0.067165	0.181019	0.024080	0.054830
ItemKNN	0.244445	0.429763	0.072583	0.185768	0.026765	0.058711
UserKNN	0.215549	0.411208	0.069182	0.189843	0.024622	0.058391
BPRMF	0.229223	0.420338	0.051544	0.137057	0.020712	0.049332
FunkSVD	0.308334	0.461811	0.090379	0.197960	0.032999	0.059441
MF	0.291033	0.454244	0.084721	0.192212	0.031658	0.060565
PMF	0.200790	0.400596	0.039676	0.106798	0.016590	0.040972
PureSVD	0.263963	0.440784	0.063859	0.153982	0.024338	0.050437
Slim	0.268236	0.443046	0.090346	0.217553	0.031022	0.063359
ConvMF	0.001732	0.041586	0.000188	0.004545	0.000057	0.001613
DeepFM	0.255255	0.436009	0.075118	0.185041	0.028706	0.061188
NeuMF	0.250751	0.433450	0.073987	0.184449	0.028061	0.060287
MultiVAE	0.262508	0.440002	0.079689	0.193272	0.029544	0.061310
AMF	0.136670	0.343503	0.021895	0.063579	0.009506	0.027966

**Table A.14:** All stereotype prominence metrics for **gender** stereotypes in  $GR_{Ch}$  based on **ChatGPT 3.5**.

	$HIT_{BAD,Race}$		$MRR_{BAD,Race}$		$REC-ST_{Race}$	
	mean	std	mean	std	mean	std
MostPop	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
Random	0.133028	0.340386	0.035891	0.124914	0.015346	0.047654
ItemKNN	0.110092	0.313724	0.033115	0.133608	0.011343	0.037379
UserKNN	0.082569	0.275863	0.014124	0.050680	0.006589	0.024697
AttributeItemKNN	0.114679	0.319367	0.022088	0.073318	0.008924	0.030094
AttributeUserKNN	0.055046	0.228595	0.008969	0.044866	0.003503	0.017797
BPRMF	0.091743	0.289327	0.022040	0.095428	0.007590	0.029442
FunkSVD	0.082569	0.275863	0.017755	0.085512	0.006756	0.028864
MF	0.055046	0.228595	0.012263	0.057218	0.005505	0.024809
MF2020	0.064220	0.245709	0.029689	0.153364	0.006922	0.031550
PMF	0.082569	0.275863	0.019865	0.089806	0.007506	0.028916
PureSVD	0.119266	0.324847	0.026638	0.099018	0.010759	0.033681
Slim	0.142202	0.350061	0.032503	0.106961	0.013344	0.039661
ConvMF	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
DeepFM	0.077982	0.268760	0.017666	0.085757	0.006422	0.026235
NeuMF	0.096330	0.295723	0.021454	0.090361	0.008257	0.029623
MultiVAE	0.123853	0.330172	0.033967	0.123437	0.012594	0.038568
AMF	0.009174	0.095562	0.001274	0.013538	0.000584	0.006621
VSM	0.050459	0.219393	0.008453	0.044690	0.003253	0.017281

**Table A.15:** All stereotype prominence metrics for **race** stereotypes in  $ML_{Ch,S}$  based on **ChatGPT 3.5**.

	$HIT_{BAD,Race}$		$MRR_{BAD,Race}$		$REC-ST_{Race}$	
	mean	std	mean	std	mean	std
MostPop	0.003742	0.061057	0.000432	0.007241	0.000138	0.002729
Random	0.033654	0.180338	0.010007	0.072200	0.003437	0.020922
ItemKNN	0.041738	0.199992	0.011273	0.072257	0.004113	0.022410
UserKNN	0.057976	0.233700	0.018349	0.103770	0.005687	0.026641
BPRMF	0.076362	0.265579	0.013827	0.061310	0.005657	0.023073
FunkSVD	0.098998	0.298662	0.028833	0.117687	0.010117	0.034550
MF	0.082644	0.275347	0.025123	0.113788	0.008515	0.032169
PMF	0.053564	0.225158	0.010309	0.054814	0.004274	0.020797
PureSVD	0.122765	0.328171	0.043479	0.155163	0.013928	0.041498
Slim	0.080589	0.272205	0.027968	0.127414	0.008810	0.033532
ConvMF	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
DeepFM	0.093546	0.291200	0.027423	0.116482	0.009576	0.033819
NeuMF	0.059800	0.237119	0.015138	0.081415	0.005595	0.025403
MultiVAE	0.058946	0.235526	0.018623	0.101390	0.006158	0.027880
AMF	0.015129	0.122068	0.002160	0.022602	0.000747	0.007542

**Table A.16:** All stereotype prominence metrics for **race** stereotypes in  $GR_{Ch}$  based on **ChatGPT 3.5**.

	$HIT_{BAD,Religion}$		$MRR_{BAD,Religion}$		$REC-ST_{Religion}$	
	mean	std	mean	std	mean	std
MostPop	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
Random	0.032110	0.176698	0.009646	0.075894	0.002752	0.019041
ItemKNN	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
UserKNN	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
AttributeItemKNN	0.013761	0.116767	0.002192	0.019089	0.001084	0.009891
AttributeUserKNN	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
BPRMF	0.004587	0.067729	0.000459	0.006773	0.000083	0.001231
FunkSVD	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
MF	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
MF2020	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
PMF	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
PureSVD	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
Slim	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
ConvMF	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
DeepFM	0.004587	0.067729	0.002294	0.033864	0.000751	0.011083
NeuMF	0.009174	0.095562	0.002867	0.034869	0.001001	0.011666
MultiVAE	0.009174	0.095562	0.001147	0.011945	0.000500	0.005212
AMF	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
VSM	0.087156	0.282713	0.016556	0.061305	0.007339	0.026942

**Table A.17:** All stereotype prominence metrics for **religion** stereotypes in  $ML_{Ch,S}$  based on **ChatGPT 3.5**.

	$HIT_{BAD,Religion}$		$MRR_{BAD,Religion}$		$REC-ST_{Religion}$	
	mean	std	mean	std	mean	std
MostPop	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
Random	0.004851	0.069478	0.001666	0.031987	0.000522	0.008364
ItemKNN	0.000115	0.010746	0.000015	0.001489	0.000006	0.000694
UserKNN	0.000277	0.016646	0.000101	0.007867	0.000033	0.002115
BPRMF	0.000115	0.010746	0.000018	0.001768	0.000008	0.000865
FunkSVD	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
MF	0.001016	0.031864	0.000249	0.010156	0.000095	0.003348
PMF	0.000393	0.019812	0.000065	0.004118	0.000024	0.001478
PureSVD	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
Slim	0.000023	0.004806	0.000003	0.000601	0.000001	0.000262
ConvMF	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
DeepFM	0.001455	0.038119	0.000471	0.016556	0.000161	0.004562
NeuMF	0.000670	0.025873	0.000146	0.007512	0.000053	0.002448
MultiVAE	0.000855	0.029222	0.000184	0.007997	0.000073	0.002875
AMF	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000

**Table A.18:** All stereotype prominence metrics for **religion** stereotypes in  $GR_{Ch}$  based on **ChatGPT 3.5**.