Understanding Biases in ChatGPT-based Recommender Systems: Provider Fairness, Temporal Stability, and Recency

YASHAR DELDJOO, Department of Electrical Engineering and Information Technology, Polytechnic University of Bari, Bari, Italy

This paper explores the biases inherent in ChatGPT-based recommender systems, focusing on provider fairness (item-side fairness). Through extensive experiments and over a thousand API calls, we investigate the impact of prompt design strategies—including structure, system role, and intent—on evaluation metrics such as provider fairness, catalog coverage, temporal stability, and recency. The first experiment examines these strategies in **classical top-K recommendations,** while the second evaluates **sequential in-context learning (ICL**).

In the first experiment, we assess seven distinct prompt scenarios on top-K recommendation accuracy and fairness. Accuracy-oriented prompts, like Simple and Chain-of-Thought (COT), outperform diversification prompts, which, despite enhancing temporal freshness, reduce accuracy by up to 50%. Embedding fairness into system roles, such as "act as a fair recommender," proved more effective than fairness directives within prompts. We also found that diversification prompts led to recommending newer movies, offering broader genre distribution compared to traditional collaborative filtering (CF) models. The system showed high consistency across multiple runs.

The second experiment explores sequential ICL, comparing zero-shot and few-shot learning scenarios. Results indicate that including user demographic information in prompts affects model biases and stereotypes. However, ICL did not consistently improve item fairness and catalog coverage over zero-shot learning. Zero-shot learning achieved higher NDCG and coverage, while ICL-2 showed slight improvements in hit rate (HR) when age-group context was included. Overall, our study provides insights into biases of RecLLMs, particularly in provider fairness and catalog coverage. By examining prompt design, learning strategies, and system roles, we highlight the potential and challenges of integrating large language models into recommendation systems, paving the way for future research. Further details can be found at https://github.com/yasdel/Benchmark_RecLLM_Fairness.

Additional Key Words and Phrases: Recommender Systems, Large Language Models, Bias and Fairness in RS, Movie Recommendation Analysis, Prompt Design Strategies, ChatGPT, Stability and Diversity in Recommendations

1 Introduction

Context. Recommender systems are integral to various large-scale internet services, benefiting consumers, producers, and other stakeholders in multi-stakeholder markets [1, 22, 26, 81]. The advent of generative models, particularly large language models (LLMs), holds the promise of offering better personalization experiences. LLMs enable conversational natural language (NL) interactions [4, 35], unlocking rich NL data like item descriptions, reviews, and queries within recommender systems (RS). Harnessing the general reasoning abilities of pretrained LLMs allows for addressing diverse, nuanced NL user preferences and feedback through highly personalized interactions. This contrasts with rigid ID-based methods that heavily rely on non-textual data. [19, 20]. This paper focuses on the emerging role of LLMs, specifically ChatGPT, in recommender systems and scrutinizes *their biases,* with a particular emphasis on "item-side fairness."

Item-side fairness is important to ensure diverse item groups receive fair exposure, benefiting item producers such as micro-businesses in job recommendations and promoting content related to vulnerable populations. Recent research shows that LLM-based recommender systems (RecLLMs), with their reliance on semantic clues and generative capabilities, can introduce unique biases not present in conventional systems [15, 18, 22, 40, 80]. Therefore, it is essential to investigate how these biases manifest and impact various stakeholders.

1.1 Background and Motivation

At a high level, the idea of generative modeling involves creating applications that not only make decisions based on data but also generate new data by learning patterns. This powerful idea has enabled various applications in AI disciplines, such as image generation, text synthesis, and music composition [28, 44, 52, 79]. Generative modeling has recently gained prominence due to advances in model paradigms such as Generative Adversarial Networks (GANs) [32], Variational Autoencoders

Authors' Contact Information: Yashar Deldjoo, Department of Electrical Engineering and Information Technology, Polytechnic University of Bari, Bari, Italy; e-mail: deldjooy@acm.org.

ACM 2770-6699/2024/8-ART
https://doi.org/10.1145/3690655

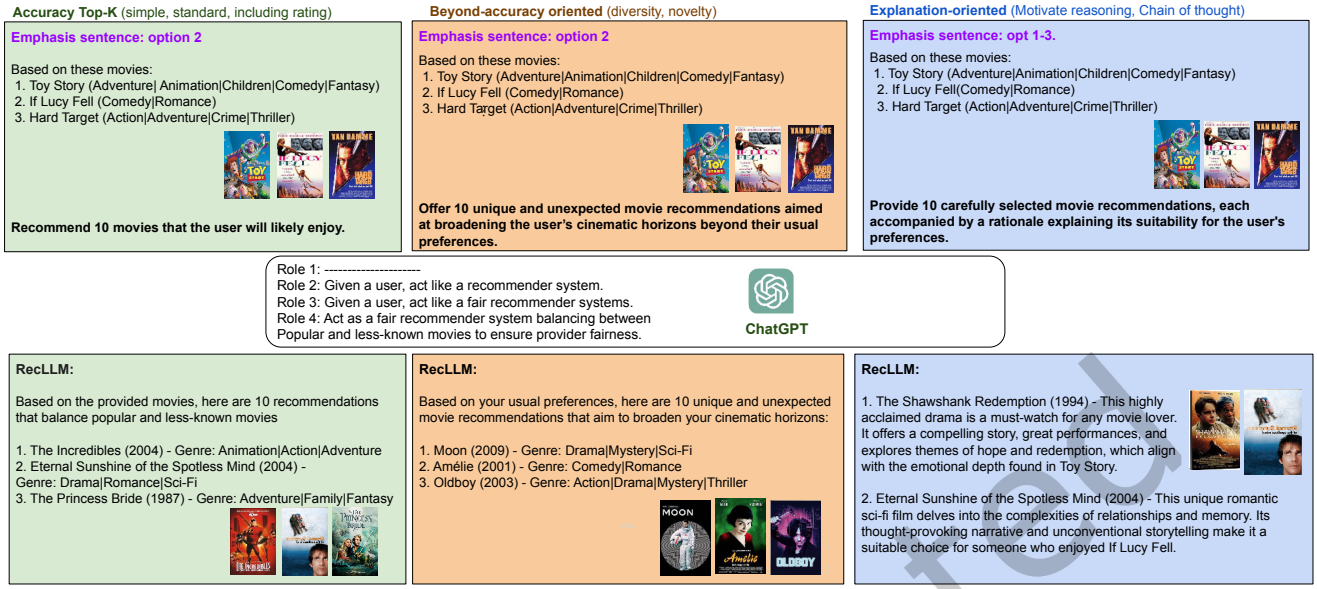


Fig. 1. Conceptual idea behind experiment 1, prompt-design scenarios.

(VAEs) [42], Diffusion Models [36, 64], and Transformer-based architectures such as GPT [8, 71] and other LLMs. In the recommender system (RS) field, generative models are not entirely new and have been used in various capacities for tasks such as data synthesis and augmentation [10, 78], model regularization [47, 51, 60], and generating complex recommendation structures [41, 49]. These advancements fall under Deep Generative Models (DGMs), which combine traditional probabilistic models with Deep Neural Networks (DNNs). The core strength of DGMs lies in their ability to model and sample from the data distribution they are trained on for various inferential purposes.

However, these models have experienced a resurgence of interest in the RS community, similar to trends in the broader AI landscape. This resurgence can be partly attributed to the introduction of LLMs such as ChatGPT, which have significantly improved the natural language understanding capabilities of these systems. LLMs demonstrate impressive in-context learning (ICL) and few-shot generalization abilities, making them valuable assets in both direct and indirect applications within recommender systems, where our goal here is to use them as direct recommenders (shown in Figure 1).

Notwithstanding their great success, the vast and unregulated nature of the internet data used to train LLMs raises concerns about possible biases against specific races, genders, popular brands, and other sensitive attributes that could be encoded in these networks. For example, if an LLM is predominantly trained on data from popular e-commerce sites, it might disproportionately recommend products from more recognized brands, overlooking niche or emerging brands. Similarly, biases in language around gender or race could skew recommendations in subtle but impactful ways. Another example could be the over-representation of content from certain geographical regions, leading to a lack of diversity in recommended media or news articles. Hence, the unchecked employment of these systems in commercial recommender systems may lead to unfair treatment of minority groups, reinforcing existing stereotypes, or exacerbating economic disparities.

This research explores different biases of ChatGPT-based RecLLMs, specifically on the **provider side,** to understand how these biases manifest in recommendations and affect various stakeholders. The primary goal of this research is two-fold:

(1) To enhance our understanding of the general performance of ChatGPT-based RecLLMs in terms of accuracy, provider fairness, and other nuanced evaluation dimensions (genre dominance, temporal stability, temporal freshness); and
(2) To determine whether factors such as prompt design and ICL type can be leveraged in RecLLM design to enhance item fairness without compromising accuracy;

To the best of our knowledge, the factors considered in this study are novel and have not been given enough attention in previous research. We believe that combining these studies into a single research effort opens up new avenues for future research in RecLLMs. Through **two** separate but extensive experiments, we introduce the following catalysts in the current work in hand:

- **Prompt Design.** We examine how different prompt structures and instructions affect recommendation outcomes. As shown in Figure 1 and Table 3, we designed seven prompt scenarios classified into three categories: *accuracy-oriented*,

beyond-accuracy, and *reasoning*. For instance, prompts instructing the model to diversify resulted in recommendations that were temporally fresher but not significantly novel or diverse, leading to a significant reduction in accuracy (e.g., NDCG) by 50%. In contrast, accuracy-oriented and reasoning-focused prompts, such as Simple and Chain-of-Thought (COT), performed much better.

- **System Roles.** We further investigated the impact of assigning specific **system roles**, such as "acting as a fair recommender." We observed that incorporating statements about fairness tends to improve system quality more effectively than including fairness directives directly within the prompt.
- **Stability Over Time.** We considered that external factors such as trending data or algorithmic updates might introduce randomness into the responses of GPT-based models. To study this, each experiment was conducted five times specifically for GPT-based models. We assessed the consistency of recommendations over multiple runs to ensure reliability.
- **In-Context Learning (ICL).** We analyzed the effect of zero-shot versus few-shot in-context learning (ICL) on provider fairness, focusing on demographic information impact on recommendations. We noted that incorporating user demographic data in prompts affected model biases and stereotypes. We structured the user profile into context, example, and demographic information parts. Our study highlights the nuanced effects of user profile attributes and sampling strategies on recommender systems.

Example related to Experiment 1. We illustrate the impact of different "prompt design" strategies on the output of ChatGPT-based recommendation systems through a clear example, as shown in Figure 1. We use a randomly chosen user profile from the MovieLens dataset to demonstrate this effect. The user's historical interactions with the system include a diverse range of movies such as "Toy Story" (categorized with genres Adventure|Animation|Children|Comedy|Fantasy), "If Lucy Fell" (Comedy|Romance), and "Hard Target" (Action|Adventure|Crime|Thriller).

To highlight the differences in recommendations based on the design of the system, we designed a total of seven prompts categorized into three distinct classes. These prompts were tailored to guide the recommendation system in different directions, showcasing how even with the same underlying user data, the outputs of the system can vary significantly depending on the design strategy employed.

(1) **Accuracy-Oriented Strategy.** This design approach is focused on delivering high Top$-K$ accuracy. It aims to provide recommendations that align closely with the user's established preferences. To achieve this, we designed *three prompts* in this category, aimed to guide the system to consider a combination of factors from the user profile, including items consumed (watched), favorite genres, and provided user ratings. This approach is tailored to reinforce the user's known tastes and preferences in the recommendations. It could be seen that it can recommend movies "The Incredibles (2004)" (genres: Action|Adventure), "Eternal Sunshine of the Spotless Mind (2004)" - (genres: Drama|Romance|Sci-Fi), and "the Princess Bride (1987)" - (genres: Adventure|Family|Fantasy), relying on their popularity and relevance to the user's taste.

(2) **Beyond-Accuracy-Oriented Strategy.** The goal of this beyond-accuracy-oriented approach is to broaden the user's viewing experience by introducing *diversity* and *novelty*. It steers away from strictly aligning with known preferences and instead presents a variety of unique and perhaps surprising movie recommendations. It could be noted that for the same given query, this results in the user being recommended a movie like "Moon (2009)" - (genres: Drama|Mystery|Sci-Fi), which may be outside the user's typical viewing history but offers a new cinematic perspective.

(3) **Reasoning-Oriented Strategy.** This strategy is centered on enhancing user engagement and understanding. It not only recommends movies but also provides detailed *explanation* and *reasoning* regarding the suggestions. While using these explanations can potentially assist users in exploring and understanding why certain movies are recommended, we are particularly investigating this scenario since previous research has shown that motivating the LLM to reason, for example, through the use of a chain-of-thought (COT) reasoning, could result in more accurate responses from LLMs (i.e., relevant recommendations). An example of recommendation under this strategy could be "The Shawshank Redemption (1994)", accompanied by an explanation of its themes of hope and redemption that resonate with the motivational depth found in "Toy Story."

1.2 Contributions.

We designed two carefully-structured experiments focusing on classical *top-k recommendation* and *sequential in-context learning*. These experiments involved thousands of API calls to gather the findings presented in this work. The first experiment investigates how different prompt design strategies impact recommendation accuracy, provider fairness, as well as genre

dominance and temporal freshness in classical top-k recommendations. The second experiment evaluates the effectiveness and fairness of RecLLMs in sequential in-context learning tasks, comparing zero-shot and few-shot learning scenarios across different datasets. This work makes several contributions to the field of recommender systems, as listed below:

- **Enhanced Analysis of Prompt Design in Zero-Shot RecLLMs:** We provide an in-depth study of how different prompt designs (in terms of intent, structure, and system role) affect the performance of ChatGPT-based recommendation systems. This includes an analysis of various prompt strategies (e.g., accuracy-oriented, beyond-accuracy-oriented, explanation-oriented) and their impacts on recommendation personalization, diversity, recency, and fairness.
- **Identification and Analysis of Biases in Zero-Shot RecLLMs:** Our research identifies and analyzes numerous biases present in RecLLMs, such as item fairness, genre preference bias, and temporal (recency) bias. We compare these biases with traditional CF models, providing insights into how they uniquely manifest in LLM-based systems;
- **Stability Analysis in Zero-Shot RecLLMs:** We study the stability and consistency of recommendations provided by LLM-based systems over time. Specifically, we examine variations caused by changes in underlying data, model updates, and their implications on the quality of recommendations obtained by RecLLMs;
- **Generalization of Fairness Principles in Sequential Recommendations:** We evaluate the effectiveness of RecLLMs in sequential recommendation tasks across different datasets, focusing on zero-shot versus few-shot learning. This analysis provides insights into the adaptability and robustness of these models in maintaining fairness across varied recommendation contexts.
- **Impact of User Profile Attributes and Demographic Information on Fairness and Accuracy:** We explore the influence of user profile attributes (historical interaction sampling) and the inclusion of demographic information (e.g., gender, age-group) on recommendation fairness and accuracy, providing new insights into how revealing such information can affect model biases and stereotypes.

These contributions collectively enhance our understanding of prompt-based recommendation systems using LLMs, offering new perspectives on their biases, fairness, and stability. Our findings highlight the potential and challenges of integrating LLMs into recommender systems, paving the way for future research and improvements in this domain.

The structure of the paper is organized as follows. Section 2 presents the related work, focusing on the exploration of fairness in recommendation systems (RS) and pre-trained language models (LMs), as well as their applications in enhancing RS. Section 3 is dedicated to presenting a thorough evaluation of ChatGPT. This is this research's core contribution which we present through the suite of goal-oriented prompts' (Section 3.1.1), the repeated experiments for stability analysis (Section 3.1.2), an understanding of the system role in ChatGPT-based RecLLM (Section 3.1.3), and the promotion of fairness (Section 3.1.4). Additionally, we explore the explicit versus implicit scenario, which is indicated in prompt structuring (Section 3.1.5). Following this, Section 4 outlines the experimental setup employed in our study. The results and key findings of these experiments are thoroughly discussed in Section 5. The paper concludes with Section 6, where we summarize our findings and outline potential avenues for future research.

2 Related work

In this section, we briefly review some related work on recommendation systems and LLM techniques.

2.1 Fairness in Recommender Systems

Fairness has become a pivotal topic in AI, gaining scrutiny across branches of trustworthy AI, including security, privacy, and explainability. Fairness in recommender systems has gained significant attention due to their multi-stakeholder nature [22, 26]. Unfairness, even in its minimal form, can adversely impact various stakeholders, including consumers, producers, system designers, supply chains, and even the environment – the latter are often referred to as 'side-stakeholder'.

The body of literature on fairness in RS is diverse, covering multiple perspectives and dimensions. Recent surveys [2, 22, 26], categorize these aspects into several dimensions, either orthogonal or partially orthogonal. Key dimensions recognized in the literature include the *main stakeholder* in question (e.g., consumer vs. producer), the target benefits associated with each (such as effectiveness vs. item exposure), the granularity of sensitive groups for assessing fairness (individual vs. group level), and other dimensions including the core definition of fairness, temporal aspects, and more. These dimensions offer a comprehensive view of the fairness landscape in RS, as detailed in these surveys.

To position this study within the literature of fair recommender systems, we introduced Table 1. This table is designed to categorize existing literature along two principal dimensions: the *stakeholder* in question (consumer vs. producer) and the *nature of the core RS* under scrutiny (traditional RS vs. those based on recent advancements in LLMs). We briefly review these dimensions in the following.

Table 1. Research landscape in Recommender Systems focusing on Consumer and Producer fairness with respect to Age, Gender, Other, and Others

		Traditional RS	RecLLM
Consumer Fairness	Activity	[33, 46, 57, 76]	–
	Demographics	[16, 17, 27, 73, 74]	[15, 18, 63, 80]
	Merits	[31, 65]	–
	Others	[48, 67]	–
Producer Fairness	Popularity	[13, 25, 29, 82]	[45]
	Demographics	[6, 43]	–
	Price/Brand/Location	[9, 16, 50, 62]	–
CP Fairness	Mixed attributes	[11, 24, 53, 55, 56, 75]	–

Core RS under scrutiny. We distinguish between 'traditional' RS and those enhanced by 'RecLLM'. This distinction is crucial because, while RecLLM, such as those based on GPT-like architectures, promise to significantly advance RS landscape with more nuanced and personalized recommendations, they also raise concerns about inherent biases. The extensive and unregulated nature of internet data used to train LLMs raises concerns about biases against specific races, genders, and other sensitive attributes. For example, if an LLM is trained on e-commerce data where men's products are described more positively than women's products, it might skew recommendations towards male-oriented items. Therefore, *measuring* these biases is a crucial initial step, and forms the key goal of our current work, in developing effective mitigation strategies. We can briefly review the research on fairness in recommendation systems according to Table 1 as follows:

- **Traditional RS.** This column lists studies that have focused on traditional methods of recommender systems [23]. They primarily rely on CF algorithms (possibly using side information of users and items) without the advanced natural language processing capabilities of LLMs. Within these works, some are dedicated to building evaluation frameworks for evaluating RS unfairness, while others focus on developing various mitigation strategies.
- **RecLLM**: This column spotlights the burgeoning research domain that merges language models (LMs) with RS representing a significant shift towards harnessing advanced NLP techniques to enhance the accuracy and relevance of recommendations. These studies explore the use of various LMs, including BERT-based models [63] and recent LLMs such as GPT-like architectures [45, 80]. Additionally, beyond the scale of LMs, the target tasks within RS–such as classical recommendation (top-k ranking [45, 80], sequential), conversational RS [63], explanation generation, multi-modal recommendations–provide dimensions that could be used for further categorizing these works.

Stakeholder. As mentioned earlier, a major aspect that can be utilized to classify almost all literature on group fairness is the market side focus—whether they concentrate on a single-side market (defined either by consumers or providers) or on both sides [21, 53, 58]. Within each of these segments, as you can observe, we can further categorize the literature based on which sensitive attribute groups are defined. For example, sensitive attributes such as the demographics of consumers (age, gender) and producers, as well as the popularity of items (on the produce side) are quite common focal points.

- **Consumer Fairness.** This category is subdivided into various attributes like Activity, Demographics, Merits, and Others. It includes studies that focus on ensuring fairness among consumers of the recommender system based on these attributes. For instance, ensuring that recommendations are not biased towards a particular demographic consumer group.
- **Producer Fairness.** This focuses on the fairness towards the providers or producers of the content or products recommended by the system. It includes subcategories like Popularity, Demographics, and Price/Brand/Location. These studies might address issues like ensuring lesser-known or niche producers get fair visibility and opportunity in the recommendation process.
- **CP Fairness.** This category involves studies that consider both consumer and producer fairness simultaneously, addressing the balance between the two.

It is noteworthy that while traditional Fair-RS research has been extensively explored, RecLLM is an emerging field, presenting its own unique considerations and challenges. Our work is situated in the 'Producer Fairness' category under 'RecLLM', focusing on producer fairness in the context of ChatGPT. It focuses on producer fairness in the context of ChatGPT, particularly examining how *prompt engineering* techniques can be leveraged to address or potentially enhance fairness.

The study by Zhang et al. [80] evaluates the consumer fairness side of zero-shot GPT recommendations, focusing on a variety of consumer demographic attributes but not addressing producer-side fairness. Their work introduces a novel benchmark, FaiRLLM, for evaluating the fairness of RecLLM, highlighting ChatGPT's biases towards certain sensitive user attributes in music and movie recommendations. The work by Deldjoo et al. [15, 18] address the shortcomings by scrutinizing whether changes in recommendations, due to the inclusion of sensitive attributes, result in unfairness. It also explores better ways to normalize and improve the fairness of these models. Conversely, the work by Li et al. [45] aligns more closely with ours, focusing on producer unfairness however within the specific domain of *news recommendation*. This study investigates performance of ChatGPT in news recommendation, exploring aspects like personalization, provider fairness, and fake news detection. However, these studies do not address the intricacies of prompt engineering in RecLLMs, nor do they comprehensively address the various forms of biases studies in the current study. Our research goes beyond examining provider fairness based on item unfairness and also considers *other* potential *harms*, such as the recency of recommended items and the stability of recommendations—aspects not explored in the aforementioned studies.

2.2 Leveraging Pre-trained LMs and Prompting for Recommender Systems

Recent advancements in recommender systems (RS) have been significantly influenced by the integration of LLMs and innovative prompting strategies. The use of natural language in recommendation tasks has been explored in various ways. For instance, Hou et al. [37] utilize natural language descriptions and tags as inputs into LLMs to create user representations for more effective recommendations. This contrasts with the narrative-driven recommendations [5] that rely on verbose descriptions of specific contextual needs.

In terms of prompting techniques, early methods relied on few-shot prompting, where training examples are used as a guide for LLMs [7]. With prompt learning, tasks are adapted to LLMs rather than the other way around, utilizing discrete prompts or continuous/soft prompts for task performance. This approach has shown promise across various tasks, including recommendation tasks.

Personalizing LLMs for recommendation is crucial for understanding a user's intent and addressing their personalized needs. Recent efforts like P5 [30] and OpenP5 [77] have integrated several recommendation tasks into one LLM using personalized prompts. This approach reformulates recommendation tasks as sequence-to-sequence generation problems, demonstrating the flexibility of LLMs in handling diverse recommendation scenarios. Lastly, prompt transfer research, such as SPoT [66] and ATTEMPT [3], focuses on learning from source tasks and utilizing this knowledge for target tasks. This method, including knowledge distillation techniques, shows potential in intra-task prompt distillation and cross-task prompt transfer, contributing to the efficiency and effectiveness of LLM-based recommendation models.

In recent surveys [19, 20], the wide area of generative models has been extensively explained. These surveys highlight the potential of generative models in transforming traditional RS by leveraging their capabilities in understanding and generating complex data distributions. For instance, they have shown how generative models can effectively handle multimodal data (text, images, and videos), thereby enhancing the richness and accuracy of recommendations. These surveys also shed light on the emergent reasoning abilities of LLMs, such as in-context learning, which allows models to adapt to new tasks with minimal additional training. Moreover, they discuss the application of retrieval-augmented generation (RAG) techniques, which combine information retrieval with generative modeling to produce contextually relevant recommendations. Overall, these insights underscore the significant advancements and ongoing challenges in utilizing generative models for RS.

3 Evaluation of ChatGPT-based RecLLM

We designed two experiments to evaluate the performance and bias of ChatGPT-based RecLLMs.

- **Experiment 1.** Examining **prompt design** strategies in classical *top-k recommendation* (cf. Section 3.1), and
- **Experiment 2.** Studying **generalization** of findings across datasets, and learning strategies (zero vs. few-shot ICL) in *sequential recommendation* task (cf. Section 3.2)

Table 2 provides a summary of these experiments.

3.1 Experiment 1: Examining Prompt Design Strategies in Classical Top-K Recommendations

In the first experiment, we investigate prompt design strategies in a classical top-K recommendation setting. We use an LLM (ChatGPT) as a zero-shot recommender, querying it with various prompt formulations to obtain recommendations. Our goal is to understand how much the recommendation outcome (performance) can be influenced by prompts carrying different objectives (e.g., relevance, increasing diversity, or motivating reasoning). ur goal is to determine if a *one-to-one mapping* exists between the desired natural language query (e.g., enhance diversity) and the actual diversity of the model recommendations.

Table 2. Summary of Experiments

Exps.	Aim	Approach	Evaluation Metrics
Exp 1	(i) Investigate prompt design strategies in classical top-K recommendations (ii) Analyze stability of results over different RecLLM runs, (iii) Look at new content-related metrics (genre diversity, temporal freshness)	Explore prompt formulations (accuracy, diversity, fairness), system role directive, specific information in prompt in the movie domain	Accuracy (top-K precision and recall), beyond-accuracy metrics (diversity, novelty), fairness metrics, genre diversity, temporal freshness
Exp 2	(i) Assess generalization of fairness across datasets and beyond zero-shot learning to ICL, (ii) Look at the impact of user profile construction and demographic information revealing	Use movie and music dataset, evaluate few-shot and zero-shot ICL in sequential recommendation	Accuracy, provider fairness, catalogue coverage

To achieve the goals of this paper, we examine the relevance and item fairness (including diversity) of the recommendations. Additionally, we explore new aspects of recommendation, such as content diversity and freshness, including genre diversity and temporal freshness.

3.1.1 Goal-oriented prompts. Seven distinct prompt scenarios were designed, categorized into three classes focused on personalization, beyond-accuracy metrics, and reasoning. These classes aim to explore how variations in the nature of the prompt influence the recommendations. The scenarios are detailed in Table 3 and further exemplified in Section 1.

In particular, scenarios **S1** to **S3** are focused on core personalization aspects, encompassing basic recommendations without additional context (**S1**), genre preferences (**S2**), and the incorporation of explicit user ratings (**S3**). Meanwhile, Scenarios (**S4**) and (**S5**) are designed to enhance the diversity and novelty of the recommendations, with **S4** emphasizing lesser-known film recommendation and **S5** aiming to broaden the user's cinematic horizons with unique and unexpected choices. Scenarios **S6** and **S7** are explanation-motivation oriented, where **S6** seeks to provide reasoning for each recommendation to enhance transparency, and **S7** involves a logical ste-by-step reasoning process to arrive at the recommendations. These scenarios compose a competitive and insightful set of prompts, some of which such as COT, have been successfully tested in other ML disciplines [12, 70, 72].[1]

3.1.2 Repeated Experiment for the Stability of the Analysis. The experiments were conducted at a temperature setting of 0.0, aiming for predictable responses. However, we observed variations in the recommendations, possibly due to factors like trending data, recent updates in the database, or changes in the behavior of ChatGPT. This suggests that the recommendation system may interpret the same input differently in each iteration, independent of the temperature setting.

To assess the robustness and reliability of our system, we repeatedly conducted recommendation queries for each user, five times each, to examine if outputs vary despite consistent input. We further adopted a *bootstrapping sampling strategy* during the evaluation stage, as detailed in Section 4.1.3. This method is particularly effective in revealing the reliability of the results, even when dealing with datasets that are small or may not perfectly represent the entire population. This thorough evaluation approach, therefore, enhances our confidence in the performance of the system across various potential real-world scenarios.

3.1.3 Understanding the impact of "System" Role in ChatGPT. In this study, we sought to understand the impact of different system roles assigned to ChatGPT on recommendation outcomes. We focused on assessing whether responses significantly differ when specific "system" instructions are attached, and how these variations influence the functionality of a recommender system. Specifically, we aimed to determine the effectiveness of embedding fairness directly into the system role versus incorporating it in the prompt itself. Table 4 summarizes the roles assigned to ChatGPT during our experiments:

[1]**Note.** The CF baselines used in this work operate in an implicit setting. For consistency and fairness, all scenarios in the generative part adhered to this setting, thereby not revealing user movie ratings to the LLM in the prompts. Scenario 'Rating-focused' **S3** is the only exception, included solely for completeness. This is discussed in Section 3.1.5.

Table 3. Overview of prompt scenarios designed for the experiment 1.

Scenario	Description and Prompt
Personalization Focused	
S1 - Simple	Basic recommendations without additional context. *Prompt: "Recommend 10 movies that the user will likely enjoy."*
S2 - Genre-focused	Recommendations focusing on genres and themes similar to the user's past favorites. *Prompt: "Recommend 10 movies that the user will likely enjoy, particularly focusing on genres and themes similar to their past favorites."*
S3 - Rating-focused	Incorporation of the user's explicit ratings into the recommendations. *Prompt: "Recommend 10 movies the user will likely enjoy, taking into account both their favorite genres and past movie ratings."*
Beyond-Accuracy Focused	
S4 - Diversify Recommendations	Suggesting lesser-known films to diversify the user's experience. *Prompt: "Suggest 10 high-quality, lesser-known films that diverge from mainstream blockbusters, yet align with the user's tastes."*
S5 - Surprise	Offering unexpected recommendations for exploring new preferences. *Prompt: "Offer 10 unique and unexpected movie recommendations aimed at broadening the user's cinematic horizons beyond their usual preferences."*
Reasoning-Explanation Focused	
S6 - Motivate Reasoning	Providing reasoning for each recommendation to enhance transparency. *Prompt: "Provide 10 carefully selected movie recommendations, each accompanied by a rationale explaining its suitability for the user's preferences."*
S7 - Chain of Thought (COT)	Engaging in a logical reasoning process to arrive at the recommendations. *Prompt: "Let's think this through: What would be 10 great movie recommendations for this user and why?"*

3.1.4 Fairness Emphasis. We also aim to understand whether integrating a fairness statement in the system or in the prompt itself is more effective. To assess the impact of explicit directives on the recommendation system, each scenario was optionally combined with a 'Fairness Emphasis Statement.' These statements directed the model towards specific objectives, here ensuring fairness. The considered emphasis options included

Table 4. Summary of Different Roles Assigned to ChatGPT during experiments

Role ID	Description
R0 - No Role	Direct user-driven prompts without system context.
R1 - System Role as Recommender	System-centric role focusing purely on user information.
R2 - System Role as Fair Recommender	System role with an explicit fairness objective.

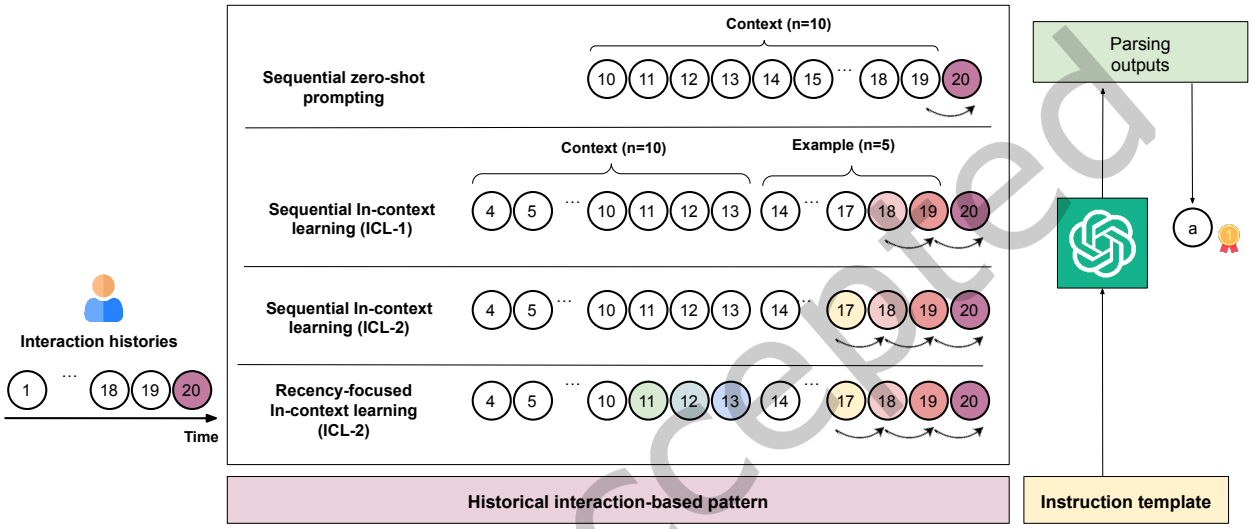


Fig. 2. Sequential in-context learning for various scenarios explored in Experiment 2 of this research.

- **E0 - Without Fairness Statement:** No fairness statement is included in this option. Prompts are used as described in Table 3.
- **E1 - With Fairness Statement:** This option emphasizes ensuring fair recommendation between popuar nd less popular movies. "*Prompt: Ensure a fair representation of both popular and less-known movies. Based on these movies: {user_movies_string}, recommend 10 movies that the user will likely enjoy*". The latter prompt is combining Scenario (S1/E1).

3.1.5 Explicit vs. Implicit Scenario. This test evaluated whether displaying the movies a user has rated, as indicated in {user_movies_string}, enhances the quality of RS, based on the premise that including such information could improve recommendations. Scenarios **S1 to S6**, except **S3**, were conducted implicitly, without user ratings in the prompts, mirroring the implicit mode of the CF baseline experiments. Uniquely, **S3** was examined in both explicit and implicit contexts. In the explicit context, user ratings were integrated into the prompts, but were absent in the implicit context. This explicit approach was specific to **S3**. An example of {user_movies_string} and the presentation of user movie lists and ratings in the explicit context is as follows.

Example. To illustrate the experimental scenarios, consider a user's movie list represented in this manner.

```
The Matrix (Genres: Action|Sci-Fi), Inception (Genres: Action|Thriller)
```

In the Explicit scenario, the prompt would be constructed to include these ratings, exemplified here:

```
The Matrix (Genres: Action/Sci-Fi, Rating: 5/5), Inception (Genres: Action/Thriller, Rating: 4.5/5)
```

The primary hypothesis under investigation is whether the inclusion of explicit ratings enables the system to tailor recommendations more closely to the user's demonstrated preferences, thereby enhancing the personalization of the recommendation process.

3.2 Experiment 2: Sequential In-Context Learning

The task of recommending suitable movies/music to users can be modeled as a sequential ranking problem where the historical interactions of a user are utilized to predict future preferences. In this context, the goal is to rank candidate items $C = \{i_j\}_{j=1}^{m}$ such that the most relevant items for the user appear at the top. Given a user's interaction history $H = \{i_1, i_2, \ldots, i_n\}$, to leverage it for generating recommendations, we opt to divide it into two main parts, as displayed in Figure 2:

(1) **Context part.** This part is sampled from the *entire interaction history* of the user to provide general context about the user's interests, passion, and preference (toward music and movies). We use two strategies for sampling:
   - *Frequent/Highest-rated:* Selects the most enjoyed items by the user, based on the ratings provided (for movies) or frequency of consumption (for music);
   - *Recency-Focused:* Utilizes a weighted average method to emphasize recently consumed items. This strategy selects items that have not only been liked by the user but also have been recently listened to or watched, ensuring relevance and timeliness in the recommendations.

(2) **Example part.** This part is taken from *recent interactions* to guide specific next recommendations. It directly determines the next items to be consumed in sequential tasks and is used in ICL-1, and ICL-2, which denote the *few-shot learning* scenarios, using one or two examples.

Based on this division, we explore different prompting strategies, as outlined in the following. We note that Experiment 2 has been inspired by Hou et al. [39] and extends it by exploring the above profile investigation, and evaluation going beyond pure accuracy.

Note. In the prompting scenarios outlined below, we *optionally* incorporate "user demographic information" (such as gender, or age category) into the prompts to examine how revealing this information might influence model biases and stereotypes. Specifically, where we have `{is demographic}` information, it is used in statements like "`The user is female`", "`The user is young`". Details of the results of comparisons with and without demographic information on fairness are provided in Table 10.

Sequential Zero-Shot Prompting

This method only uses the context part of the user interaction history

Example:

```
The user {is female, and} has watched the following movies:

- "From Russia with Love (1963)" in genre(s) Action with rating 5
- "Star Trek IV: The Voyage Home (1986)" in genre(s) Action|Adventure|Sci-Fi with rating 4
- "Planet of the Apes (1968)" in genre(s) Action|Sci-Fi with rating 3
- "Star Wars: Episode I - The Phantom Menace (1999)" in genre(s) Action|Adventure|Fantasy|Sci-Fi with rating 4
- "Final Conflict, The (a.k.a. Omen III: The Final Conflict) (1981)" in genre(s) Horror with rating 3

This selection reflects the user's movie preferences.
What would be the top-1 suitable next recommendation?
```

as stated earlier, `{is female}` is essentially denoted with `{is demographic}` below, which is tested in two scenarios: *included* and *omitted*.

Sequential In-Context Learning (ICL-1)

This approach takes the contest part of the preivous part, but taked the xample part of the interaction and from that take the last interatcion to guide the model how the recommendation. The examples are derived from recent interactions.

Example:

```
The user {is demographic, and} has watched the following movies:

- "From Russia with Love (1963)" in genre(s) Action with rating 5
- "Star Trek IV: The Voyage Home (1986)" in genre(s) Action|Adventure|Sci-Fi with rating 3
- "Planet of the Apes (1968)" in genre(s) Action|Sci-Fi with rating 3
- "Star Wars: Episode I - The Phantom Menace (1999)" in genre(s) Action|Adventure|Fantasy|Sci-Fi with rating 4
- "Final Conflict, The (a.k.a. Omen III: The Final Conflict) (1981)" in genre(s) Horror with rating 3

This selection reflects the user's movie preferences.
Given the user has recently watched the following movies in order:
1. "Face/Off (1997)" in genre(s) Action|Sci-Fi|Thriller with rating 4
2. "Bringing Out the Dead (1999)" in genre(s) Drama|Horror with rating 5
3. "Sixth Sense, The (1999)" in genre(s) Thriller with rating 5
4. "Austin Powers: The Spy Who Shagged Me (1999)" in genre(s) Comedy with rating 4
5. "Arlington Road (1999)" in genre(s) Thriller with rating 4

You should recommend:
Recommendation 1: "Bowfinger (1999)" in genre(s) Comedy with rating 3
What would be the top-1 suitable next recommendation after the above movies?
```

Sequential In-Context Learning (ICL-2)

Similar to ICL-1 but with a bigger selection of examples equal to 2. The examples are chosen to provide more specific guidance on generating recommendations based on recent user interactions.

Example:

```
The user {is demographic, and} has watched the following movies:

- "From Russia with Love (1963)" in genre(s) Action with rating 5
- "Star Trek IV: The Voyage Home (1986)" in genre(s) Action|Adventure|Sci-Fi with rating 3
- "Planet of the Apes (1968)" in genre(s) Action|Sci-Fi with rating 3
- "Star Wars: Episode I - The Phantom Menace (1999)" in genre(s) Action|Adventure|Fantasy|Sci-Fi with rating 4
- "Final Conflict, The (a.k.a. Omen III: The Final Conflict) (1981)" in genre(s) Horror with rating 3

This selection reflects the user's movie preferences.

Given the user has recently watched the following movies in order:
1. "Strange Days (1995)" in genre(s) Action|Crime|Sci-Fi with rating 3
2. "Face/Off (1997)" in genre(s) Action|Sci-Fi|Thriller with rating 4
3. "Bringing Out the Dead (1999)" in genre(s) Drama|Horror with rating 5
4. "Sixth Sense, The (1999)" in genre(s) Thriller with rating 5
5. "Austin Powers: The Spy Who Shagged Me (1999)" in genre(s) Comedy with rating 4

You should recommend:
Recommendation 1: "Bowfinger (1999)" in genre(s) Comedy with rating 3
Recommendation 2: "Arlington Road (1999)" in genre(s) Thriller with rating 4
What would be the top-1 suitable next recommendation after the above movies?
```

Recency-Focused Zero-shot and ICL-1 and ICL-2

We replicate all previous prompting scenarios but modify the "context part" to include the most recent interactions. This approach indicates to the ChatGPT recommender system that the best recommendations are based on the user's recent consumption.

The next sections outlines experiments conducted to assess various reproducibility aspects, including the obtained results and our observations.

4 Experimental Setup

In this section, we detail the experimental setup for two experiments, including the *datasets, evaluation metrics, bootstrapping methods, baselines employed,* and *hyperparameter tuning*.

4.1 Experiment 1. Classical Top-k Recommendation

4.1.1 Datasets and Setup. Experiment 1 used the `MovieLens-Latest-Small` dataset from the GroupLens group.[2] This dataset comprises the latest movie entries and was used in its entirety to carry out experiments in part 1 of our study. The experimental design required conducting API calls for each user, resulting in a total of 610 API calls (equal to the number of users) per individual recommendation setting. Given the scope—610 users, 7 different prompts or prompt scenarios (cf. Section 3.1.1), 2 fairness statements in the prompt (cf. Section 3.1.4), and 2 system roles (cf. Section 3.1.3)—we generated **thousands** of scenarios or API calls. Due to the computational demands, we limited our experiments in part 1 to a *single dataset*, focused on the **classical top-k recommendation task.** The detailed statistics of the dataset are provided in Table 5, where we implemented a data split of 80% training, 10% validation, and 10% testing. The validation dataset is used solely for CF baseline hyperparameter tuning, while both RecLLMs and the CF baseline utilize the same training and testing datasets to conduct recommendations.

4.1.2 Evaluation Metrics. The first experiment employs a diverse range of metrics to assess the performance of different RS. These metrics are categorized into two primary classes: *accuracy metrics*, and *provider fairness/diversity*. Additionally, we incorporate new **content-centric metrics** such as *temporal freshness* and *genre dominance/tendancy* to explore insights into potential biases and tendencies of RecLMMs compared to mainstream CF models.

Accuracy Metrics. These conventional metrics measure the quality of the top-k recommendation or the personalization of recommendations

- **NDCG@k:** Measures the ranking quality by evaluating the placement of relevant items within the top-k positions.
- **Recall@k**: The proportion of relevant items that are successfully recommended.

High scores in these metrics indicate more accurate and relevant recommendations.

Provider Fairness/Diversity. Beyond conventional accuracy metrics, these metrics evaluate the fairness of the recommendations from the provider perspective. They assess how well the system introduces new and varied content and how equitably it treats different item providers.

- **Gini Index [17]:** Measures the inequality in the distribution of item recommendations. A lower Gini Index indicates a more equitable distribution among items. The Gini index is calculated as follows:

$$\text{Gini} = \frac{\sum_{i=1}^{n}(2i - n - 1)x_i}{n\sum_{i=1}^{n} x_i}$$

  where x_i represents the number of times item i is recommended, sorted in ascending order, and n is the total number of items. The Gini index becomes 0 when all items are recommended an equal number of times, indicating perfect equality. Mathematically, if all x_i are equal, then:

$$x_i = \frac{\sum_{i=1}^{n} x_i}{n} \quad \forall i \implies \text{Gini} = 0$$

  The Gini index becomes 1 when one item is recommended exclusively, indicating maximum inequality. In our work, we interpret the Gini index to measure item fairness, *where lower Gini values indicate higher (better) item fairness.*
- **HHI (Herfindahl-Hirschman Index) [54, 68]:** This metric Assesses the concentration of recommendations among items. It is calculated as follows:

$$\text{HHI} = \sum_{i=1}^{n}\left(\frac{x_i}{\sum_{j=1}^{n} x_j}\right)^2$$

[2] https://grouplens.org/datasets/movielens/

Table 5. Statistics of the final datasets used in this work after k-core pre-processing.

Dataset	|U|	|I|	|R|	$\frac{R}{U}$	$\frac{R}{I}$	Density	Item Gini	User Gini
MovieLens Latest Small	610	9,724	100,836	165.30	10.36	98.3%	0.715	0.603

The HHI value ranges from $1/n$ to 1, where n is the total number of items. The HHI is $1/n$ when all items are recommended equally, indicating maximum fairness, and it is 1 when one item is recommended exclusively, indicating no fairness. Here, *lower values of the HHI are considered indicative of better item fairness.*

- **Entropy:** Reflects the diversity and unpredictability of recommendations. Higher entropy values indicate greater diversity. The Entropy is calculated as follows:

$$\text{Entropy} = -\sum_{i=1}^{n} p_i \log(p_i)$$

where $p_i = \frac{x_i}{\sum_{j=1}^{n} x_j}$ and $p_i > 0$. The entropy value ranges from 0 to $\log(n)$, where 0 indicates no diversity (one item recommended exclusively) and $\log(n)$ indicates maximum diversity (all items recommended equally). *Higher values of Entropy are considered indicative of better item fairness.*

While the above metrics are used to measure item fairness based on item exposure, they also serve as indicators of how recommendations produce **diverse** outcomes.

4.1.3 Bootstrapping Sampling Strategy. In experiment 1, for the evaluation, we additionally employ a *bootstrapping sampling strategy*, a statistical technique where 1000 samples were generated by repeatedly resampling our dataset with replacement [14]. This method was chosen to enhance the robustness and reliability of our analysis, particularly in measuring the performance metrics of our recommendation system, such as NDCG, Recall, and Precision.

Through this approach, we computed means for each metric across all bootstrap samples, then calculated their averages and 95% confidence intervals. This strategy is implemented for understanding the variability and confidence in our metrics, to ensure our evaluation reflects the likeness of the system performance on real-world performance. This is especially beneficial in situations with small or non-representative samples, allowing for more reliable population inferences.

4.1.4 CF Baselines. The choice of models and the corresponding hyperparameter optimization process is outlined as follows.

Models. We use a suite of competitive, CF recommendation models as baseline ranking models in our post-processing approach, as summarized below.

- **BPR** [59]: A conventional recommendation model that employs matrix factorization to learn user and item embeddings of low dimensionality, and optimize the model based on the pairwise ranking of items for each user to predict whether a user prefers a given item over another;
- **ItemKNN** [61]: An item-based K-nearest employs neighbors algorithm, utilizing similarity metrics such as cosine similarity to identify the closest item neighbors. These neighboring items are then leveraged to predict scores for user-item pairs.
- **MultiVAE** [47]: A non-linear probabilistic deep learning model that extends a variational autoencoder (VAE) structure to collaborative filtering for implicit feedback, and acquires the underlying representations of users and items from their interactions to create recommendations in an unsupervised way.
- **LightGCN** [34]: A pure collaborative filtering method that utilizes a simplified version of graph convolutional networks (GCNs) without nonlinear activation functions and additional weight matrices. It learns user and item embeddings through graph propagation rules and user-item interactions, making it scalable and efficient.
- **NGCF** [69]: A graph-based recommendation model that employs a neural network architecture and learns high-order connectivity and user-item signals based on the exploitation of the user-item graph structure, by propagating embeddings on it.

The selected CF models constitute a set of competitive baselines from various natures, representing a diverse array of recommendation approaches (classical, neural, graph-based). Additionally, the **TopPop** method is included as a non-informative baseline, as an essential baseline to provide a case where the outcomes are heavily biased towards popular items.

Hyperparameter Tuning. In our research, the primary focus is on the hyperparameter tuning of various collaborative filtering (CF) recommendation models, selected as baselines. These models are specifically tailored and evaluated within implicit feedback scenarios, where the subtleties of user preferences are inferred from indirect interactions. This approach ensures a comprehensive exploration of hyperparameter spaces to optimize model performance in these nuanced environments. The RecBole public library[3] is used for implementing and applying these models. Hyperparameter tuning is an essential step in optimizing the performance of these models. We used a bootstrapping sampling strategy, generating 1000 samples by repeatedly resampling our dataset with replacement.

The following is a summary of the hyperparameter tuning conducted for each model:

- **BPR-MF.** `Embedding size` choices `'[32, 64, 128]'` and `learning rate` choices `'[1e-4, 5e-4, 1e-3, 5e-3]'`. Total of 12 cases.
- **ItemKNN.** k choices `'[10, 50, 100, 200, 250, 300, 400]'` and `shrink` choices `'[0.0, 0.1, 0.5, 1, 2]'`. Total of 35 cases.
- **NGCF.** `Learning rate` choices `'[1e-4, 5e-4, 1e-3]'`, `hidden size list` choices `'["[64, 64, 64]", "[128, 128, 128]"]'`, `node dropout` choices `'[0.0, 0.1, 0.2]'`, `message dropout` choices `'[0.0, 0.1, 0.2]'`, and `regularization weight` choices `'[1e-5, 1e-3]'`. Total of 108 cases.
- **MultiVAE.** `Learning rate` choices `'[1e-4, 5e-4, 1e-3, 5e-3]'` and `latent dimension` choices `'[64, 128, 200, 300, 512]'`. Total of 20 cases.
- **LightGCN.** `Learning rate` choices `'[5e-4, 1e-3, 2e-3]'`, `number of layers` choices `'[1, 2, 3, 4]'`, and `regularization weight` choices `'[1e-05, 1e-04, 1e-03, 1e-02]'`. Total of 48 cases.

The total number of hyper-parameters is 14, encompassing a total of 223 experimental cases. This approach ensures that we effectively explore a wide range of hyperparameter configurations, particularly for models operating in implicit feedback scenarios, which are crucial for our research.

4.2 Experiment 2. Sequential Recommendation

4.2.1 Datasets and Setup. In this study, our objective is to understand the *generalizability* of the findings from the previous study and address aspects that were not studied previously. In particular, Experiment 2 focuses on the more practical task of **sequential recommendation** and the comparison of *zero-shot* vs. *few-shot in-context learning* (ICL), which we evaluate on datasets of different domains (movies and music). The choice of sequential recommendation, rather than classical **top-k**, was motivated by the research carried out in [38]. Therefore, the primary goal in Experiment 2 is not to assess the quality of different prompt scenarios (as carried out in Section 4.1), rather *to evaluate the effectiveness of zero-shot vs. few-shot ICL on the accuracy vs. item fairness of RS.*

We focus our attention on two main datasets from the movie and music domains: *MovieLens-1M* and *LastFM-1K*. These datasets are widely recognized in the academic community, and provide a basis for evaluating our models in terms of fairness across different types of media consumption data. Here, we opt to use a sub-sample of users instead of all, to speed up experiments. Initially, we randomly select a subset of *80 users* who exhibit a moderate level of interaction within the datasets. This allows us to handle the data efficiently while ensuring that the users selected have enough interactions to inform the training process. The data for these users is divided into training and test sets by sorting their interactions over time and splitting them so that 80% of a user interactions are used for training, with the remaining 20% held out for testing. This method respects the *chronological order* of interactions, thereby simulating a realistic scenario where a model can only learn from past data to make predictions about future user behavior.

[3] https://www.recbole.io/

Table 6. Statistics of the final datasets used in the sequential ICL task.

Dataset	|U|	|I|	|R|	$\frac{R}{U}$	$\frac{R}{I}$	Density	Item Gini	User Gini
LastFM-1K	80	5,500	277,607	3,470.09	50.47	98.6%	0.697	0.621
MovieLens 1M	80	3,900	20,987	262.34	5.38	99.2%	0.745	0.657

4.2.2 Baselines. In this experiment, we do not compare the built RecLLMs with traditional sequential models, as this would require a separate and labor-intensive set of experiments, which is beyond the scope of this study. Instead, we compare the fairness and accuracy of different versions of RecLLMs against each other. Additionally, in one part of the experiment, we explore the impact of the presence or absence of *user demographics* in the prompts on recommendation fairness and quality. Since CF baselines typically do not use such demographic information, we exclude them from this comparison.

4.2.3 Evaluation metrics. The first experiment employs a diverse range of metrics to assess the performance of our recommendation system. These metrics are categorized into two primary classes: *accuracy metrics*, and *item fairness*, and *catalogue coverage*.

Accuracy Metrics. For accuracy, we choose simpler metrics since the goal is not to compare them with strong CF baselines. We use the following two metrics to measure the quality of the top-k recommendations in sequential setting:

- **Hit Rate (HR@k):** Measures the proportion of users for whom at least one of the top-k recommended items is relevant. It is calculated as:

$$\text{HitRate@K} = \frac{|\{i \in R_Q : i \in \mathcal{G}\}|}{K}$$

  where R_Q represents the set of recommendations of the ranker in question and $\mathcal{G}$ represents the set of items in the ground truth.
- **Average Rank (AverageRank@K):** Measures the average position of relevant items in the recommendation list. It is calculated as:

$$\text{AverageRank@K} = \frac{1}{|\{i \in R_Q : i \in \mathcal{G}\}|} \sum_{i \in \mathcal{G}} \text{rank}_{R_Q}(i)$$

  where $\text{rank}_{R_Q}(i)$ is the rank of item i in the recommendation list R_Q.

High scores in these metrics indicate more accurate and relevant recommendations.

Item fairness, catalouge Coverage and long-tail distribution. Similar to the previous experiment, we use the **Gini Index**, **HHI**, and **Entropy** to measure item fairness. Additionally, we compute **catalogue coverage**, and display the **long-tail distribution** to visualize the concentration bias of recommendated items.

4.3 Parsing Output Recommendations and Finding Closest Match Items

Since language models (LLMs) naturally produce text-based outputs, we need a systematic approach to parse these outputs and map them to our existing dataset items. This ensures that when the RecLLM generates recommendations for which we do not find *an exact match* in the dataset, we can associate them with the most similar existing items, maintaining the integrity and relevance of our recommendation system.

- We start by utilizing the OpenAI GPT-3.5 API itself to parse the raw text of the recommendations. We designed a `LLM_parsing` function to extract the recommended songs and artists (or movies) from the text provided by the language model. For instance, we specify a system message to format the output in a structured way, ensuring that each recommendation follows the format: "*Song Name* by *Artist*." This structured output helps the subsequent matching process. The parsed recommendations are then collected, which contain cleaned and structured text.
- Once the recommendations are parsed, the next step involves finding the closest matches to these recommendations within our dataset. This is achieved using the `find_closest_match` function, which employs the `difflib.SequenceMatcher` from the Python standard library. This tool calculates **similarity scores** between the recommended song and artist names and the existing items in our dataset. For each recommendation, the function compares it against all items in the dataset, computing a combined score based on the similarity of both the track name and the artist name.
- A threshold is set to ensure that only matches with a high enough similarity score are considered valid. If the highest score exceeds this threshold, the corresponding item ID from the dataset is selected as the closest match. This method ensures robustness by preventing low-similarity matches from being accepted, thus maintaining the relevance and accuracy of the recommendations.

In summary, the implementation of these methods leverages essential Python tools and libraries, `openai` for API interaction, and `difflib` for sequence matching. These tools provide a robust and efficient framework for parsing and matching recommendations, ensuring that the RecLLM system can be effectively evaluated even when it generates items not originally present in the dataset.

5 Results and Discussion

In this section, we present the results and discussion for two experiments. We formulate hypotheses for each part, then showcase the results and provide analysis. Finally, we answer the hypotheses.

5.1 Experiment 1. Zero-Shot Top-k Recommendation

Through experiment 1, conducted in a zero-shot setting for the classical top-k recommendation, we aim to answer the following experimental research questions. Details about Experiment 1 can be found in Section 3.1 and Section 4.1.

We evaluate the average performance (Avg Perf.) in terms of NDCG and Recall across different conditions/models. We compare regular performance values (average) with bootstrap averages to gain deeper insights. The bootstrap method involves repeated resampling and offers a more robust understanding of performance variability. Table 7 illustrates this comparison. While the regular average provides a single estimate, the bootstrap mean extends this by offering confidence intervals, indicating the range within which the true performance measure is likely to fall.

Experimental Research Questions

RQ1. How does the incorporation of various goal-oriented prompts impact the **accuracy** (personalization) of GPT-based model in comparison to CF baselines? *(cf. Section 5.1.1)*

RQ2. What is the **stability** and **consistency** of the personalization performance metrics (e.g., NDCG, Recall) for GPT-based recommendation systems across multiple runs? How does this variability compare to CF baselines? *(cf. Section 5.1.2)*

RQ3. How do GPT-based recommendation systems compare to CF baselines in terms of provider fairness? Can the inclusion of a "Fair Recommender" system role might mitigate item unfairness and enhance diversity? *(cf. Sections 5.1.3*)

RQ4. Do GPT-based recommendation systems exhibit a bias towards recommending newer or older movies, or certain genre tendancy, compared to CF baselines, and how does the temporal freshness of recommendations hold? How does the inclusion of explicit user ratings in prompts affect the personalization performance? *(cf. Sections 5.1.4).*

RQ5. How do different algorithmic in-context learning (ICL) strategies, — in particular zero-shot vs. few-shot learning – impact the quality and biases of RecLLMs? What are the specific prompt design aspects that may influence this towards better or worse performance? *(cf. Section 5.2)*

RQ6. What are the economic and practical implications of using GPT-based models for recommendation systems, specifically in terms of inference costs and latency issues? *(cf. Section 5.3)*

5.1.1 Personalization of Recommendations. This section focuses on the personalization performance measured by recommendation top-K accuracy. Results are presented in Table 7, with each generative scenario (using ChatGPT) in the upper sections and CF baselines in the bottom section. Not every possible scenario combination was tested; rather, a select subset was examined.

Hypotheses. To investigate the effectiveness of different recommendation strategies, we propose the following hypotheses:

- **H1.** Incorporating goal-oriented prompts, including *personalization-based, beyond-accuracy*, and *reasoning-based prompts* (as detailed in scenarios **S1 to S7** in Section 3.1.1), with different system assignment roles, results in substantially varied performance outcomes in RecLLMs;
- **H2.** In terms of performance, GPT-based recommenders in *zero-shot setting* can provide comparable performance compared to collaborative filtering (CF) baselines.

Discussion. Our evaluation highlights several findings regarding the personalization of recommendations using RecLLM.

- First, there is a clear indication that incorporating the "system role" improves the performance of RecLLM, and this consistency is observed across most scenarios. For example, for a randomly chosen scenario S1, in **Tab1** and **Tab2** (with no system role assignment), the NDCG performances are 0.008803 and 0.010431, respectively. In contrast, **Tab3** and **Tab4** (with system role assignments, either as "act as a recommender system" or "act as a fair recommender system") show performance improvements to 0.013007 and 0.014268, respectively. This represents roughly a 45% improvement. In some scenarios, the improvement reaches up to 100%. It becomes thus evident that *the system role plays a crucial part in optimizing the effectiveness of recommendation systems.*
- Second, the performance varied considerably across different scenarios (S1 to S7), indicating that the context and nature of the prompts affect recommendation outcomes. For example, in **Tab4**, the "Simple" prompt scenario (S1) achieved an NDCG of 0.014, while the "Diversify" prompt (S4) had a lower NDCG of 0.002. The best-performing methods were **simple**, **reasoning-based** (COT and motivated reasoning), providing relatively better accuracy. Thus, based on these results, we may conclude that *requesting ChatGPT to diversify recommendation or to provide novel recommendations significantly reduces accuracy, which should be avoided or carefully considered.* Overall these variations underscore the importance of selecting appropriate prompts based on the desired recommendation attributes.
- Finally, comparing RecLLMs with baselines, our results reveal that RecLLM, even in zero-shot settings, generally performs lower than CF baselines, though in certain contexts, this difference shrinks considerably. For instance, the average NDCG of CF models such as BPR-MF and ItemKNN were 0.04776 and 0.04905, respectively. In comparison, the best-performing RecLLM prompt achieved an NDCG of 0.014, which is approximately 70% lower.

Answer to Hypotheses. To address the proposed hypotheses, we present the following findings:

- **H1.** ✓ **Supported** – The integration of specific system roles (e.g., "act as a recommender," or "act as fair recommender") leads to considerable performance improvements in RecLLMs.
- **H2.** × **Rejected** – Performance of RecLLMs in zero-shot settings consistently and significantly lags behind CF baselines.

5.1.2 Stability and Confidence of Personalization Metrics. As stated in Section 3.1.2, we considered the possibility that external factors such as trending data or algorithmic updates could introduce randomness into responses of GPT-based models. To study the impact of such randomness, each experiment was conducted five times specifically for GPT-based models.[4] In the analysis of this section, our goal is to observe the variability of GPT-based models across different runs, rather than comparing them to the CF baselines.

Hypotheses. To rigorously evaluate the robustness and consistency of GPT-based models under varying conditions, we propose the following hypothesis:

- **H3.** Running multiple iterations of GPT-based recommendation queries for a specific user, results in consistent recommendation quality.

Discussion. The graph in Figures 3 illustrates the NDCG and Recall scores for GPT-based recommendation models across five different runs. These runs overall have been conducted at different points in time that could span from *hours to a few days*. Notably, there is observable variability in the performance metrics from run to run. For instance,

- **Simple** (**S1**): The NDCG scores are [0.0088, 0.0091, 0.0095, 0.0094, 0.0095], with a std[5] of approximately 0.000305.
- **Motivate reasoning** (**S6**): The NDCG scores are [0.0101, 0.0097, 0.0093, 0.0094, 0.0095], with a std of approximately 0.000316.
- **COT** (**S7**): The NDCG scores are [0.0066, 0.0067, 0.0070, 0.0069, 0.0068], with a std of approximately 0.000158.

Despite the inherent randomness in individual iterations, the general tendency of the results remains consistent, suggesting that while individual runs may produce fluctuating scores, the overall behavior of the models does not significantly deviate. This is evidenced by the low standard deviations in the NDCG scores (0.000305 for **S1**, 0.000316 for **S6**, and 0.000158 for **S7**), indicating a high level of result stability over different runs. Such consistency aligns with the expected behavior of generative models, which can exhibit some degree of randomness in their output due to the stochastic nature of the underlying data/algorithms. However, the close clustering of these scores, as reflected in the narrow stds, indicates a stable pattern of performance.

While the current study demonstrates a high degree of stability in model results over multiple runs, it is crucial to note that these findings do not necessarily guarantee stability over longer time frames, such as weeks or months. The dynamic nature of data and algorithms in generative models means that updates and changes over time could lead to different behaviors and outputs. Nonetheless, *caution is required* when generalizing these results, and continuous monitoring and periodic re-evaluation of the performance over extended periods are recommended to ensure that the insights remain relevant and accurate.

[4]Note that prior to these experiments, we lacked specific knowledge about the performance of Chat-GPT prompts. Our experiments thus focused on a randomly selected case (**R0/E0** scenario).
[5]standard deviation

Table 7. Performance results measured in terms of recommendation Top-10 accuracy.

Tab 1 - System Role: No (R0), Emphasis: No (E0)

Prompt: Based on these movies: {user_movies_string}, recommend 10 movies that the user will likely enjoy.
System Role: -

Type	Model	Avg Perf. NDCG	Avg Perf. Recall	Bootstrap Mean NDCG	Bootstrap Mean Recall	Bootstrap Conf. NDCG Conf.	Bootstrap Conf. Recall Conf.
Generative (ChatGPT)	**S1.** Simple	0.008803	0.00994	0.008784	0.010011	(0.006323, 0.011484)	(0.00674, 0.013824)
	S2. Genre-focused	0.006964	0.007672	0.006994	0.007689	(0.004881, 0.009200)	(0.005079, 0.0106819)
	S4. Diversify	0.001906	0.002946	0.001896	0.002946	(0.000822, 0.003185)	(0.001559, 0.004582)
	S5. Surprise	0.007265	0.006968	0.007286	0.00696	(0.004671, 0.010279)	(0.004130, 0.010021)
	S6. Motivate Reasoning	0.010152	0.010906	0.010073	0.01098	(0.007446, 0.013124)	(0.007511, 0.0148920)
	S7. COT	0.006573	0.005848	0.006554	0.005821	(0.004513, 0.008940)	(0.003853, 0.008015)

Tab 2 - System Role: No (R0), Emphasis: Fair (E1)

Prompt: Ensure a fair representation of both popular and less-known movies. Based on these movies: {user_movies_string}, recommend ... .
System Role: -

Type	Model	Avg Perf. NDCG	Avg Perf. Recall	Avg. Bootstrap NDCG	Avg. Bootstrap Recall	Bootstrap Conf. NDCG	Bootstrap Conf. Recall
Generative (ChatGPT)	**S1.** Simple	0.010431	0.010557	0.010404	0.010579	(0.007491, 0.013792)	(0.007295, 0.014063)
	S2. Genre-focused	0.005221	0.005471	0.005211	0.00555	(0.003413, 0.007300)	(0.003361, 0.008196)
	S4. Diversify	0.001717	0.003029	0.001713	0.003024	(0.000607, 0.003261)	(0.001504, 0.005083)
	S5. Surprise	0.004095	0.005089	0.004059	0.005112	(0.002399, 0.006132)	(0.002645, 0.008248)
	S6. Motivate Reasoning	0.004315	0.005097	0.004302	0.005031	(0.002246, 0.006819)	(0.002668, 0.007909)
	S7. COT	0.007207	0.009344	0.007177	0.009373	(0.004493, 0.010162)	(0.005808, 0.013662)

Tab 3 - System Role: Normal Recommender (R1), Emphasis: Fair (E1)

Prompt: Based on these movies: {user_movies_string}, recommend 10 movies that the user will likely enjoy.
System Role: Act as a fair recommender system balancing between popular and less-known movies to ensure provider fairness.

Type	Model	Avg Perf. NDCG	Avg Perf. Recall	Avg. Bootstrap NDCG	Avg. Bootstrap Recall	Bootstrap Conf. NDCG Conf.	Bootstrap Conf. Recall Conf.
Generative (ChatGPT)	**S1.** Simple	0.013007	0.012606	0.012984	0.012514	(0.009799, 0.016379)	(0.008610, 0.016971)
	S2. Genre-focused	0.00854	0.00987	0.008582	0.00983	(0.006277, 0.010984)	(0.006564, 0.013434)
	S4. Diversify	0.00314	0.005606	0.003162	0.005523	(0.001477, 0.005548)	(0.003086, 0.008230)
	S5. Surprise	0.006473	0.006435	0.006403	0.00642	(0.004065, 0.009319)	(0.003627, 0.009950)
	S6. Motivate Reasoning	0.009379	0.00957	0.009447	0.009496	(0.006160, 0.012722)	(0.006056, 0.013061)
	S7. COT	0.013571	0.012887	0.013622	0.012941	(0.009836, 0.018061)	(0.009306, 0.017057)

Tab 4 - System Role: Fair Recommender (R2), Emphasis: No (E0)

Prompt: *Ensure a fair representation of both popular and less-known movies. Based on these movies: {user_movies_string}, recommend ... .*
System Role: *Given a user, act as recommender system.*

Type	Model	Avg Perf. NDCG	Avg Perf. Recall	Avg. Bootstrap NDCG	Avg. Bootstrap Recall	Bootstrap Conf. NDCG Conf.	Bootstrap Conf. Recall Conf.
Generative (ChatGPT)	**S1.** Simple	0.014268	0.015796	0.014257	0.015731	(0.011243, 0.017614)	(0.011896, 0.019893)
	S2. Genre-focused	0.008847	0.01132	0.00883	0.011382	(0.006700, 0.011233)	(0.007998, 0.015372)
	S4. Diversify	0.002844	0.004901	0.002825	0.004872	(0.001303, 0.004922)	(0.003205, 0.006746)
	S5. Surprise	0.007504	0.00729	0.007446	0.007304	(0.004587, 0.010622)	(0.004066, 0.011234)
	S6. Motivate Reasoning	0.012354	0.011968	0.012407	0.011929	(0.008901, 0.016671)	(0.008133, 0.016052)
	S7. COT	0.013212	0.012862	0.013162	0.012908	(0.009972, 0.017185)	(0.009256, 0.016893)
CF Baselines							
Discriminative	BPR-MF	0.04776	0.060699	0.047742	0.060537	(0.040465, 0.054958)	(0.050181, 0.071453)
	ItemKNN	0.04905	0.065558	0.048995	0.065367	(0.041122, 0.056398)	(0.054682, 0.076526)
	NGCF	0.04445	0.053534	0.044376	0.053448	(0.037306, 0.051784)	(0.044374, 0.063413)
	VAE	0.047158	0.056589	0.047187	0.056502	(0.039967, 0.054677)	(0.047093, 0.065803)
	LightGCN	0.048295	0.059108	0.048215	0.059019	(0.041621, 0.055491)	(0.049809, 0.068851)
	TopPop	0.030556	0.032327	0.030539	0.032127	(0.024253, 0.036734)	(0.025213, 0.039704)

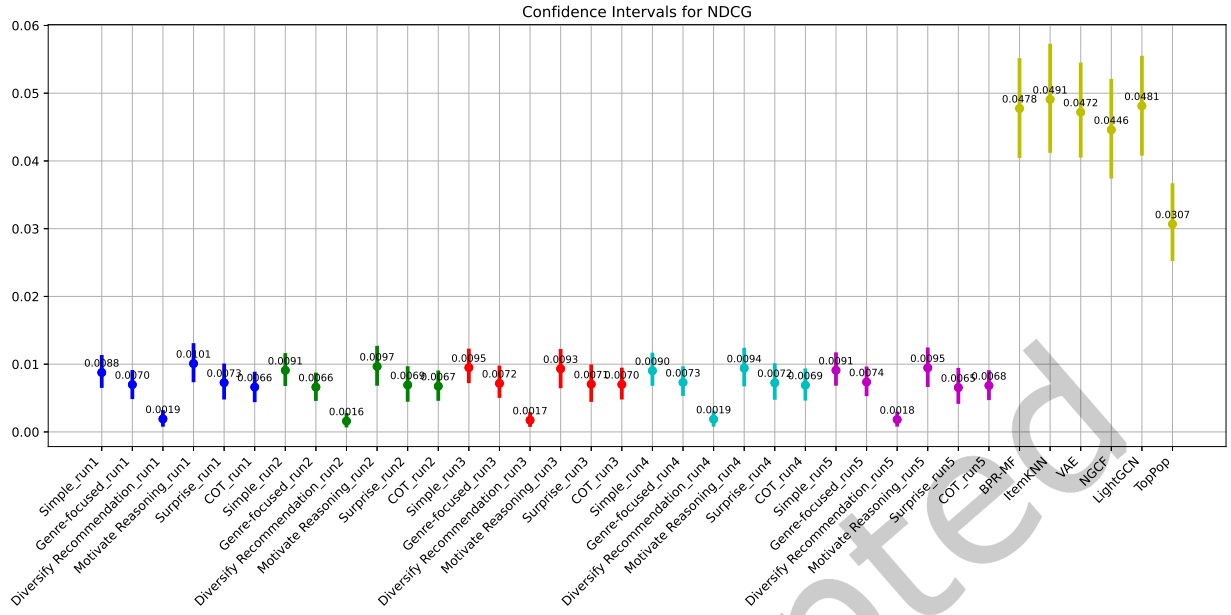
Answer to hypotheses. To evaluate the validity of our proposed hypotheses, we present detailed findings below, supported by statistical data and analyses:

- **H3.** ✓ **Supported** – This hypothesis, which anticipated consistent performance across different GPT-based model runs, is supported. The data presented in the NDCG graph aligns with this expectation, showing only minor variability across multiple runs. Confidence intervals overlap significantly for different runs of each GPT-based model, which confirms the reliability of the system in providing personalized recommendations. However, it is important to consider these results with caution for long-term deployment as they do not necessarily guarantee stability over extended periods such as weeks or months.

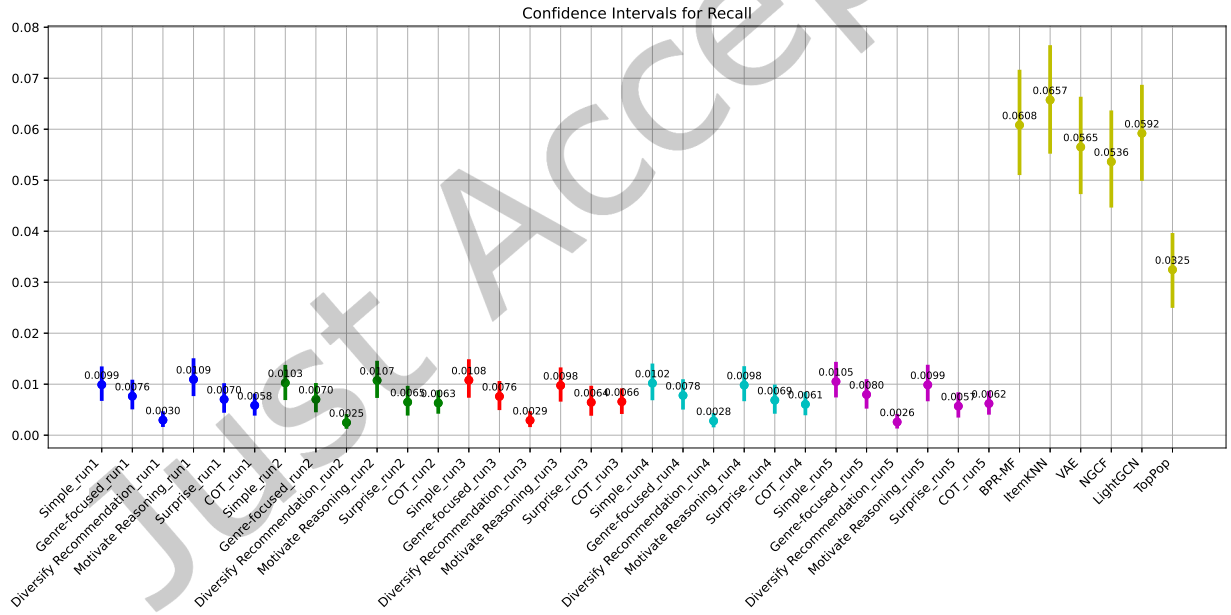
5.1.3 Item Fairness. This section explores the item fairness metrics of different recommender models, crucial for evaluating their effectiveness in providing balanced and varied recommendations.

Hypotheses. To ascertain the impact of different model strategies on recommendation fairness, we carefully analyzed the outcomes in the context of our formulated hypotheses:

- **H4:** Due to their training on large-scale internet data, GPT-based models are hypothesized to exhibit lower item fairness than classical CF models. However, asking the system to produce more diverse or additional recommendations (scenarios **S3** and **S4**) is expected to increase/improve item fairness.
- **H5:** Integrating a "Fair Recommender" as a system role in GPT-based systems is expected to enhance item fairness.



(a) NDCG@10.



(b) Recall@10

Fig. 3. Studying the stability of GPT-based performance metric across different runs.

Results and Discussion. We now address the above research hypotheses, using Table 8 to inform our discussion. We primarily use the Gini coefficient and entropy for this analysis, noting that HHI shows somewhat similar performance to the Gini coefficient. Overall, by examining the Gini indices and entropy values, we observe that two CF models, Item-KNN and NGCF, provide the best performances in terms of item fairness. Among the RecLLM models, the *Genre-focused, Motivated Reasoning*, and *Chain-of-thought* scenarios exhibit notable item fairness after the CF baselines.

Summarizing the best-performing models, we have: $ItemKNN/NGCF > Genre/Motivate/COT > rest$. One notable and interesting result from the table is that *asking GPT to produce novel or diverse recommendations has little or no significant impact on improving its recommendability.* The entropy generally increases for RecLLM models when switching to Fair recommenders, indicating higher novelty in recommendations. This pattern is not observed in the CF models, which show no change in their metrics. To address **H4**, we find that GPT-based models generally exhibit lower item fairness compared to classical CF models, as hypothesized. However, contrary to the expectation, asking the system to produce more diverse or additional recommendations did not significantly increase item fairness. For **H5**, integrating a "Fair Recommender" role in GPT-based

Table 8. Provider fairness of the tested models in Experiment 1

Model	Normal Recommender			Fair Recommender		
	Gini Coefficient ↓	HHI ↓	Entropy↑	Gini Coefficient ↓	HHI ↓	Entropy↑
Simple	0.982463	0.017204	5.042821	0.978925	0.010899	5.387465
Genre-focused	0.964743	0.006455	5.919697	0.959879	0.004771	6.110040
Diversify Recommendation	0.992349	0.034724	4.232139	0.992603	0.030010	4.321307
Surprise	0.997906	0.059857	3.227737	0.998365	0.067952	3.023948
Motivate Reasoning	0.981745	0.019189	5.026322	0.979133	0.011218	5.366627
Chain-of-thought (COT)	0.986889	0.027030	4.619500	0.979313	0.011167	5.365294
BPR-MF	0.991758	0.012550	4.658056	0.991758	0.012550	4.658056
Item-KNN	0.914271	0.002877	6.671847	0.914271	0.002877	6.671847
NGCF	0.950845	0.002762	6.420996	0.950845	0.002762	6.420996
VAE	0.989722	0.009554	4.903511	0.989722	0.009554	4.903511
LightGCN	0.989610	0.010546	4.861879	0.989610	0.010546	4.861879
TopPop	0.994859	0.020000	3.912023	0.994859	0.020000	3.912023

Cyan shows the best performing methods.
Green shows good performing methods (relative to others).
Yellow shows lower performance models.

systems did enhance recommendation fairness, as indicated by the increased entropy in the RecLLM models. This suggests a mitigation of item unfairness and an enhancement of recommendation diversity.

Answer to Hypotheses. We can answer to the above hypotheses as following:

- **H4:** ○ **Partially Supported** — While classical CF models such as Item-KNN and NGCF exhibit higher fairness compared to RecLLM models, the GPT-based models named *"Genre-focused"* and *"Motivated Reasoning"* showed improvements in fairness. This hypothesis is partially supported as diversification strategies and requests for more novel recommendations did not increase item fairness in RecLLM models.
- **H5:** ✓ **Supported** — Implementing a *"Fair Recommender"* role within GPT-based systems leads to noticeable improvements in fairness. This supports the hypothesis that transitioning from normal to fair recommender systems enhances fairness, demonstrating the controllability in RecLLM models, particularly in terms of diversity and novelty.

5.1.4 Analysis of Temporal Freshness and Genre Bias in Recommender Systems. In this unified analysis, we explore the tendencies of various GPT-based and CF models in terms of recency and genre preferences in movie recommendations. Our study aims to discern whether these models demonstrate a bias towards recommending newer or older films and if they exhibit specific genre preferences that could illuminate performance differences between model types.

Research Hypotheses. We provide the following statement of research hypotheses:

- **H6.** We hypothesize that GPT-based models, given their extensive training on diverse and recent datasets, are more likely to recommend **newer films** compared to classical CF models, which may favor older, well-established films.
- **H7.** We expect that GPT-based models will show a broader genre distribution in their recommendations due to their training on diverse data sources, compared to CF models, which may adhere more strictly to users' historical genre preferences.

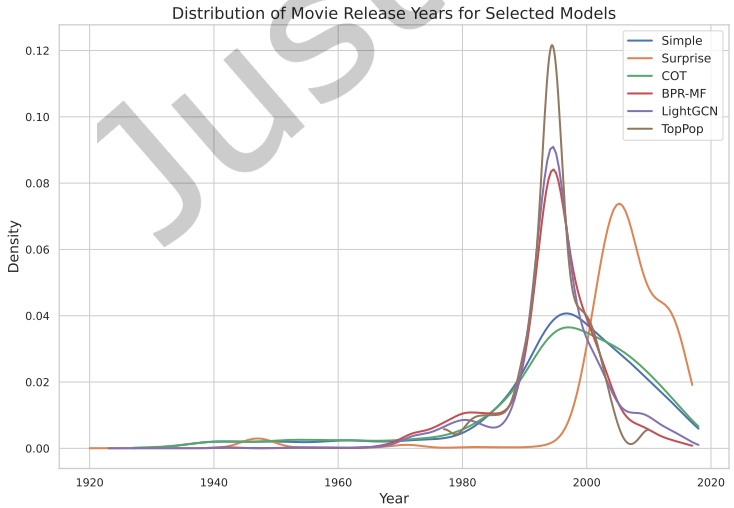


Fig. 4. Distribution of Movie Release Years as recommended by different models

Table 9. Model Statistics Reflecting Recency Bias

Model	Median Year	Std Year
Simple	1999	15.09
Genre-focused	1997	14.99
Diversify	2007	11.76
Surprise	2006	10.62
Motivate reasoning	2002	15.71
COT	1999	15.82
BPR-MF	1995	8.27
ItemKNN	1995	12.14
NGCF	1995	12.25
VAE	1995	8.50
LightGCN	1995	8.34
Pop	1995	5.65

Results and Discussions. Table 9 presents the statistics that inform our understanding of each model tendency towards recommending newer or older films. Notably, the GPT-based models 'Surprise' and 'Diversify' push the boundaries towards recommending more recent items, with median release years of 2006 and 2007. This indicates a clear orientation towards newer movies, possibly interpreting 'surprise' and 'diversification' in their temporal sense. Since GPT models do not possess inherent information about the novelty of content in terms of user interactions, they likely interpret these directives based on the freshness of the content itself. In contrast, the CF models tend to recommend older films, median release years consistently at 1995, reflecting a stronger inclination towards well-established, historically popular films. This distinction suggests that GPT-based models might be leveraging their extensive training datasets to prioritize more recent films, which can be beneficial in scenarios where up-to-date recommendations are desired. The broader temporal range of recommendations from GPT-based models, as evidenced by the standard deviation in the year of movies recommended (Surprise: Std Year 10.62, Diversify: Std Year 11.76), further underscores their flexibility in catering to diverse temporal tastes.

Upon analyzing the word clouds, it is evident that CF models such as **BPR-MF**, **LightGCN**, and **VAE** predominantly recommend genres such as "Action," "Adventure," "Thriller," and "Sci-Fi." These genres appear frequently and in larger font sizes, indicating their dominance in the recommendations made by these models. In contrast, GPT-based models, represented in the lower word clouds, show a more varied genre distribution with "Comedy," "Drama," and "Romance" and even "Thriller" and "Adventure" being more prominent. These results are interesting and suggest that GPT-based RecLLMs may be less constrained by the users' historical genre affinities and more exploratory in their recommendations, possibly leading to a broader but potentially less precise genre match to individual user profiles.

- **H6.** ✓ **Supported** – GPT-based models, particularly *Diversify* and *Surprise*, tend to recommend significantly newer movies than CF models. This finding supports the hypothesis, confirming that GPT-based models are effective in capturing recent trends and offering more temporally relevant recommendations.
- **H7.** ○ **Partially Supported** – While CF models are found to predominantly recommend genres such as *Action* and *Sci-Fi*, GPT-based models demonstrate a broader genre distribution, including *Comedy*, *Drama*, and *Romance*. This diversity in GPT-based recommendations supports the hypothesis to a certain extent, indicating that these models are utilizing their extensive training on diverse datasets to introduce a wider variety of content.

Implicit vs. explicit scenario. Analyzing the impact of including ratings in the prompt shows noteworthy results. In our study, we initially avoided using ratings for the GPT models, aligning with the implicit setting of the Collaborative Filtering (CF) baseline. However, for completeness, we explored the effect of revealing user ratings for selected items.

The "Include-rating" model demonstrates a significant improvement compared to other GPT-based scenarios. With an NDCG of 0.013230 and a Recall of 0.014721, it surpasses genre-focused, Diversify Recommendation, Surprise, Motivate Reasoning, and chain-of-thought (COT) models. The Bootstrap NDCG and Recall for the Include-rating model are 0.013280 and 0.014673, respectively, indicating consistency in performance. The confidence intervals for NDCG (0.009737 to 0.017012) and Recall (0.010261 to 0.019795) further emphasize its reliability.

These results suggest that including ratings significantly enhances the recommendation performance of the system. It provides a more personalized and accurate reflection of user preferences, which is a key aspect in recommendation systems. While other GPT-based models showed varied performance, the inclusion of ratings seems to offer a more consistent and improved outcome. This insight could be valuable in refining other scenarios and models, potentially leading to better overall system performance.

5.2 Sequential Recommendations

This section focuses on the performance of RecLLMs in sequential recommendation task, measured by recommendation top-K accuracy, item fairness, and coverage. Results are presented in Table 10, with each generative scenario (using ChatGPT) in the upper sections and CF baselines in the bottom section.

Hypotheses. To investigate the effectiveness of different recommendation strategies, we propose the following hypotheses:

- **H1.** Incorporating sequential in-context learning (ICL) prompts, including *ICL-1* and *ICL-2*, results in improved recommendation accuracy compared to zero-shot learning.
- **H2.** Sequential in-context learning (ICL) prompts will result in better item fairness and coverage compared to zero-shot learning.
- **H3.** A combination of improved accuracy, item fairness, and coverage will be observed in certain settings, highlighting the optimal recommendation strategy.

Fig. 5. WordCloud of Movie Genres as recommended by different models. Top model correspond to CF models (**BPR-MF**, **LightGCN**, **RecVAE**), while lower models include GPT-based recommenders (**Simple**, **Diversity**, **COT**).)

Table 10. Recommendation alignment between $(\mathcal{R}_m, \mathcal{R}_m^a)$ based on Item Similarity β_{item}

Attribute	Type	ICL	MovieLens					LastFM				
			Acc.		Item Fairness		Cov.	Acc.		Item Fairness		Cov.
			HR	NDCG	Gini	Entropy	-	HR	NDCG	Gini	Entropy	-
No Info	Freq.	0-shot	0.179	0.53750	0.57739	3.47952	0.02621	0.204	0.66875	0.24882	4.89089	0.00195
		ICL-1	0.154	0.55000	0.66303	2.87493	0.01685	0.154	0.47500	0.24464	4.91536	0.00201
		ICL-2	0.150	0.40000	0.65835	2.88530	0.01685	0.154	0.44375	0.25137	4.92419	0.00201
	Rec.-Freq.	0-shot	0.208	0.65000	0.55075	3.66855	0.03033	0.225	0.65000	0.18229	5.07136	0.00220
		ICL-1	0.162	0.54375	0.60812	3.08940	0.01760	0.183	0.53750	0.20217	4.99446	0.00206
		ICL-2	0.175	0.56875	0.64473	2.92146	0.01610	0.208	0.58125	0.20439	5.00125	0.00205
Gender	Freq.	0-shot	0.146	0.58750	0.56064	3.60813	0.02845	0.196	0.68750	0.21299	5.00305	0.00211
		ICL-1	0.167	0.50625	0.67541	2.71790	0.01460	0.158	0.51875	0.26066	4.82855	0.00188
		ICL-2	0.171	0.43125	0.66145	2.76958	0.01498	0.154	0.42500	0.22654	4.99862	0.00215
	Rec.-Freq.	0-shot	0.158	0.55000	0.54779	3.68560	0.02958	0.221	0.64375	0.18254	5.08343	0.00223
		ICL-1	0.154	0.46250	0.64715	2.99814	0.01909	0.208	0.59375	0.19480	5.01322	0.00209
		ICL-2	0.175	0.50625	0.64839	2.73183	0.01310	0.204	0.65625	0.18622	5.06197	0.00218
Age-Group	Freq.	0-shot	0.154	0.70000	0.54193	3.72305	0.03145	0.183	0.56875	0.24498	4.91475	0.00198
		ICL-1	0.175	0.57500	0.67186	2.73694	0.01498	0.146	0.48750	0.24892	4.90102	0.00199
		ICL-2	0.150	0.45625	0.64637	2.87851	0.01572	0.146	0.41250	0.23557	4.97022	0.00211
	Rec.-Freq.	0-shot	0.171	0.66875	0.56433	3.63473	0.02995	0.263	0.82500	0.18816	5.07481	0.00223
		ICL-1	0.158	0.46875	0.63580	2.92408	0.01610	0.188	0.56250	0.20575	4.96456	0.00200
		ICL-2	0.204	0.60000	0.63038	2.93595	0.01498	0.204	0.60000	0.19985	5.01733	0.00210

Cyan shows the best performing methods.
Green shows good performing methods (relative to others).
Yellow shows lower performance models.

Discussion. Our evaluation highlights several findings regarding the personalization of recommendations using RecLLM.

- Regarding **H1**, the results indicate that recommendation **accuracy,** measured by Hit Rate (HR) and NDCG, varies with different learning strategies. For the `MovieLens` dataset, *without demographic information*, the HR for zero-shot learning is 0.179, which decreases slightly for ICL-1 (0.154) and ICL-2 (0.150). In the `LastFM` dataset, zero-shot learning HR is 0.204, which decreases to 0.154 for both ICL-1 and ICL-2.
  However, *including demographic information*, especially in the `Age-Group` category, improves accuracy in the movie domain. For instance, in the `MovieLens` dataset with age-group context, the HR for ICL-2 improves to 0.204, which is higher than zero-shot learning with such information (0.171) or zero-shot learning with no demographic information (0.179). Another clear trend is the impact of the Recency-Focused (Rec.-Freq) sampling in the context part on recommendation accuracy. When sensitive attributes are included, Rec-Freq and ICL-2 generally show better performance compared to the same age-category included in zero-shot learning. For example:
  - In the `MovieLens` dataset with age-group context, the HR for Rec-Freq ICL-2 is 0.204, compared to 0.171 for zero-shot learning.
  - In the `MovieLens` dataset with gender context, the HR for Rec-Freq ICL-2 is 0.175, compared to 0.158 for zero-shot learning.
  - In the `LastFM` dataset with age-group context, the HR for Rec-Freq ICL-2 is 0.204, compared to 0.263 for zero-shot learning.
  - In the `LastFM` dataset with gender context, the HR for Rec-Freq ICL-2 is 0.204, compared to 0.221 for zero-shot learning.

  In terms of NDCG, however, results generally favor zero-shot learning scenarios across different combinations of demographic information and sampling strategies in both datasets. While there is a slight improvement in NDCG for the Rec-Freq ICL-2 scenario in the `LastFM` dataset compared to other scenarios (zero-shot or ICL-1), the differences are marginal. For example:
  - In the `LastFM` dataset with age-group context, the NDCG for Rec-Freq ICL-2 is 0.60000, compared to 0.56250 for ICL-1, and 0.82500 for zero-shot learning.
  - In the `LastFM` dataset with gender context, the NDCG for Rec-Freq ICL-2 is 0.65625, compared to 0.59375 for ICL-1, and 0.64375 for zero-shot learning.

  These examples illustrate that incorporating sensitive attributes and employing the Rec-Freq sampling method in ICL-2 can enhance recommendation accuracy, although the improvements in NDCG are not as pronounced.

- Regarding **H2**, **item fairness and coverage** show mixed results across different settings. For item fairness, as measured by Gini and Entropy, zero-shot learning and ICL strategies show slight differences. For instance, in the `MovieLens` dataset with no demographic info, Gini for zero-shot learning is 0.57739 (lower is better), which increases to 0.66303 for ICL-1 and 0.65835 for ICL-2, indicating worse fairness. Entropy, where higher values are better, decreases from 3.47952 (zero-shot) to 2.87493 (ICL-1) and 2.88530 (ICL-2), suggesting that ICL prompts might lead to slightly worse item fairness. **Coverage** remains consistently low across all scenarios in the movie domain, and zero-shot learning provides considerably better coverage performance. For example:
  - In the `MovieLens` dataset without demographic information, the coverage for zero-shot learning is 0.02621, compared to 0.01685 for ICL-1 and ICL-2.
  - In the `MovieLens` dataset with gender context, the coverage for zero-shot learning is 0.02845, compared to 0.01498 for ICL-2.

  In the music domain, results generally favor ICL-2. For example:
  - In the `LastFM` dataset with age-group context, the coverage for Rec-Freq ICL-2 is 0.00210, compared to 0.00198 for zero-shot learning.
  - In the `LastFM` dataset with gender context, the coverage for Rec-Freq ICL-2 is 0.00218, compared to 0.00211 for zero-shot learning.

  Notably, including demographic information generally enhances fairness metrics. For example, with `Gender` context in `MovieLens`, the Gini index improves, and Entropy increases, indicating fairer item distribution.
- Regarding **H3**, a combination of improved accuracy, item fairness, and coverage is generally observed with the `Recency-Focused` approach in both zero-shot and ICL-2 scenarios. By visually examining the placements of highlighted values, we note that `Recency-Focused` zero-shot and ICL-2 tend to provide the best combined performance of accuracy, item fairness, and coverage. Additionally, revealing demographic information, particularly age-group (young/old) and gender, has a noticeable positive impact on the fairness and coverage of the methods, thereby enhancing the utility of the system from both consumer and producer perspective.

Answer to Hypotheses. To address the proposed hypotheses, we present the following findings:

- **H1.** ○ **Partially Supported** – The incorporation of sequential in-context learning prompts (ICL-1 and ICL-2) improves recommendation accuracy in certain contexts, particularly with demographic information (e.g., `Age-Group` in `MovieLens`), though not consistently across all settings.
- **H2.** × **Rejected** – Sequential in-context learning prompts do not consistently lead to better item fairness and coverage compared to zero-shot learning. In some cases, they worsen item fairness, as indicated by higher Gini indices and lower Entropy values.

- **H3.** ✓ **Supported** – Certain settings, particularly `Recency-Focused ICL-2` with demographic information, show a balance between accuracy and item fairness, indicating potential optimal recommendation strategies.

5.3 Economic and Practical Implications

Integrating large language models (LLMs) like GPT-3.5-turbo into recommender systems involves several economic and practical considerations. Below, we discuss the inference costs, latency issues, and potential mitigation strategies.

Inference Costs

Using LLMs involves significant API costs. For example, the cost per API call can be calculated as:

$$\text{Cost per call} = \left(\frac{\text{tokens per call}}{1000}\right) \times \alpha$$

where α is the cost per 1000 tokens. For a system with n users and m prompts, the total cost C is:

$$C = n \times m \times \text{Cost per call}$$

For example, if an average recommendation call uses 1500 tokens, the cost per call is 0.03. For 610 users with 7 prompts each, the total cost would be 128.10. To mitigate these costs, we can explore strategies such as token optimization, batch processing, and selective use of LLMs. Our code implements these strategies by limiting the maximum tokens per response and grouping prompts to minimize the number of API calls.

Latency

Latency, the time taken to generate recommendations, affects user experience. Observed latencies ranged from a *few to over ten seconds per call.* Mitigation strategies include asynchronous processing to handle multiple requests concurrently, caching mechanisms to store common recommendations, and using faster, distilled versions of LLMs. Our code addresses latency by measuring and optimizing the time taken for each API call and incorporating mechanisms to batch and streamline requests. Additionally, practical considerations such as using cloud services for scalability, combining LLMs with traditional collaborative filtering models, and regularly evaluating performance metrics ensure the system remains efficient and cost-effective.

Summary.

Answers to Research Questions for Experiment 1 and 2

Answer to RQ1: The incorporation of various goal-oriented prompts impacts the accuracy of GPT-based models, with some prompts enhancing performance while others, like diversity-focused prompts, reducing it. However, GPT-based models generally lag behind CF baselines in zero-shot settings.

Answer to RQ2: GPT-based recommendation systems show consistent personalization performance across multiple runs, with minor variability. This stability is comparable to CF baselines, indicating reliable behavior despite inherent randomness in individual runs.

Answer to RQ3: GPT-based recommendation systems exhibit lower provider fairness compared to CF baselines. However, incorporating a "Fair Recommender" system role in GPT-based models improves fairness and diversity, demonstrating controllability and potential for mitigation of item unfairness.

Answer to RQ4: GPT-based models tend to recommend newer movies and show a broader genre distribution compared to CF baselines, which prefer older, well-established films. Including explicit user ratings in prompts enhances personalization performance significantly.

Answer to RQ5: Different ICL strategies impact RecLLM quality and biases variably, with zero-shot learning generally providing better accuracy, while ICL strategies, especially with demographic information, offer improvements in certain contexts. Prompt design plays a crucial role in these outcomes.

Answer to RQ6: The economic and practical implications of using GPT-based models include significant inference costs and latency issues. Mitigation strategies like token optimization, batch processing, and using distilled models can help manage these challenges, ensuring efficient and cost-effective system performance.

6 Conclusion and Future Directions

This research provides a comprehensive analysis of biases of Recommender Systems using Large Language Models (RecLLMs), specifically focusing on ChatGPT-based systems. Our study highlights the distinct capabilities and biases of these systems compared to traditional Collaborative Filtering (CF) methods within the domain of movie and music recommendations. The experimental results emphasize the significant impact of *prompt design* strategies on various aspects of recommendation quality, including accuracy, provider fairness, catalog coverage, diversity, temporal stability, genre dominance, and recency.

In particular, we conducted two experiments using different datasets to evaluate the effectiveness of these strategies: classical top-K recommendations and sequential in-context learning (ICL).

Key contributions and findings of our work include:

- **Enhanced Fairness and Diversity through Role-Based Prompts:** Utilizing role-based prompts (e.g., "act as a recommender" or "act as a fair recommender") significantly enhances fairness and diversity in recommendations, effectively mitigating item unfairness.
- **Sequential In-Context Learning (ICL):** Incorporating sequential in-context learning prompts (ICL-1 and ICL-2) shows improvements in recommendation accuracy in certain contexts, particularly with demographic information. For instance, in the MovieLens dataset with age-group context, the Hit Rate (HR) for ICL-2 improves compared to zero-shot learning. However, ICL prompts do not consistently lead to better item fairness and coverage compared to zero-shot learning.
- **Improved Personalization with Ratings:** Incorporating user ratings into prompts improves the personalization of recommendations, highlighting the potential for fine-tuning prompts to better reflect user preferences.
- **Temporal Bias towards Recent Releases:** GPT-based models exhibit a strong tendency to recommend newer movies, particularly those released post-2000, contrasting with the older preferences of CF models. This suggests that GPT models can capture recent trends better, making them more temporally relevant.
- **Item Fairness and Diversity:** While GPT-based models generally exhibit lower item fairness compared to classical CF models, scenarios involving fair recommender roles show improved fairness and diversity, as indicated by higher entropy values.
- **Stability in Performance Metrics:** Despite inherent randomness, GPT-based models show consistent performance across multiple runs, as evidenced by low standard deviations in metrics such as NDCG. This stability is crucial for reliable recommendation systems.

Our research identifies several key areas for further exploration that can push forward the capabilities and effectiveness of recommendation systems leveraging large language models (RecLLMs). Expanding research to include various content domains such as books, music, and e-commerce, along with conducting long-term user engagement studies, will provide deeper insights into this research. Firstly, refining prompt design and enhancing generalization across various datasets and tasks is a crucial step to improving recommendation accuracy and relevance. Additionally, implementing more advanced few-shot learning techniques, such as carefully selecting which examples to use and how to present them, can enhance the contextual relevance and accuracy of recommendations. These examples could be chosen from similar users (peer users) to make the system behave more like CFg models, thereby combining the strengths of both approaches.

Moreover, integrating GPT with collaborative filtering (CF) in a Retrieval-Augmented Generation (RAG) framework could leverage the strengths of both methodologies, leading to more robust and effective recommender systems. Future work should also address additional fairness dimensions, such as ensuring consumer-side fairness to prevent

recommendations from disproportionately favoring certain user groups. Calibration, or aligning recommendations with user expectations to offer a balanced mix of familiar and novel items, is another critical aspect. Furthermore, ensuring counterfactual fairness, where recommendations remain fair even when user attributes or behaviors are altered, is also an essential and interesting future direction.

We explicitly acknowledge the limitations of our current work in using a single LLM based on ChatGPT. Future studies should explore other large language models (LLMs) and compare their performance to further understand their potential and limitations in diverse recommendation scenarios.

References

[1] Himan Abdollahpouri and Robin Burke. 2021. Multistakeholder recommender systems. In *Recommender systems handbook*. Springer, 647–677.

[2] Enrique Amigó, Yashar Deldjoo, Stefano Mizzaro, and Alejandro Bellogín. 2023. A unifying and general account of fairness measurement in recommender systems. *Information Processing & Management* 60, 1 (2023), 103115.

[3] Akari Asai, Mohammadreza Salehi, Matthew E Peters, and Hannaneh Hajishirzi. 2022. Attempt: Parameter-efficient multi-task tuning via attentional mixtures of soft prompts. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*. 6655–6672.

[4] Giovanni Maria Biancofiore, Yashar Deldjoo, Tommaso Di Noia, Eugenio Di Sciascio, and Fedelucio Narducci. 2024. Interactive question answering systems: Literature review. *Comput. Surveys* 56, 9 (2024), 1–38.

[5] Toine Bogers and Marijn Koolen. 2017. Defining and supporting narrative-driven recommendation. In *Proceedings of the eleventh ACM conference on recommender systems*. 238–242.

[6] Ludovico Boratto, Gianni Fenu, and Mirko Marras. 2021. Interplay between upsampling and regularization for provider fairness in recommender systems. *User Modeling and User-Adapted Interaction* 31, 3 (2021), 421–455.

[7] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems* 33 (2020), 1877–1901.

[8] Sébastien Bubeck, Varun Chandrasekaran, Ronen Eldan, Johannes Gehrke, Eric Horvitz, Ece Kamar, Peter Lee, Yin Tat Lee, Yuanzhi Li, Scott Lundberg, Harsha Nori, Hamid Palangi, Marco Tulio Ribeiro, and Yi Zhang. 2023. Sparks of Artificial General Intelligence: Early experiments with GPT-4. arXiv:2303.12712 [cs.CL]

[9] Robin Burke, Nasim Sonboli, and Aldo Ordonez-Gauger. 2018. Balanced neighborhoods for multi-sided fairness in recommendation. In *Conference on fairness, accountability and transparency*. PMLR, 202–214.

[10] Dong-Kyu Chae, Jin-Soo Kang, Sang-Wook Kim, and Jaeho Choi. 2019. Rating augmentation with generative adversarial networks towards accurate collaborative filtering. In *The World Wide Web Conference*. 2616–2622.

[11] Abhijnan Chakraborty, Aniko Hannak, Asia J Biega, and Krishna P Gummadi. 2017. Fair sharing for sharing economy platforms. (2017).

[12] Zheng Chu, Jingchang Chen, Qianglong Chen, Weijiang Yu, Tao He, Haotian Wang, Weihua Peng, Ming Liu, Bing Qin, and Ting Liu. 2023. A survey of chain of thought reasoning: Advances, frontiers and future. *arXiv preprint arXiv:2309.15402* (2023).

[13] Diego Corrêa da Silva, Marcelo Garcia Manzato, and Frederico Araújo Durão. 2021. Exploiting personalized calibration and metrics for fairness recommendation. *Expert Systems with Applications* 181 (2021), 115112.

[14] Yashar Deldjoo. 2023. Fairness of ChatGPT and the Role of Explainable-Guided Prompts. *arXiv preprint arXiv:2307.11761* (2023).

[15] Yashar Deldjoo. 2024. FairEvalLLM. A Comprehensive Framework for Benchmarking Fairness in Large Language Model Recommender Systems. *arXiv preprint arXiv:2405.02219* (2024).

[16] Yashar Deldjoo, Vito Walter Anelli, Hamed Zamani, Alejandro Bellogin, and Tommaso Di Noia. 2021. A flexible framework for evaluating user and item fairness in recommender systems. *User Modeling and User-Adapted Interaction* (2021), 1–55.

[17] Yashar Deldjoo, Alejandro Bellogin, and Tommaso Di Noia. 2021. Explaining recommender systems fairness and accuracy through the lens of data characteristics. *Information Processing & Management* 58, 5 (2021), 102662.

[18] Yashar Deldjoo and Tommaso Di Noia. 2024. CFaiRLLM: Consumer Fairness Evaluation in Large-Language Model Recommender System. *arXiv preprint arXiv:2403.05668* (2024).

[19] Yashar Deldjoo, Zhankui He, Julian McAuley, Anton Korikov, Scott Sanner, Arnau Ramisa, René Vidal, Maheswaran Sathiamoorthy, Atoosa Kasirzadeh, and Silvia Milano. 2024. A Review of Modern Recommender Systems Using Generative Models (Gen-RecSys). *KDD'24* (2024).

[20] Yashar Deldjoo, Zhankui He, Julian McAuley, Anton Korikov, Scott Sanner, Arnau Ramisa, René Vidal, Maheswaran Sathiamoorthy, Atoosa Kasirzadeh, Silvia Milano, and Francesco Ricci. 2024. Recommendation with Generative Models. *arXiv* (2024).

[21] Yashar Deldjoo, Dietmar Jannach, Alejandro Bellogin, Alessandro Difonzo, and Dario Zanzonelli. 2022. A survey of research on fair recommender systems. *arXiv preprint arXiv:2205.11127* 10 (2022).

[22] Yashar Deldjoo, Dietmar Jannach, Alejandro Bellogin, Alessandro Difonzo, and Dario Zanzonelli. 2023. Fairness in recommender systems: research landscape and future directions. *User Modeling and User-Adapted Interaction* (2023), 1–50.

[23] Yashar Deldjoo, Markus Schedl, Paolo Cremonesi, and Gabriella Pasi. 2018. Content-Based Multimedia Recommendation Systems: Definition and Application Domains. In *Proceedings of the 9th Italian Information Retrieval Workshop*.

[24] Virginie Do, Sam Corbett-Davies, Jamal Atif, and Nicolas Usunier. 2021. Two-sided fairness in rankings via Lorenz dominance. *Advances in Neural Information Processing Systems* 34 (2021).

[25] Qiang Dong, Shuang-Shuang Xie, and Wen-Jun Li. 2021. User-item matching for recommendation fairness. *IEEE Access* 9 (2021), 130389–130398.

[26] Michael D Ekstrand, Robin Burke, and Fernando Diaz. 2019. Fairness and discrimination in recommendation and retrieval. In *Proceedings of the 13th ACM Conference on Recommender Systems*. 576–577.

[27] Golnoosh Farnadi, Pigi Kouki, Spencer K Thompson, Sriram Srinivasan, and Lise Getoor. 2018. A fairness-aware hybrid recommender system. *arXiv preprint arXiv:1809.09030* (2018).

[28] Pedro Ferreira, Ricardo Limongi, and Luiz Paulo Fávero. 2023. Generating music with data: application of deep learning models for symbolic music composition. *Applied Sciences* 13, 7 (2023), 4543.

[29] Yingqiang Ge, Shuchang Liu, Ruoyuan Gao, Yikun Xian, Yunqi Li, Xiangyu Zhao, Changhua Pei, Fei Sun, Junfeng Ge, Wenwu Ou, et al. 2021. Towards long-term fairness in recommendation. In *Proceedings of the 14th ACM international conference on web search and data mining*. 445–453.

[30] Shijie Geng, Shuchang Liu, Zuohui Fu, Yingqiang Ge, and Yongfeng Zhang. 2022. Recommendation as language processing (rlp): A unified pretrain, personalized prompt & predict paradigm (p5). In *Proceedings of the 16th ACM Conference on Recommender Systems*. 299–315.

[31] Elizabeth Gómez, Carlos Shui Zhang, Ludovico Boratto, Maria Salamó, and Mirko Marras. 2021. The winner takes it all: geographic imbalance and provider (un) fairness in educational recommender systems. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 1808–1812.
[32] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. In *Advances in Neural Information Processing Systems*.
[33] Qianxiu Hao, Qianqian Xu, Zhiyong Yang, and Qingming Huang. 2021. Pareto optimality for fairness-constrained collaborative filtering. In *Proceedings of the 29th ACM International Conference on Multimedia*. 5619–5627.
[34] Xiangnan He, Kuan Deng, Xiang Wang, Yan Li, Yongdong Zhang, and Meng Wang. 2020. Lightgcn: Simplifying and powering graph convolution network for recommendation. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*. 639–648.
[35] Zhankui He, Zhouhang Xie, Rahul Jha, Harald Steck, Dawen Liang, Yesu Feng, Bodhisattwa Prasad Majumder, Nathan Kallus, and Julian McAuley. 2023. Large language models as zero-shot conversational recommenders. In *Proceedings of the 32nd ACM international conference on information and knowledge management*. 720–730.
[36] Jonathan Ho, Ajay Jain, and Pieter Abbeel. 2020. Denoising diffusion probabilistic models. *Advances in neural information processing systems* 33 (2020), 6840–6851.
[37] Yupeng Hou, Shanlei Mu, Wayne Xin Zhao, Yaliang Li, Bolin Ding, and Ji-Rong Wen. 2022. Towards universal sequence representation learning for recommender systems. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 585–593.
[38] Yupeng Hou, Junjie Zhang, Zihan Lin, Hongyu Lu, Ruobing Xie, Julian McAuley, and Wayne Xin Zhao. 2023. Large language models are zero-shot rankers for recommender systems. *arXiv preprint arXiv:2305.08845* (2023).
[39] Yupeng Hou, Junjie Zhang, Zihan Lin, Hongyu Lu, Ruobing Xie, Julian McAuley, and Wayne Xin Zhao. 2024. Large language models are zero-shot rankers for recommender systems. In *European Conference on Information Retrieval*. Springer, 364–381.
[40] Meng Jiang, Keqin Bao, Jizhi Zhang, Wenjie Wang, Zhengyi Yang, Fuli Feng, and Xiangnan He. 2024. Item-side Fairness of Large Language Model-based Recommendation System. In *Proceedings of the ACM on Web Conference 2024*. 4717–4726.
[41] Ray Jiang, Sven Gowal, Yuqiu Qian, Timothy Mann, and Danilo J Rezende. 2018. Beyond Greedy Ranking: Slate Optimization via List-CVAE. In *International Conference on Learning Representations*.
[42] Diederik P Kingma and Max Welling. 2013. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114* (2013).
[43] Ömer Kırnap, Fernando Diaz, Asia Biega, Michael Ekstrand, Ben Carterette, and Emine Yilmaz. 2021. Estimation of fair ranking metrics with incomplete judgments. In *Proceedings of the Web Conference 2021*. 1065–1075.
[44] Jing Yu Koh, Daniel Fried, and Russ R Salakhutdinov. 2024. Generating images with multimodal language models. *Advances in Neural Information Processing Systems* 36 (2024).
[45] Xinyi Li, Yongfeng Zhang, and Edward C Malthouse. 2023. A Preliminary Study of ChatGPT on News Recommendation: Personalization, Provider Fairness, Fake News. *arXiv preprint arXiv:2306.10702* (2023).
[46] Yunqi Li, Hanxiong Chen, Zuohui Fu, Yingqiang Ge, and Yongfeng Zhang. 2021. User-oriented fairness in recommendation. In *Proceedings of the Web Conference 2021*. 624–632.
[47] Dawen Liang, Rahul G Krishnan, Matthew D Hoffman, and Tony Jebara. 2018. Variational autoencoders for collaborative filtering. In *Proceedings of the 2018 world wide web conference*. 689–698.
[48] Chen Lin, Xinyi Liu, Guipeng Xv, and Hui Li. 2021. Mitigating sentiment bias for recommender systems. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 31–40.
[49] Shuchang Liu, Fei Sun, Yingqiang Ge, Changhua Pei, and Yongfeng Zhang. 2021. Variation control and evaluation for generative slate recommendations. In *Proceedings of the Web Conference 2021*. 436–448.
[50] Weiwen Liu, Feng Liu, Ruiming Tang, Ben Liao, Guangyong Chen, and Pheng Ann Heng. 2020. Balancing between accuracy and fairness for interactive recommendation with reinforcement learning. In *Advances in Knowledge Discovery and Data Mining: 24th Pacific-Asia Conference, PAKDD 2020, Singapore, May 11–14, 2020, Proceedings, Part I 24*. Springer, 155–167.
[51] Jianxin Ma, Chang Zhou, Peng Cui, Hongxia Yang, and Wenwu Zhu. 2019. Learning disentangled representations for recommendation. *Advances in neural information processing systems* 32 (2019).
[52] Bonan Min, Hayley Ross, Elior Sulem, Amir Pouran Ben Veyseh, Thien Huu Nguyen, Oscar Sainz, Eneko Agirre, Ilana Heintz, and Dan Roth. 2023. Recent advances in natural language processing via large pre-trained language models: A survey. *Comput. Surveys* 56, 2 (2023), 1–40.
[53] Mohammadmehdi Naghiaei, Hossein A Rahmani, and Yashar Deldjoo. 2022. Cpfair: Personalized consumer and producer fairness re-ranking for recommender systems. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 770–779.
[54] U.S. Department of Justice. 2018. Herfindahl-Hirschman Index. https://www.justice.gov/atr/herfindahl-hirschman-index. Accessed: 25-Jun-2024.
[55] Gourab K Patro, Arpita Biswas, Niloy Ganguly, Krishna P Gummadi, and Abhijnan Chakraborty. 2020. Fairrec: Two-sided fairness for personalized recommendations in two-sided platforms. In *Proceedings of The Web Conference 2020*. 1194–1204.
[56] Hossein A Rahmani, Yashar Deldjoo, Ali Tourani, and Mohammadmehdi Naghiaei. 2022. The Unfairness of Active Users and Popularity Bias in Point-of-Interest Recommendation. In *Bias@ECIR'22*.
[57] Hossein A Rahmani, Mohammadmehdi Naghiaei, Mahdi Dehghan, and Mohammad Aliannejadi. 2022. Experiments on generalizability of user-oriented fairness in recommender systems. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 2755–2764.
[58] Hossein A. Rahmani, Mohammadmehdi Naghiaei, and Yashar Deldjoo. 2024. A Personalized Framework for Consumer and Producer Group Fairness Optimization in Recommender Systems. *ACM Transaction on Recommender Systems (TORS)* (2024).
[59] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. 2012. BPR: Bayesian personalized ranking from implicit feedback. *arXiv preprint arXiv:1205.2618* (2012).
[60] Noveen Sachdeva, Giuseppe Manco, Ettore Ritacco, and Vikram Pudi. 2019. Sequential variational autoencoders for collaborative filtering. In *International Conference on Web Search and Data Mining*.

[61] Badrul Sarwar, George Karypis, Joseph Konstan, and John Riedl. 2000. Analysis of recommendation algorithms for e-commerce. In *Proceedings of the 2nd ACM Conference on Electronic Commerce*. 158–167.
[62] Dougal Shakespeare, Lorenzo Porcaro, Emilia Gómez, and Carlos Castillo. 2020. Exploring artist gender bias in music recommendation. *arXiv preprint arXiv:2009.01715* (2020).
[63] Tianshu Shen, Jiaru Li, Mohamed Reda Bouadjenek, Zheda Mai, and Scott Sanner. 2023. Towards understanding and mitigating unintended biases in language model-driven conversational recommendation. *Information Processing & Management* 60, 1 (2023), 103139.
[64] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. 2015. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*. PMLR, 2256–2265.
[65] Tom Sühr, Sophie Hilgard, and Himabindu Lakkaraju. 2021. Does fair ranking improve minority outcomes? understanding the interplay of human and algorithmic biases in online hiring. In *Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society*. 989–999.
[66] Tu Vu, Brian Lester, Noah Constant, Rami Al-Rfou, and Daniel Cer. 2021. Spot: Better frozen model adaptation through soft prompt transfer. *arXiv preprint arXiv:2110.07904* (2021).
[67] Mengting Wan, Jianmo Ni, Rishabh Misra, and Julian McAuley. 2020. Addressing marketing bias in product recommendations. In *Proceedings of the 13th international conference on web search and data mining*. 618–626.
[68] Jiayin Wang, Weizhi Ma, Jiayu Li, Hongyu Lu, Min Zhang, Biao Li, Yiqun Liu, Peng Jiang, and Shaoping Ma. 2022. Make fairness more fair: Fair item utility estimation and exposure re-distribution. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 1868–1877.
[69] Xiang Wang, Xiangnan He, Meng Wang, Fuli Feng, and Tat-Seng Chua. 2019. Neural graph collaborative filtering. In *Proceedings of the 42nd international ACM SIGIR conference on Research and development in Information Retrieval*. 165–174.
[70] Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2022. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171* (2022).
[71] Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, et al. 2022. Emergent abilities of large language models. *arXiv preprint arXiv:2206.07682* (2022).
[72] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems* 35 (2022), 24824–24837.
[73] Leonard Weydemann, Dimitris Sacharidis, and Hannes Werthner. 2019. Defining and measuring fairness in location recommendations. In *Proceedings of the 3rd ACM SIGSPATIAL international workshop on location-based recommendations, geosocial networks and geoadvertising*. 1–8.
[74] Chuhan Wu, Fangzhao Wu, Xiting Wang, Yongfeng Huang, and Xing Xie. 2021. Fairness-aware news recommendation with decomposed adversarial learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 35. 4462–4469.
[75] Yao Wu, Jian Cao, Guandong Xu, and Yudong Tan. 2021. TFROM: A Two-sided Fairness-Aware Recommendation Model for Both Customers and Providers. *arXiv preprint arXiv:2104.09024* (2021).
[76] Yang Xiao, Qingqi Pei, Lina Yao, Shui Yu, Lei Bai, and Xianzhi Wang. 2020. An enhanced probabilistic fairness-aware group recommendation by incorporating social activeness. *Journal of Network and Computer Applications* 156 (2020), 102579.
[77] Shuyuan Xu, Wenyue Hua, and Yongfeng Zhang. 2023. OpenP5: Benchmarking Foundation Models for Recommendation. *arXiv preprint arXiv:2306.11134* (2023).
[78] Feng Yuan, Lina Yao, and Boualem Benatallah. 2020. Exploring missing interactions: A convolutional generative adversarial network for collaborative filtering. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*. 1773–1782.
[79] Chenshuang Zhang, Chaoning Zhang, Mengchun Zhang, and In So Kweon. 2023. Text-to-image diffusion models in generative ai: A survey. *arXiv preprint arXiv:2303.07909* (2023).
[80] Jizhi Zhang, Keqin Bao, Yang Zhang, Wenjie Wang, Fuli Feng, and Xiangnan He. 2023. Is chatgpt fair for recommendation? evaluating fairness in large language model recommendation. *arXiv preprint arXiv:2305.07609* (2023).
[81] Yong Zheng. 2019. Multi-stakeholder recommendations: case studies, methods and challenges. In *Proceedings of the 13th ACM Conference on Recommender Systems*. 578–579.
[82] Ziwei Zhu, Jingu Kim, Trung Nguyen, Aish Fenton, and James Caverlee. 2021. Fairness among new items in cold start recommender systems. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 767–776.

Received 18 January 2024; revised 3 July 2024; accepted 4 August 2024