Task-Aware Automated User Profile Generation for Recommendation Simulation Using Large Language Models

Xinye Wanyan
xinye.wanyan@student.rmit.edu.au
RMIT University
Melbourne, VIC, Australia

Chenglong Ma
chenglong.ma@rmit.edu.au
RMIT University
Melbourne, VIC, Australia

Danula Hettiachchi
danula.hettiachchi@rmit.edu.au
RMIT University
Melbourne, VIC, Australia

Ziqi Xu
ziqi.xu@rmit.edu.au
RMIT University
Melbourne, VIC, Australia

Jeffrey Chan
jeffrey.chan@rmit.edu.au
RMIT University
Melbourne, VIC, Australia

Abstract

Large Language Model (LLM)-based agent simulation has emerged as a promising approach to meet the increasing demand for real-time and rigorous evaluation in modern recommender systems. A typical LLM-driven simulation framework comprises three essential components: the profile module, memory module, and action module. However, existing studies have primarily concentrated on enhancing the memory and action modules, with limited attention to profile generation, which plays a pivotal role in ensuring realistic agent behaviours and aligning simulated interactions with real user dynamics. Moreover, the scarcity of datasets specifically designed for recommendation simulations has led to heavy reliance on manually crafted profiles, significantly limiting the scalability and generalisability of simulation frameworks across different datasets. To address these challenges, this work proposes an Automated Profile Generation Framework for Recommendation Simulation, APG4RecSim, that constructs realistic, coherent, and robust user profiles with minimal supervision. Extensive experiments on three benchmark datasets demonstrate that APG4RecSim achieves the best overall performance on discrimination, ranking, and rating tasks, improving ranking quality by up to 7% in nDCG@10 and reducing rating distribution divergence by 8% in JSD compared to existing profile-generation baselines. Beyond overall performance gains, our results show that profiles generated by APG4RecSim are resilient to popularity- and position-induced biases and maintain stable performance across datasets and different LLMs.[1]

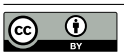
CCS Concepts

• **Information systems → Recommender systems**; • **Computing methodologies → Interactive simulation**; **Agent / discrete models**.

Keywords

Recommender System, Generative AI, User Behaviour Simulation

[1]The source code is available at https://github.com/WennyXY/APG4Sim.



This work is licensed under a Creative Commons Attribution 4.0 International License.
SIGIR '26, Melbourne, VIC, Australia
© 2026 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-2599-9/2026/07
https://doi.org/10.1145/3805712.3809707

ACM Reference Format:
Xinye Wanyan, Chenglong Ma, Danula Hettiachchi, Ziqi Xu, and Jeffrey Chan. 2026. Task-Aware Automated User Profile Generation for Recommendation Simulation Using Large Language Models. In *Proceedings of the 49th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '26), July 20–24, 2026, Melbourne, VIC, Australia.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3805712.3809707

1 Introduction

Modern recommender systems (RSs) require real-time user interactions to facilitate effective performance evaluation and continuous system refinement [12, 14, 40]. To address the limitations of static datasets and costly user studies, LLM-based agent simulation has emerged as a promising approach for generating instantaneous, realistic user feedback and supporting large-scale, concurrent evaluation environments for RSs [9, 20, 35, 41, 44]. A typical LLM-based agent consists of three fundamental components, with the LLM serving as the core reasoning engine [9, 10, 20, 31]. The **profile module** encodes the agent's identity, preferences, and behavioural traits, thereby defining how the agent tends to act [19, 29]. The **memory module** maintains the agent's interaction history, allowing it to reason with accumulated context rather than behaving independently at each time step [42]. The **action module** takes the recommendations as input and generates decisions conditioned on the profile and memory, ultimately determining how the agent interacts with the RS throughout the simulation loop.

Despite agent behaviour being governed by the synergy of three distinct modules, existing research primarily focuses on exploring the design of the memory and action modules [31, 36, 43], while largely overlooking profile generation. RecAgent [31] advances user simulation by implementing a hierarchical memory architecture designed to mirror human cognitive processes. Agent4Rec [36] proposes a hybrid memory system that intertwines objective interaction history with subjective emotional memories, allowing the agent to mimic the nuance of human decision-making. Zhang et al. [43] enhances the interaction inference of the action module by combining an LLM-based logical model with two statistical models to ensure the robustness and reliability of agent behaviours. However, current frameworks typically approach profile construction by manually engineering a set of target attributes before employing LLMs for extraction and inference. Empirical evidence demonstrates that guided profile generation significantly enhances personalisation

across diverse tasks [38], highlighting its crucial role in advancing the reliability and effectiveness of user behaviour simulation within RSs [3]. In LLM-based simulation, these findings highlight the limitations of unguided or purely heuristic profile extraction and motivate the need for structured, context-aware approaches that align profiles with downstream decision-making requirements.

The oversight of profile construction and optimisation introduces three fundamental **limitations** to current simulation frameworks: (1) **Information Bottleneck due to Rigid Schema**: By relying on manually defined attribute lists, existing methods suffer from a "closed-world" constraint. This prevents the agent from capturing latent or evolving user traits (e.g., shifting interests or complex conditional preferences) that fall outside the designer's static taxonomy. (2) **Behavioural Misalignment**: Without a rigorous, statistically grounded profiling mechanism, agents often lack a strong "instructional anchor". Poorly constructed profiles directly degrade the realism of simulated behaviours and weaken the alignment between simulated and real user interactions. (3) **Lack of Domain Generalisability**: Manual profile engineering is highly domain-dependent. A framework optimised for e-commerce attributes (e.g., price, brand) fails to generalise to content streaming (e.g., genre, mood) without significant human re-engineering, limiting the scalability of the simulation. Existing profiling methods fail to account for the situational relevance of user traits; an attribute critical for one recommendation task (e.g., price sensitivity in e-commerce) may be irrelevant noise in another (e.g., content rating in social media). Furthermore, the granularity of user traits must align with the specific metadata available in the target domain to be actionable.

Based on the identified limitations of existing studies, this work investigates the following research questions:

- **RQ1** How to automatically construct user profiles from interaction data without manually defined attribute schemas, while capturing latent and task-relevant user traits through context-aware consolidation?
- **RQ2** How do automatically generated profiles improve the behavioural alignment between simulated agents and real users across multiple recommendation tasks?
- **RQ3** Can the proposed profile generation framework generalise across domains, datasets, and LLM backbones under a unified and controlled evaluation protocol?

We answer these questions through the design of our framework and extensive experimental evaluation. In summary, the main contributions of this paper can be concluded as follows:

- We propose an automated profile generation framework that constructs diverse and coherent user profiles from interaction data without relying on manually engineered attribute schemas.
- We introduce a context-aware consolidation mechanism that aligns profile attributes with task-specific decision processes, improving behavioural realism in agent-based simulation.
- We design a systematic evaluation protocol that isolates profile quality from agent architectures and demonstrate consistent improvements compared to existing profile generation methods across ranking, discrimination, and rating tasks on multiple datasets and LLM backbones.

2 Related Work

2.1 Agent-based Simulation for RSs

Generating synthetic interaction data through agent-based simulation offers a practical solution to several challenges in RSs. By leveraging the advanced reasoning and language understanding capabilities of LLMs [27], these approaches enable agents to perform complex and realistic behaviours interacting with online systems [26, 31, 36, 37]. Ungruh et al. [28] critique the "monolith" approach which relied on monolithic behavioural assumptions that treated users as uniform decision-makers, demonstrating that incorporating diverse behavioural choice models, a "mosaic" of user patterns, is essential for high-fidelity simulation. RecAgent [31] primarily focuses on memory module and action module designing a three-level hierarchical memory module combined with recommendation-related behaviour and two types of social behaviour. Agent4Rec [36] proposes to integrate both factual and emotional memories and an emotion-driven reflection mechanism for decision-making which supports both taste-driven and emotion-driven actions in RSs. Zhang et al. [43] combine an LLM-based logical model with statistical models for binary interactions of *like* or *dislike* inference. SimUSER [4] leverages LLM-based agents to simulate user behaviour and analyse the impact of thumbnails, exposure effects, and reviews on user engagement within a standard modular architecture comprising profiling, memory, planning, and action components. Cai et al. [5] propose the Agentic Feedback Loop (AFL) framework, which models a collaborative dialogue between the system and the user to mutually enhance performance. Notably, their user agent bypasses explicit natural language profiling, instead relying on a pre-trained reward model (e.g., SASRec) to dictate preferences, effectively limiting the LLM's role to the narration of predetermined scores. DyTA4Rec [33] focuses on incorporating temporal reasoning to infer users' dynamic behaviour in recommender systems.

Existing agent-based simulation studies predominantly focus on improving agent reasoning, memory, or planning mechanisms, while user profile construction is often assumed to be manual, static, and weakly evaluated.

2.2 User Profile in RS Simulation

User profiling is a foundational component of recommender systems, traditionally represented through feature vectors, latent embeddings, or demographic attributes derived from historical interactions [2, 25, 30]. These representations are primarily designed to optimise recommendation accuracy rather than to support behavioural simulation, and they are often opaque or difficult to interpret in simulation settings. RecAgent [31] designs user profile with attributes consisting of personality, interests, and behaviour features. They mention three strategies to generate user profiles for different needs: handcraft, GPT-based, and real-data alignment methods but don't delve deep into it. Ma et al. [24] propose a personality-driven user behaviour simulator powered by LLMs integrating the Big Five personality traits to model personalised behavioural patterns in recommender systems. However, these studies have not delved deep into the profile generation. They only provide the attributes they design, and haven't explored the evaluation, reason, and optimisation. Zhang et al. [43] consider the importance

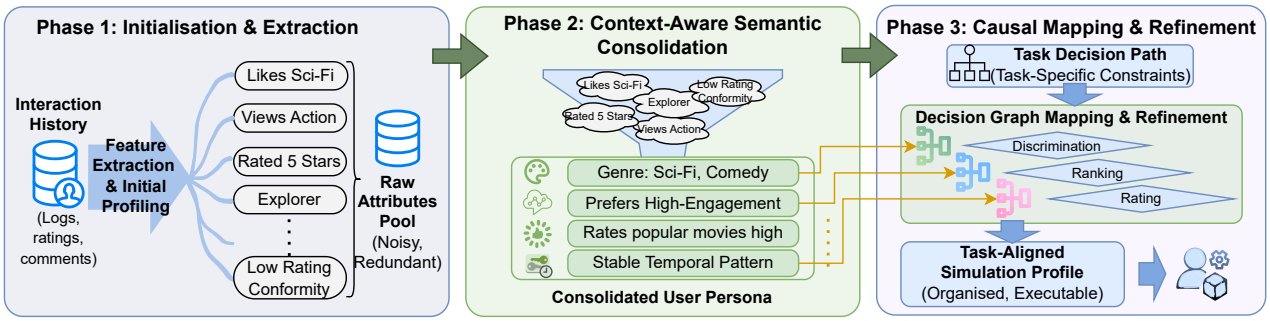


Figure 1: Overview of APG4RecSim, a training-free and context-adaptive LLM-based profile generation workflow for recommender system simulation. This diagram illustrates the proposed three-stage methodology for creating high-fidelity profiles for agents. Stage 1 (Initialisation & Extraction) involves mining raw features from a user's interaction history (e.g., logs, ratings, reviews) to create a pool of attributes which may contain noises. Stage 2 (Context-Aware Semantic Consolidation) employs a semantic merging and deduplication process to refine the raw attributes into a concise, coherent user profile based on given context. Finally, Stage 3 (Decision Path Mapping & Refinement) maps these consolidated attributes to specific nodes in a decision path, resulting in a task-aligned simulation profile with executable policies for a simulation agent.

of the agent profiling and propose to utilise LLM as the evaluator to assess the alignment between profile and interaction history. Lu et al. [23] introduce UserIP-Tuning to optimise user profiling via continuous latent vectors, effectively bypassing the token constraints and noise inherent in textual history. However, the learned soft prompts are opaque embeddings that cannot be directly utilised as natural language profiles for standard LLM agents.

Despite the central role of profiles in shaping agent behaviour, user profiles are rarely treated as a first-class research object in recommender system simulation. Even recent LLM-based profile generation approaches typically rely on predefined attribute schemas, limiting their ability to adapt across tasks and domains and constraining behavioural generalisation. In contrast, our work centres on automated and context-aware profile generation, introducing a training-free and generalisable framework that explicitly aligns user profiles with task-specific decision processes to enable more realistic user behaviour simulation.

Leveraging these insights, we introduce a multi-stage, context-sensitive framework for guided profile generation. This approach automates the construction of user profiles without requiring additional training, ensuring explicit alignment with task-specific decision logic to facilitate more consistent, realistic, and generalisable behaviour simulation.

3 Methodology

We propose a three-stage framework, APG4RecSim, to convert raw user interaction histories into *task-executable* agent profiles for recommendation simulation. Given a user history $\mathcal{H}_u$, our goal is to produce (i) a compact, domain-grounded description of stable user traits, and (ii) task-specific decision policies that operationalise these traits for a target simulation task (e.g., discrimination, ranking, or rating). Accordingly, the pipeline consists of: (1) **Attribute Initialisation and Extraction**, which elicits a high-recall set of candidate traits from $\mathcal{H}_u$; (2) **Context-Aware Semantic Consolidation**, which deduplicates and grounds traits to the dataset schema under an explicit context C; and (3) **Causal Mapping and Refinement**, which instantiates a task decision structure and maps consolidated traits to executable policies via a counterfactual relevance test.

3.1 Problem Definition

For each user u, we define the interaction history $\mathcal{H}_u = \{(i_k, r_k, t_k)\}_{k=1}^{n}$, where each tuple consists of an item i_k, the associated feedback signal r_k (e.g., rating, click, or review), and a timestamp t_k. We define a *Task-Aligned User Profile* as a tuple

$$\mathcal{P}_u(\mathcal{T}) = \langle \mathcal{I}_{meta}, \mathcal{A}_{dom}, \mathcal{G}_{policy}(\mathcal{T}) \rangle,$$

where $\mathcal{I}_{meta}$ stores immutable identifiers (e.g., user ID, optional demographic tags), $\mathcal{A}_{dom}$ is a domain-grounded set of consolidated natural-language traits (task-agnostic), and $\mathcal{G}_{policy}(\mathcal{T})$ is a set of task-specific decision policies parameterised by task specification $\mathcal{T}$. Our objective is to learn a mapping

$$f : \mathcal{H}_u \to \mathcal{P}_u(\mathcal{T}),$$

implemented as a staged decomposition:

$$\mathcal{A}_{raw} \leftarrow g_{\text{ext}}(\mathcal{H}_u); \quad \mathcal{A}_{dom} \leftarrow g_{\text{merge}}(\mathcal{A}_{raw}, C);$$
$$\mathcal{G}_{policy}(\mathcal{T}) \leftarrow g_{\text{map}}(\mathcal{A}_{dom}, \mathcal{T}).$$

3.2 Stage 1: Initialisation and Extraction

The objective of the first stage is to construct a high-recall candidate pool of user traits. To mitigate the stochastic nature of LLM generation and ensure comprehensive coverage of latent preferences, we do not rely on a single generation pass. Instead, we initialise $N = 3$ distinct preliminary profiles in parallel. This ensemble size is empirically motivated; our preliminary experiments indicate that trait recall saturates at 2–3 initialisations, making $N = 3$ sufficient to capture the user's persona without incurring unnecessary computational overhead. We prompt an LLM to explain the observed history $\mathcal{H}_u$ by generating natural-language descriptors that account for recurring interaction patterns, rather than predicting future items.

Formally, we aggregate these into a raw feature pool:

$$\mathcal{A}_{raw} = \bigcup_{j=1}^{3} \mathcal{A}^{(j)}$$

$$\mathcal{A}^{(j)} = g_{\text{ext}}(\mathcal{H}_u;\, \theta) = \text{LLM}(\mathcal{H}_u \mid \text{Prompt}_{\text{ext}}),$$

where $\mathcal{A}_{raw}$ is intentionally over-complete and may contain redundancy, paraphrases, and attributes not aligned with the dataset vocabulary. This union strategy prioritises recall to minimise missed latent preferences, accepting that the pool will contain redundancy and noise, which are subsequently resolved in the domain-aware Stage 2.

3.3 Stage 2: Context-Aware Consolidation

Raw attributes $\mathcal{A}_{raw}$ often exhibit semantic redundancy (e.g., paraphrases) and vocabulary mismatch with the target dataset. We consolidate $\mathcal{A}_{raw}$ into a compact, domain-grounded trait set $\mathcal{A}_{dom}$ by conditioning the merging process on an explicit context

$$C = \langle \mathcal{D}_{info}, \mathcal{T}_{desc}, \mathcal{E}_{shot} \rangle,$$

where $\mathcal{D}_{info}$ describes the dataset schema and admissible item metadata fields, $\mathcal{T}_{desc}$ specifies the simulation task family and the required semantic granularity, and $\mathcal{E}_{shot}$ provides a representative example instance to calibrate terminology. We then apply LLM-guided semantic consolidation:

$$\mathcal{A}_{dom} = g_{\text{merge}}(\mathcal{A}_{raw}, C) = \text{LLM}(\mathcal{A}_{raw} \mid C, \text{Prompt}_{\text{merge}}),$$

which performs (i) deduplication into canonical descriptors and (ii) grounding to dataset-native concepts, producing a task-agnostic profile ready for policy mapping.

3.4 Stage 3: Causal Mapping and Refinement

To make traits executable for simulation, we instantiate a task decision path and map each trait to the decision step(s) it directly affects.

3.4.1 Decision Path Instantiation. For a task $\mathcal{T}$ (discrimination, ranking, or rating), we represent the agent's decision structure as an ordered *decision path* $\Pi(\mathcal{T}) = [s_1, \ldots, s_m]$, where each step s_j corresponds to a cognitively meaningful judgement (e.g., hard filtering, trade-off comparison, final choice/rating). The path is obtained either from a domain-expert heuristic template or generated by an LLM conditioned on the task description and a single input-output exemplar.

3.4.2 Counterfactual Trait-to-Step Mapping. We map a trait $a \in \mathcal{A}_{dom}$ to a step s_j if altering only a (while keeping all other traits and the candidate set fixed) changes the agent's output at that step under the task interface. Operationally, we create a counterfactual profile $\mathcal{A}_{dom}^{(a \leftarrow a')}$ by replacing a with a minimally perturbed variant a' (e.g., negation or weakened preference), run the agent on the same instance, and mark a as relevant to s_j if the predicted decision differs (selection set changes for discrimination, ranking order changes for ranking, or rating changes beyond a small threshold for rating). Traits that never trigger a counterfactual change are deprioritised or merged further, whereas traits with verified counterfactual influence are preserved to avoid over-pruning. The final output is a task-aligned policy set $\mathcal{G}_{policy}(\mathcal{T})$ that attaches each retained trait to the step(s) it affects, enabling controllable and interpretable simulation.

4 Experiments

This section presents a comprehensive experimental evaluation of the proposed framework, detailing the experimental setup, baseline comparisons, and results across multiple tasks and datasets. Furthermore, we conduct rigorous ablation studies, robustness analyses, and an investigation into the impact of LLM parametric priors on simulation.

4.1 Experimental Settings

4.1.1 Datasets and Data Split. We conduct experiments on three public recommendation datasets, including MovieLens-1M (ML-1M) [15], Amazon-Book [17], and Amazon-Beauty [17]. For each dataset, interactions are ordered chronologically by timestamp. For each user, the earliest 80% of interactions are used for profile generation, while the remaining 20% are reserved for profile evaluation.

4.1.2 Baselines and Evaluation Settings. We compare our proposed approach with three established profile-generation baselines across three tasks and three benchmark datasets. (1) **Recent-Interaction Baseline**: the most recent raw interacted items are directly used as contextual input for the agent to perform task prediction. (2) **RecAgent** [31]: an LLM-based approach that extracts a fixed set of pre-defined user attributes from the interaction history, consisting of personality, interests, and behaviour features. This approach categorises users into five distinct behavioural types, including watcher, explorer, critic, chatter, or poster, and prompts the LLM to classify each user based on the predefined roles and the user's historical interactions. (3) **Agent4Rec** [36]: a hybrid framework combining statistical metrics with LLM-derived traits, we adapt this baseline to focus strictly on its generative capabilities. In our implementation, only the LLM-extracted features, specifically the user tastes and their associated rationales are retained to ensure a fair, content-centric comparison with our proposed method.

To evaluate the performance of profiles, we create three simulation scenarios for different interaction tasks, including interacted items discrimination, rating, and ranking tasks.

The **discrimination task** is to assess the agent's ability to distinguish interacted items from non-interacted ones under controlled candidate sets. For each test instance, we construct a candidate pool of size $C = 10$, containing $P = 1, 3, 5$ ground-truth positive items randomly drawn from the test set to simulate varying preference patterns. The remaining $C - P$ items, drawn from non-interacted items, are used as distractors. Given a recommendation list, the agent's task is to use the learned profile to distinguish all items that belong to the ground-truth positive set. The task evaluates whether the agent can correctly identify and prioritise the ground-truth items over non-relevant ones. We use the Overlap Ratio to measure the proportion of ground-truth items that the agent can discriminate.

The **ranking task** evaluates the agent's ability to identify the most relevant item given a user profile. Specifically, we adopt a top-1 ranking setting following previous work [18, 33, 39, 45], where a single positive item is sampled from the evaluation set and mixed with a set of negative items. Conditioned on the provided user

profile, the agent is required to rank all candidate items and assign the highest rank to the positive item. Its ranking performance is measured by the position of the held-out target item within this ranking. Standard ranking metrics, including nDCG@5, nDCG@10, and Hit Rate@3 are computed to quantify how effectively the agent identifies the user's real next choice among the candidate items.

The **rating task** evaluates the agent's ability to predict users' explicit feedback based on the given profile. For each test instance, the agent is provided 10 target items sampled from the evaluation set and is asked to generate ratings conditioned on the user profile and item information. The predicted rating is then compared against the ground-truth user rating. To comprehensively assess rating performance, we evaluate results at both the micro level and the macro level. At the micro level, we adopt Root Mean Squared Error (RMSE) to measure the deviation between the rating predicted by the agent and the corresponding ground-truth rating for the same item, thereby quantifying the accuracy of individual rating predictions. At the macro level, we assess the overall alignment between simulated and real rating behaviours by grouping items according to their rating values. For each rating group, we compute the Jensen–Shannon Divergence (JSD) [22] to quantify the discrepancy between the rating distributions produced by the agent and the ground-truth distributions. This macro-level evaluation captures not only numerical accuracy but also the consistency of global rating patterns, providing a complementary perspective to the micro-level analysis.

4.1.3 Implementation Details. In our experiments, we primarily employ gpt-4o-mini (temperature is set to 0.1) as the reasoning agent to simulate user behaviour conditioned on the generated profiles. We also conduct generalisation experiments using Llama-3.3-70B-Instruct, GPT76-5.1, and DeepSeek-V3.2 to further assess the framework's robustness across different LLM architectures. As our study focuses on evaluating the quality and effectiveness of user profiles rather than agent architecture design, the agent is not equipped with additional components (e.g., memory, action, or planning modules) beyond profile-conditioned reasoning. The LLM is prompted with the user profile and instructed to role-play the corresponding user when interacting with recommended items. To mitigate the risk of propagating stereotypical biases often observed in recommendation algorithms [7, 8, 21], we enforce a strictly behavioural profiling strategy. We deliberately exclude explicit demographic attributes (e.g., gender, age, location) from both the profile generation and behavioural inference phases, ensuring the agent's decisions are grounded solely in observed interaction patterns rather than demographic priors. To ensure consistency across experiments, we employ a fixed context window for profile generation. Specifically, all user profiles are initialised based on the most recent items from the interaction history, except where noted otherwise. All experiments are conducted over five independent runs, and results are reported as averages to ensure robustness and reduce variance.

4.2 Main Results

Table 1, 2, and 3 present the overall performance of different profile-generation methods across ranking, discrimination, and rating tasks on three benchmark datasets. In these tables, the best results are highlighted in bold, and the second-best results are underlined. Note that Semantic Merge refers to the ablated version of our framework comprising only the first two stages (Initialisation and Semantic Consolidation). APG4RecSim denotes the complete proposed framework, incorporating the Causal Mapping and Refinement stage.

We observe that performance varies across metrics and tasks, and the proposed framework achieves the best performance on 16 out of 24 evaluation metrics and ranks second on 7 out of 24, demonstrating consistently strong performance across tasks and datasets. APG4RecSim consistently delivers the strongest overall performance across tasks and datasets, achieving strong results across all tasks and datasets without sacrificing performance on any single evaluation dimension. In particular, our method maintains competitive or superior results across both micro- and macro-level evaluation metrics, demonstrating robust and well-balanced effectiveness rather than optimising for a single task or metric. This behaviour reflects the design goal of the framework, which prioritises task-aligned and generalisable profile quality over metric-specific optimisation. These results indicate that the proposed automated and context-aware profile generation framework provides a more reliable and transferable foundation for agent-based recommender system simulation than existing profile-generation baselines (**RQ1 and RQ2**).

4.2.1 Discrimination Task. Table 1 summarises the performance of different profile-generation methods on the discrimination task across three benchmark datasets. Performance is evaluated using Overlap Ratio under varying levels of difficulty, requiring the agent to identify 1, 3, or 5 ground-truth items within a candidate list of 10 items (denoted as 1:10, 3:10, and 5:10 settings). The proposed approaches, Semantic Merge and APG4RecSim, generally demonstrate superior discrimination capabilities compared to the baseline methods across most dataset settings. APG4RecSim achieves the best performance (bolded values) in 7 out of the 9 experimental scenarios. Semantic Merge frequently secures the second-best performance (underlined values), particularly in the tougher 1:10 settings across all datasets.

4.2.2 Ranking Task. Table 2 reports results of the ranking task on ML-1M, Amazon-Book, and Amazon-Beauty using nDCG@5, nDCG@10, and HR@3. Overall, APG4RecSim achieves the most consistent and robust performance across datasets, outperforming profile-free baselines and existing LLM-based profile generators. On ML-1M, it delivers the best top-ranked relevance, as reflected by nDCG@5 and HR@3. On Amazon-Book, APG4RecSim attains the highest nDCG@10 and matches the best HR@3, demonstrating superior performance at deeper ranking positions. On Amazon-Beauty, it consistently achieves the best or near-best results across all metrics, confirming effective cross-domain generalisation. While not dominating every individual metric, APG4RecSim provides the most reliable and well-balanced ranking performance overall.

4.2.3 Rating Task. Table 3 presents rating performance across ML-1M, Amazon-Book, and Amazon-Beauty, evaluated at both the micro level using RMSE and the macro level using JSD. Except for the first two profile-free settings, LLM agents equipped with our generated profiles achieve the best overall rating performance, effectively balancing micro-level prediction accuracy with macro-level

Table 1: Discrimination ability comparison of different profile-generation methods across three benchmark datasets. Performance is evaluated using Overlap Ratio under three discrimination settings, including discriminate 1,3, and 5 positive item(s) from 10 recommended items, where the agent is required to identify 1, 3, or 5 ground-truth items from a candidate list of 10 items. Best results are bolded; second-best are underlined.

Model	ML-1M			Amazon-Book			Amazon-Beauty		
	1:10	3:10	5:10	1:10	3:10	5:10	1:10	3:10	5:10
No Profile	0.1100	0.2495	0.5650	0.1500	0.3415	0.5750	0.0700	0.2490	0.4620
RecAgent	0.0900	0.2745	0.5100	0.1700	0.5930	0.7300	0.1000	0.2560	0.4360
Agent4Rec	0.1000	0.2750	0.5200	0.4250	0.5925	0.7380	0.1500	0.4665	**0.5700**
Semantic Merge	0.1900	0.3760	**0.5740**	0.4300	0.6175	0.7280	0.2900	0.4470	0.5120
APG4RecSim	**0.2000**	**0.3955**	0.5665	**0.4500**	**0.6270**	**0.7400**	**0.3300**	**0.4710**	0.5380

Table 2: Ranking performance comparison on ML-1M, Amazon-Books, and Amazon-Beauty, where agents are conditioned on user profiles generated by different profile-generation models. Performance is evaluated using nDCG@5, nDCG@10, and Hit Rate@3 (HR@3). Best results are bolded; second-best are underlined.

Model	ML-1M			Amazon-Book			Amazon-Beauty		
	nDCG@5	nDCG@10	HR@3	nDCG@5	nDCG@10	HR@3	nDCG@5	nDCG@10	HR@3
No Profile	0.3275	0.4980	0.2500	0.3505	0.5240	0.2250	0.2950	0.4320	0.2665
RecAgent	0.2790	0.5080	0.2500	**0.7710**	0.8035	0.7165	0.4645	0.5670	0.5165
Agent4Rec	0.4120	0.5605	0.3625	0.6895	0.7465	0.7700	**0.6185**	**0.7000**	0.5665
Semantic Merge	0.4020	0.5800	0.3835	0.7705	0.7950	**0.8000**	0.5610	0.6740	0.6000
APG4RecSim	**0.4300**	**0.5850**	**0.4165**	0.7385	**0.8105**	**0.8000**	0.5790	0.6790	**0.6335**

Table 3: Micro-level (RMSE) and macro-level (JSD) rating performance comparison across different profile-generation methods. We evaluate (1) an empty profile only, (2) raw recent interactions as context only, (3) profiles generated by RecAgent, (4) profiles generated by Agent4Rec, (5) APG4RecSim with semantic merging only, and (6) APG4RecSim with hybrid semantic–causal merging. The first two rows represent agents operating without generated profiles. Among the profile-based methods, the best performance is highlighted in bold, and the second-best is underlined.

	ML-1M		Book		Beauty	
	Micro↓	Macro↓	Micro↓	Macro↓	Micro↓	Macro↓
1	1.0969	0.3241	1.0145	0.2758	1.2504	0.4376
2	1.0888	0.3157	1.0166	0.2780	1.2486	0.4368
3	1.1704	0.2863	1.1181	0.2947	**1.2562**	0.4447
4	1.3068	**0.2162**	1.1498	**0.1795**	1.4290	0.4595
5	1.1603	0.2515	1.1113	0.2276	1.4475	0.4538
6	**1.1225**	0.2507	**1.1028**	0.2113	1.3967	**0.4404**

distribution consistency. While some methods attain competitive RMSE, they often suffer from higher JSD, indicating misaligned global rating behaviour. In contrast, APG4RecSim, particularly with hybrid semantic–causal merging, consistently yields the lowest JSD across domains, demonstrating superior population-level rating alignment without sacrificing item-level accuracy. The results reveal clear domain-dependent patterns, with different datasets exhibiting distinct trade-offs between micro-level accuracy and macro-level distribution alignment, underscoring the importance of adaptable profile generation rather than domain-specific heuristics.

We observe that agents conditioned on (1) an empty profile and (2) raw historical interactions achieve competitive performance on the micro-level RMSE metric, yet perform poorly on the macro-level distributional metric measured by JSD. Visualisation of the rating distributions on the Amazon-Books dataset, as shown in Figure 2a, reveals that both settings produce ratings almost exclusively within the range of 3–5. This conservative rating behaviour yields low point-wise error for individual items which benefits RMSE but fails to reproduce the true rating distribution, resulting in substantial divergence at the population level. This pattern suggests that, in the absence of structured and explicit user profiles, LLM-based agents tend to default to "safe" or neutral rating strategies that reflect general priors rather than personalised preference expression. While such behaviour is sufficient to minimise per-item prediction error, it masks the lack of calibration and diversity in simulated ratings, thereby undermining the realism of the simulation at the aggregate level. In contrast, profile-conditioned agents are better able to express both positive and negative preferences, leading to rating distributions that more closely match the ground truth despite occasionally incurring higher point-wise errors.

Overall, these results confirm that context-aware automated profiles lead to better behavioural alignment in rating simulation across

domains, validating both the effectiveness and generalisability of our approach.

4.3 Impact of Initialisation Examples

We conducted a sensitivity analysis to evaluate the impact of input history length on the model's performance in the discrimination task across diverse recommender datasets. In this setup, the sequence of past user interactions serves as the contextual prompt, enabling the LLM to infer underlying preferences. By systematically varying the number of raw historical items provided in the context window, we analyse how the depth of interaction history correlates with the model's predictive accuracy. Figure 2b illustrates the impact of input history length on discrimination accuracy across the ML-1M, Amazon-Book, and Beauty datasets. We observe that discrimination performance drops as the historical length increases. Results from the Amazon-Book dataset identify a peak corresponding to a context length of 15 items, yielding a maximal overlap ratio of 0.641. The model appears more sensitive to a distinct lack of context than to an overabundance; reducing the interaction history from 15 to 10 results in a sharp drop of 10.4% (to 0.537), whereas extending the history to 25 results in a more moderate decline of 4.7% (to 0.594). These findings highlight the fragility of direct-history prompting, where the optimal context length is highly sensitive and dataset-dependent.

Building on the insights regarding interaction history, we subsequently investigated the influence of the number of initialisation examples on profile generation task. Focusing on the Amazon-Book dataset, we systematically varied the count of provided examples to determine how the volume of few-shot demonstrations affects the fidelity and discriminative power of the generated user profiles.

Trend Analysis. The Interactions Only baseline (blue square) exhibits significant volatility. It suffers a sharp performance drop when history is short (10 items) and declines again as history extends (20–25 items), illustrating a susceptibility to both the "cold-start" problem and the "lost-in-the-middle" phenomenon, in which excessive context dilutes salient signals. In contrast, APG4RecSim (purple star) maintains a stable and superior trajectory. Notably, at the shortest history length (5 items), APG4RecSim achieves an overlap ratio of ≈ 0.62, significantly outperforming the raw interaction baseline (≈ 0.60), confirming that our generated profiles effectively distil latent traits to provide a "warm start" even with sparse data.

Stability Analysis. The box plot, as shown in Figure 2d, aggregates the performance variance across all tested history lengths. APG4RecSim demonstrates superior robustness, achieving not only the highest median performance but also the tightest inter-quartile range. Conversely, the Interactions Only and RecAgent baselines show a much wider spread, indicating that their effectiveness is highly dependent on the specific length of the input history. This confirms that APG4RecSim's structured profiling decouples simulation quality from the variability of user history length, ensuring consistent behavioural alignment.

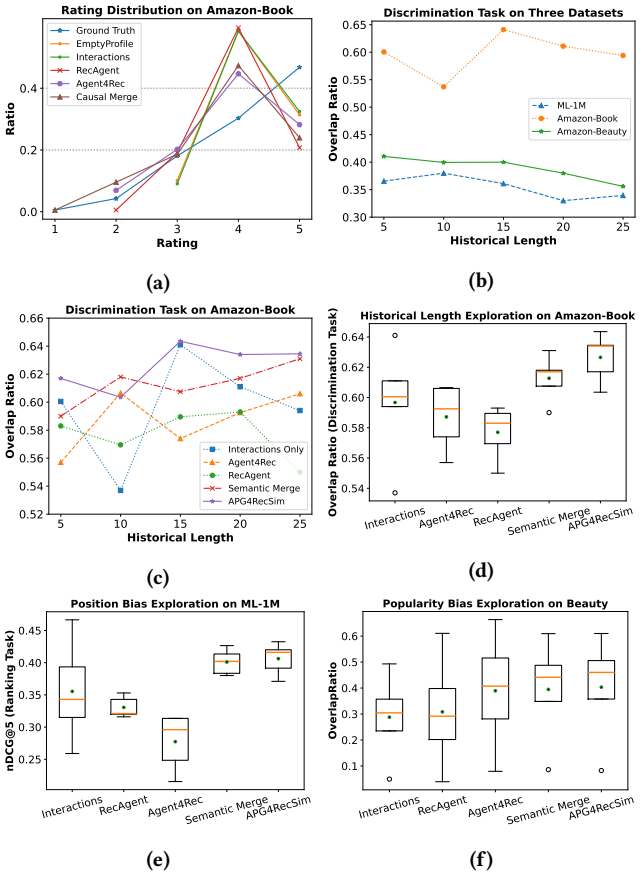


Figure 2: (a) visualisation of the rating distribution on the Amazon-Book dataset; (b) impact of interaction history length on discriminative performance across multiple datasets; (c, d) discriminative performance of generated profiles on the Amazon-Book dataset across varying interaction history lengths; (e) evaluation of robustness to position bias on the ML-1M dataset for a ranking task; (f) evaluation of robustness to popularity bias on the Amazon-Beauty dataset for a discrimination task (3 from 10).

4.4 Robustness and Generalisation Across LLMs

To assess the generalisability of APG4SimRec, we benchmark the framework on the ML-1M and Amazon-Book datasets across diverse LLM architectures. We instantiate the framework using three distinct backbones, including Llama-3.3-70B-Instruct, GPT-5.1, and DeepSeek-V3.2, to assess profile performance across ranking and discrimination tasks. Following our main experimental setup, we evaluate discrimination performance using a P-out-of-C protocol ($P \in \{3, 5\}, C = 10$), employing Overlap Ratio to measure the agent's ability to identify multiple ground-truth items within the constrained candidate set. Ranking quality is assessed under a leave-one-out setting using nDCG@10 and Hit Rate@3 (HR@3) to quantify the recovery of held-out user preferences.

Comparison results are shown in Table 4. Our framework demonstrates consistency across LLM architectures on different datasets.

Table 4: Generalisation results of the proposed automated profile generation framework on ML-1M and Amazon-Book across different LLMs. Discrimination ability is assessed using Overlap Ratio, and ranking quality is evaluated with nDCG@10 and Hit Rate@3. Best results are bolded.

LLM	Model	ML-1M				Book			
		Discrimination		Ranking		Discrimination		Ranking	
		3:10	5:10	nDCG@10	HR@3	3:10	5:10	nDCG@10	HR@3
Llama3.3-70B-Instruct	RecAgent	0.2930	0.5000	0.5365	0.3000	0.5935	0.7460	0.7160	0.6835
	Agent4Rec	0.3225	0.5440	0.5690	0.3500	0.5665	0.7500	0.7365	0.7165
	Semantic Merge	**0.3665**	0.5540	**0.5725**	0.3335	0.6170	**0.7800**	0.7480	0.7000
	APG4SimRec	0.3525	**0.5760**	0.5585	**0.3510**	**0.6435**	0.7500	**0.7505**	**0.7500**
gpt-5.1-2025-11-13	RecAgent	0.3655	**0.5440**	**0.4980**	0.2835	0.6035	**0.7680**	0.7455	0.7170
	Agent4Rec	0.3155	0.4880	0.4750	0.2500	0.6010	0.7600	0.7510	0.7000
	Semantic Merge	0.3365	0.4940	**0.4980**	0.2835	0.6005	0.7200	**0.8190**	**0.7500**
	APG4SimRec	**0.3665**	0.5410	0.4870	**0.3170**	**0.6275**	0.7540	0.8080	**0.7500**
DeepSeek-V3.2	RecAgent	0.3035	**0.5480**	0.5110	0.2165	0.5695	**0.7660**	0.7120	0.5665
	Agent4Rec	0.3265	0.5280	0.5235	0.2665	**0.6315**	0.7540	0.7200	0.6500
	Semantic Merge	0.3330	0.5240	0.5090	0.3000	0.6210	0.7580	0.7425	0.6165
	APG4SimRec	**0.3920**	0.5380	**0.5415**	**0.3165**	0.6305	0.7440	**0.7775**	**0.7335**

APG4RecSim achieves the best performance in 14 out of 24 total evaluation settings (spanning 3 LLMs, 2 datasets, and 4 metrics) and secures the second-best result in 7 additional cases. This robust performance demonstrates that our task-aware causal profiling strategy generalises effectively, operating independently of the specific capabilities of the underlying LLM backbone. For instance, APG4SimRec achieves the highest HR@3 scores on all settings. This suggests that the proposed causal mapping strategy effectively grounds the agent's decision-making process, mitigating the dependency on the idiosyncratic reasoning patterns of any single LLM. Furthermore, the performance gains are most pronounced with open-weight models (Llama-3.3-70B-Instruct and DeepSeek-V3.2), suggesting that our structured profiling effectively augments the capabilities of accessible models, bridging the gap between efficient architectures and proprietary giants.

This experimental design allows us to systematically compare different LLM backbones and examine the robustness and transferability of the proposed profile generation framework across datasets and model choices. (**RQ3**)

4.5 Robustness against Position Bias

Position bias describes the tendency of users to disproportionately interact with top-ranked items regardless of relevance regardless of their actual relevance or quality [8, 11, 16]. In LLM-agent simulations, this input-order sensitivity (or primacy effect) influences the preference evaluation, as high performance metrics may reflect mere position compliance rather than genuine interest [33].

Figure 2e evaluates the ranking performance (nDCG@5) across agents with different profiles generation methods on ML-1M dataset. The results indicate a significant performance gap between our proposed approach and the baselines. The profiles generated by APG4RecSim achieve the highest overall performance, with a median nDCG@5 of approximately 0.415 and the highest mean score (indicated by the green dot). The Semantic Merge profiles follow closely as the second-best, with a median near 0.40. Both proposed methods substantially outperform all three baselines. Furthermore, the box plot distributions show that while interactions only and Agent4Rec profile as the context exhibit high variance (large interquartile ranges and long whiskers), indicating inconsistent performance depending on the item's position, our APG4RecSim and semantic merging profiles maintain relatively compact distributions, suggesting stable decision-making regardless of item placement.

The inferior performance and high variance of the baselines indicate significant susceptibility to position bias, suggesting that agents relying on unstructured profiles or raw history struggle to decouple intrinsic item relevance from presentation order. Conversely, the stability of the agents with APG4RecSim profiles validates our structured methodology. By consolidating attributes and mapping them to explicit decision path nodes, these profiling methods ensure items are evaluated on semantic alignment rather than superficial ordering cues. The leading performance of the APG4RecSim method further indicates that explicit causal modelling within the user profile provides an effective mechanism for isolating true preferences from positional noise.

4.6 Robustness against Popularity Bias

Popularity bias is a pervasive issue in recommender systems, where algorithms disproportionately expose a small set of popular items while neglecting the majority of niche items in the "long tail" [1, 34]. In the context of user behaviour simulation, popularity bias manifests as the simulated agent systematically preferring high-exposure items over personally relevant ones, regardless of the specific user profile being simulated. Figure 2f illustrates the performance of various profile generation methodologies on the Amazon-Beauty dataset. The y-axis represents the Overlap Ratio, where a higher ratio generally indicates that the method is less affected by popularity bias, as it maintains consistency across different sampling strategies. The x-axis displays different methodologies for profile

Table 5: Discrimination performance on ML-1M using isolated item attributes to probe parametric priors. All settings utilise an empty profile, providing the agent only with specific item metadata (Title, Genre, Rating, Popularity). We compare two negative sampling strategies: Random (uniform sampling) and De-bias (popularity–rating matched sampling) to neutralise global popularity and quality biases.

Sampling	Title	Genre	Rating	Pop.	3:10(r)	3:6(r)
Random	Yes	No	No	No	0.2930	0.5390
	No	Yes	No	No	0.2365	0.4955
	No	No	Yes	No	0.3495	0.6340
	No	No	No	Yes	**0.4900**	**0.7400**
	Yes	Yes	No	No	0.2595	0.5335
	No	No	Yes	Yes	0.4225	0.6340
	Yes	Yes	Yes	Yes	0.3105	0.6340
De-bias	Yes	No	No	No	0.2710	0.5115
	No	Yes	No	No	0.2150	**0.5560**
	No	No	Yes	No	0.2490	0.5155
	No	No	No	Yes	**0.3990**	0.5395
	Yes	Yes	No	No	0.2605	0.5385
	No	No	Yes	Yes	0.2490	0.5320
	Yes	Yes	Yes	Yes	0.2660	0.4925

generation. The box plots show the robustness of each method's Overlap Ratio when subjected to different negative item sampling strategies based on popularity.

The proposed methods, Semantic Merge and APG4RecSim, generally exhibit higher median Overlap Ratios compared to the baseline methods. RecAgent profile and providing interactions only as the context have the lowest median Overlap Ratios, indicating that their performance is more significantly impacted by popularity bias. They also show a wider range of performance, suggesting lower robustness across different sampling strategies. Agent4Rec has a higher median than the other two baselines but exhibits a very large inter-quartile range, showing significant variability in its performance depending on the sampling strategy.

The results demonstrate that the proposed profile generation methods, Semantic Merge and APG4RecSim, are more robust to popularity bias in a discrimination task on the Amazon-Beauty dataset compared to baseline methods. They achieve higher median Overlap Ratios and demonstrate greater stability across different negative item sampling strategies.

4.7 Impact of Parametric Knowledge and Global Priors

LLMs can retain memory of widely used benchmark datasets like ML-1M, implicitly encoding item metadata and popularity distributions within their parametric knowledge [6, 13, 32]. This raises a critical validity concern for simulation: observed agent performance may stem from dataset-specific priors or global popularity cues, rather than genuine reasoning grounded in the provided user profile. To investigate the extent to which these internal priors drive discrimination performance, we conduct a controlled ablation study using two negative sampling strategies: **(1) Random Sampling:** Negative items are drawn uniformly at random from the uninteracted set. **(2) De-bias Sampling:** To neutralise popularity and quality biases, negative items are selected to statistically match the average popularity and rating of the ground-truth positive items. We restrict the agent's input to isolated item attributes (e.g., Title only, Popularity only) to isolate the impact of specific cues. This design effectively probes the agent's reliance on global statistics versus semantic content.

Table 5 reports the results on ML-1M. Under random sampling, providing Popularity alone yields dominant performance, consistently outperforming semantic attributes such as title and genre in both 3:10 and 3:6 settings. Providing rating information also leads to competitive results, whereas combining multiple attributes does not reliably improve performance beyond the strongest single cue, indicating diminishing returns once salient global statistics are available. These patterns suggest that, in the absence of explicit user profiles, the LLM-based agent tends to rely on coarse item-level statistics rather than integrating complementary semantic signals. However, under de-bias sampling, performance collapses across all configurations and the popularity advantage vanishes. No single attribute setting consistently dominates. This indicates that a considerable portion of the performance observed under random sampling can be attributed to differences in global item statistics, rather than fine-grained semantic reasoning or personalised signals. This contrast confirms that standard LLM agents rely heavily on spurious global statistics. While this analysis provides only indirect evidence regarding the role of LLM parametric memory, the results highlight the sensitivity of discrimination performance on ML-1M to item-level priors.

Overall, these findings motivate the need for controlled profile evaluation and support the role of APG4RecSim in encouraging task-aligned, user-specific reasoning that moves beyond statistics-driven heuristics.

5 Limitation and Future Work

We acknowledge three limitations that outline critical directions for future research.

First, as observed in the diagnostic analysis on ML-1M, LLM parametric memory and global item-level cues (e.g., popularity or rating statistics) may partially influence agent behaviour, which could confound the assessment of genuine profile-driven reasoning. Future work should incorporate stronger controls, such as item anonymisation, synthetic datasets, or less publicly exposed domains, to further disentangle parametric priors from profile-conditioned behaviour.

Second, our framework focuses on profile generation and consolidation, intentionally isolating profile quality from agent architecture. While this design choice enables controlled evaluation, it does not explore the interaction between profiles and more sophisticated agent components, such as long-term memory, planning modules, or multi-step decision policies. Extending APG4RecSim to end-to-end agent systems remains an important direction for future work.

Third, the causal mapping and refinement step is derived from LLM-based counterfactual reasoning, which remains susceptible

to inherent model stochasticity. Future work could explore hybrid approaches that integrate lightweight statistical or learning-based causal estimators to further stabilise profile refinement.

6 Conclusion

In this work, we proposed APG4RecSim, a training-free framework that automates profile generation by extracting task-aware latent traits without manual schemas (RQ1). Across multiple tasks and domains, our method achieves superior behavioural alignment, effectively balancing micro-level precision with macro-level fidelity (RQ2). Crucially, our controlled analysis confirms the framework's robustness, demonstrating that structured, task-aligned profiling mitigates the agent's reliance on global popularity priors and position bias (RQ3).

Overall, this study provides a promising direction for automating user profile generation from existing recommendation datasets, and lays the groundwork for more generalised, interpretable, and reliable LLM agent-based user behaviour simulation for recommender systems.

Acknowledgments

This research was conducted by the ARC Centre of Excellence for Automated Decision-Making and Society (ADM+S, CE200100005) and funded by the Australian Government through the Australian Research Council. This research was undertaken with the assistance of computing resources from RACE (RMIT Advanced Cloud Ecosystem).

References

[1] Himan Abdollahpouri, Masoud Mansoury, Robin Burke, Bamshad Mobasher, and Edward Malthouse. 2021. User-centered Evaluation of Popularity Bias in Recommender Systems. In *Proceedings of the 29th ACM Conference on User Modeling, Adaptation and Personalization* (Utrecht, Netherlands) *(UMAP '21)*. Association for Computing Machinery, New York, NY, USA, 119–129. doi:10.1145/3450613.3456821

[2] Mohammad Yahya H. Al-Shamri. 2016. User profiling approaches for demographic recommender systems. *Know.-Based Syst.* 100, C (May 2016), 175–187. doi:10.1016/j.knosys.2016.03.006

[3] Ananya Bhattacharjee, Yuchen Zeng, Sarah Yi Xu, Dana Kulzhabayeva, Minyi Ma, Rachel Kornfield, Syed Ishtiaque Ahmed, Alex Mariakakis, Mary P Czerwinski, Anastasia Kuzminykh, Michael Liut, and Joseph Jay Williams. 2024. Understanding the Role of Large Language Models in Personalizing and Scaffolding Strategies to Combat Academic Procrastination. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '24)*. Association for Computing Machinery, New York, NY, USA, Article 15, 18 pages. doi:10.1145/3613904.3642081

[4] Nicolas Bougie and Narimawa Watanabe. 2025. SimUSER: Simulating User Behavior with Large Language Models for Recommender System Evaluation. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 6: Industry Track)*, Georg Rehm and Yunyao Li (Eds.). Association for Computational Linguistics, Vienna, Austria, 43–60. doi:10.18653/v1/2025.acl-industry.5

[5] Shihao Cai, Jizhi Zhang, Keqin Bao, Chongming Gao, Qifan Wang, Fuli Feng, and Xiangnan He. 2025. Agentic Feedback Loop Modeling Improves Recommendation and User Simulation. In *SIGIR*. 2235–2244. https://doi.org/10.1145/3726302.3729893

[6] Nicholas Carlini, Daphne Ippolito, Matthew Jagielski, Katherine Lee, Florian Tramèr, and Chiyuan Zhang. 2023. Quantifying Memorization Across Neural Language Models. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net. https://openreview.net/forum?id=TatRHT_1cK

[7] Abhijnan Chakraborty, Johnnatan Messias, Fabricio Benevenuto, Saptarshi Ghosh, Niloy Ganguly, and Krishna Gummadi. 2017. Who makes trends? understanding demographic biases in crowdsourced recommendations. In *Proceedings of the International AAAI Conference on Web and Social Media*, Vol. 11. 22–31.

[8] Jiawei Chen, Hande Dong, Xiang Wang, Fuli Feng, Meng Wang, and Xiangnan He. 2023. Bias and Debias in Recommender System: A Survey and Future Directions. *ACM Trans. Inf. Syst.* 41, 3, Article 67 (Feb. 2023), 39 pages. doi:10.1145/3564284

[9] Luyu Chen, Quanyu Dai, Zeyu Zhang, Xueyang Feng, Mingyu Zhang, Pengcheng Tang, Xu Chen, Yue Zhu, and Zhenhua Dong. 2025. RecUserSim: A Realistic and Diverse User Simulator for Evaluating Conversational Recommender Systems. In *Companion Proceedings of the ACM on Web Conference 2025* (Sydney NSW, Australia) *(WWW '25)*. Association for Computing Machinery, New York, NY, USA, 133–142. doi:10.1145/3701716.3715258

[10] Xinshi Chen, Shuang Li, Hui Li, Shaohua Jiang, Yuan Qi, and Le Song. 2019. Generative adversarial user model for reinforcement learning based recommendation system. In *International conference on machine learning*. PMLR, 1052–1061.

[11] Andrew Collins, Dominika Tkaczyk, Akiko Aizawa, and Joeran Beel. 2018. Position bias in recommender systems for digital libraries. In *International Conference on Information*. Springer, 335–344.

[12] 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). In *Proceedings of the 30th ACM SIGKDD conference on Knowledge Discovery and Data Mining*. 6448–6458.

[13] Dario Di Palma, Felice Antonio Merra, Maurizio Sfilio, Vito Walter Anelli, Fedelucio Narducci, and Tommaso Di Noia. 2025. Do LLMs Memorize Recommendation Datasets? A Preliminary Study on MovieLens-1M. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Padua, Italy) *(SIGIR '25)*. Association for Computing Machinery, New York, NY, USA, 2582–2586. doi:10.1145/3726302.3730178

[14] Chongming Gao, Shijun Li, Wenqiang Lei, Jiawei Chen, Biao Li, Peng Jiang, Xiangnan He, Jiaxin Mao, and Tat-Seng Chua. 2022. KuaiRec: A fully-observed dataset and insights for evaluating recommender systems. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*. 540–550.

[15] F. Maxwell Harper and Joseph A. Konstan. 2015. The MovieLens Datasets: History and Context. *ACM Trans. Interact. Intell. Syst.* 5, 4, Article 19 (Dec. 2015), 19 pages. doi:10.1145/2827872

[16] Katja Hofmann, Anne Schuth, Alejandro Bellogin, and Maarten De Rijke. 2014. Effects of position bias on click-based recommender evaluation. In *European Conference on Information Retrieval*. Springer, 624–630.

[17] Yupeng Hou, Jiacheng Li, Zhankui He, An Yan, Xiusi Chen, and Julian McAuley. 2024. Bridging language and items for retrieval and recommendation. *arXiv preprint arXiv:2403.03952* (2024).

[18] 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* (Washington DC, USA) *(KDD '22)*. Association for Computing Machinery, New York, NY, USA, 585–593. doi:10.1145/3534678.3539381

[19] Tiancheng Hu and Nigel Collier. 2024. Quantifying the persona effect in llm simulations. *arXiv preprint arXiv:2402.10811* (2024).

[20] Song Jin, Juntian Zhang, Yuhan Liu, Xun Zhang, Yufei Zhang, Guojun Yin, Fei Jiang, Wei Lin, and Rui Yan. 2025. Beyond Static Testbeds: An Interaction-Centric Agent Simulation Platform for Dynamic Recommender Systems. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (Eds.). Association for Computational Linguistics, Suzhou, China, 18903–18920. doi:10.18653/v1/2025.emnlp-main.956

[21] Thorsten Krause, Lorena Göritz, and Robin Gratz. 2025. The Effect of Gender De-biased Recommendations — A User Study on Gender-specific Preferences. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems (CHI '25)*. Association for Computing Machinery, New York, NY, USA, Article 1000, 16 pages. doi:10.1145/3706598.3713155

[22] J. Lin. 1991. Divergence measures based on the Shannon entropy. *IEEE Transactions on Information Theory* 37, 1 (1991), 145–151. doi:10.1109/18.61115

[23] Yusheng Lu, Zhaocheng Du, Xiangyang Li, Pengyue Jia, Yejing Wang, Weiwen Liu, Yichao Wang, Huifeng Guo, Ruiming Tang, Zhenhua Dong, Yongrui Duan, and Xiangyu Zhao. 2025. Prompt Tuning as User Inherent Profile Inference Machine. In *Proceedings of the 34th ACM International Conference on Information and Knowledge Management* (Seoul, Republic of Korea) *(CIKM '25)*. Association for Computing Machinery, New York, NY, USA, 5898–5906. doi:10.1145/3746252.3761574

[24] Chenglong Ma, Ziqi Xu, Yongli Ren, Danula Hettiachchi, and Jeffrey Chan. 2025. PUB: An LLM-Enhanced Personality-Driven User Behaviour Simulator for Recommender System Evaluation. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR*. 2690–2694.

[25] Stuart E. Middleton, Nigel R. Shadbolt, and David C. De Roure. 2004. Ontological user profiling in recommender systems. *ACM Trans. Inf. Syst.* 22, 1 (Jan. 2004), 54–88. doi:10.1145/963770.963773

[26] Joon Sung Park, Joseph O'Brien, Carrie Jun Cai, Meredith Ringel Morris, Percy Liang, and Michael S. Bernstein. 2023. Generative Agents: Interactive Simulacra of Human Behavior. In *Proceedings of the 36th Annual ACM Symposium on User*

Interface Software and Technology (San Francisco, CA, USA) *(UIST '23)*. Association for Computing Machinery, New York, NY, USA, Article 2, 22 pages. doi:10.1145/3586183.3606763

[27] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog* 1, 8 (2019), 9.

[28] Robin Ungruh, Alejandro Bellogín, and Maria Soledad Pera. 2025. From Monolith to Mosaic: Uncovering Behavioral Differences for Choice Models in Recommender Systems Simulations. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Padua, Italy) *(SIGIR '25)*. Association for Computing Machinery, New York, NY, USA, 2717–2722. doi:10.1145/3726302.3730199

[29] Hanpeng Wang and Zijiang Yang. 2025. A Multi-Agent Approach to Investor Profiling Using Large Language Models. In *2025 International Conference on Control, Automation and Diagnosis (ICCAD)*. 1–6. doi:10.1109/ICCAD64771.2025.11099326

[30] Lu Wang, Di Zhang, Fangkai Yang, Pu Zhao, Jianfeng Liu, Yuefeng Zhan, Hao Sun, Qingwei Lin, Weiwei Deng, Dongmei Zhang, Feng Sun, and Qi Zhang. 2025. LettinGo: Explore User Profile Generation for Recommendation System. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2* (Toronto ON, Canada) *(KDD '25)*. Association for Computing Machinery, New York, NY, USA, 2985–2995. doi:10.1145/3711896.3737024

[31] Lei Wang, Jingsen Zhang, Hao Yang, Zhi-Yuan Chen, Jiakai Tang, Zeyu Zhang, Xu Chen, Yankai Lin, Hao Sun, Ruihua Song, et al. 2025. User behavior simulation with large language model-based agents. *ACM Transactions on Information Systems* 43, 2 (2025), 1–37.

[32] Xinyi Wang, Antonis Antoniades, Yanai Elazar, Alfonso Amayuelas, Alon Albalak, Kexun Zhang, and William Yang Wang. 2025. Generalization v.s. Memorization: Tracing Language Models' Capabilities Back to Pretraining Data. In *The Thirteenth International Conference on Learning Representations*. https://openreview.net/forum?id=IQxBDLmVpT

[33] Xinye Wanyan, Danula Hettiachchi, Chenglong Ma, Ziqi Xu, and Jeffrey Chan. 2025. Temporal-Aware User Behaviour Simulation with Large Language Models for Recommender Systems. In *Proceedings of the 34th ACM International Conference on Information and Knowledge Management* (Seoul, Republic of Korea) *(CIKM '25)*. Association for Computing Machinery, New York, NY, USA, 5335–5339. doi:10.1145/3746252.3760878

[34] Tianxin Wei, Fuli Feng, Jiawei Chen, Ziwei Wu, Jinfeng Yi, and Xiangnan He. 2021. Model-Agnostic Counterfactual Reasoning for Eliminating Popularity Bias in Recommender System. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining* (Virtual Event, Singapore) *(KDD '21)*. Association for Computing Machinery, New York, NY, USA, 1791–1800. doi:10.1145/3447548.3467289

[35] Ziqi Xu, Chenglong Ma, Yongli Ren, Jeffrey Chan, Wei Shao, and Feng Xia. 2025. Towards Better Evaluation of Recommendation Algorithms with Bi-directional Item Response Theory. In *Companion Proceedings of the ACM on Web Conference 2025* (Sydney NSW, Australia) *(WWW '25)*. Association for Computing Machinery, New York, NY, USA, 1455–1459. doi:10.1145/3701716.3715540

[36] An Zhang, Yuxin Chen, Leheng Sheng, Xiang Wang, and Tat-Seng Chua. 2024. On Generative Agents in Recommendation. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Washington DC, USA) *(SIGIR '24)*. Association for Computing Machinery, New York, NY, USA, 1807–1817. doi:10.1145/3626772.3657844

[37] Erhan Zhang, Xingzhu Wang, Peiyuan Gong, Zixuan Yang, and Jiaxin Mao. 2025. Exploring Human-Like Thinking in Search Simulations with Large Language Models. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Padua, Italy) *(SIGIR '25)*. Association for Computing Machinery, New York, NY, USA, 2669–2673. doi:10.1145/3726302.3730193

[38] Jiarui Zhang. 2024. Guided Profile Generation Improves Personalization with Large Language Models. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (Eds.). Association for Computational Linguistics, Miami, Florida, USA, 4005–4016. doi:10.18653/v1/2024.findings-emnlp.231

[39] Junjie Zhang, Yupeng Hou, Ruobing Xie, Wenqi Sun, Julian J. McAuley, Wayne Xin Zhao, Leyu Lin, and Ji-Rong Wen. 2024. AgentCF: Collaborative Learning with Autonomous Language Agents for Recommender Systems. In *Proceedings of the ACM on Web Conference 2024, WWW*. 3679–3689.

[40] Shuo Zhang and Krisztian Balog. 2020. Evaluating conversational recommender systems via user simulation. In *Proceedings of the 26th acm sigkdd international conference on knowledge discovery & data mining*. 1512–1520.

[41] Yu Zhang, Shutong Qiao, Jiaqi Zhang, Tzu-Heng Lin, Chen Gao, and Yong Li. 2025. A survey of large language model empowered agents for recommendation and search: Towards next-generation information retrieval. *arXiv preprint arXiv:2503.05659* (2025).

[42] Zeyu Zhang, Quanyu Dai, Xiaohe Bo, Chen Ma, Rui Li, Xu Chen, Jieming Zhu, Zhenhua Dong, and Ji-Rong Wen. 2025. A Survey on the Memory Mechanism of Large Language Model-based Agents. *ACM Trans. Inf. Syst.* 43, 6, Article 155 (Sept. 2025), 47 pages. doi:10.1145/3748302

[43] Zijian Zhang, Shuchang Liu, Ziru Liu, Rui Zhong, Qingpeng Cai, Xiangyu Zhao, Chunxu Zhang, Qidong Liu, and Peng Jiang. 2025. LLM-powered user simulator for recommender system. In *Proceedings of the Thirty-Ninth AAAI Conference on Artificial Intelligence and Thirty-Seventh Conference on Innovative Applications of Artificial Intelligence and Fifteenth Symposium on Educational Advances in Artificial Intelligence (AAAI'25/IAAI'25/EAAI'25)*. AAAI Press, Article 1483, 9 pages. doi:10.1609/aaai.v39i12.33456

[44] Zihuai Zhao, Wenqi Fan, Jiatong Li, Yunqing Liu, Xiaowei Mei, Yiqi Wang, Zhen Wen, Fei Wang, Xiangyu Zhao, Jiliang Tang, et al. 2024. Recommender systems in the era of large language models (llms). *IEEE Transactions on Knowledge and Data Engineering* 36, 11 (2024), 6889–6907.

[45] Kun Zhou, Hui Wang, Wayne Xin Zhao, Yutao Zhu, Sirui Wang, Fuzheng Zhang, Zhongyuan Wang, and Ji-Rong Wen. 2020. S3-Rec: Self-Supervised Learning for Sequential Recommendation with Mutual Information Maximization. In *CIKM '20: The 29th ACM International Conference on Information and Knowledge Management*. 1893–1902.