A Survey on LLM-powered Agents for Recommender Systems

Qiyao Peng[1] , **Hongtao Liu**[2] , **Hua Huang**[1] , **Qing Yang**[2] , and **Minglai Shao**[1]

[1]School of New Media and Communication, Tianjin University, Tianjin, China
[2]Du Xiaoman Financial Technology, Beijing, China
{qypeng, huanghua18, shaoml}@tju.edu.cn, {liuhongtao01,yangqing}@duxiaoman.com

Abstract

Recommender systems are essential components of many online platforms, yet traditional approaches still struggle with understanding complex user preferences and providing explainable recommendations. The emergence of Large Language Model (LLM)-powered agents offers a promising approach by enabling natural language interactions and interpretable reasoning, potentially transforming research in recommender systems. This survey provides a systematic review of the emerging applications of LLM-powered agents in recommender systems. We identify and analyze three key paradigms in current research: (1) Recommender-oriented approaches, which leverage intelligent agents to enhance the fundamental recommendation mechanisms; (2) Interaction-oriented approaches, which facilitate dynamic user engagement through natural dialogue and interpretable suggestions; and (3) Simulation-oriented approaches, which employ multi-agent frameworks to model complex user-item interactions and system dynamics. Beyond paradigm categorization, we analyze the architectural foundations of LLM-powered recommendation agents, examining their essential components: profile construction, memory management, strategic planning, and action execution. Our investigation extends to a comprehensive analysis of benchmark datasets and evaluation frameworks in this domain. This systematic examination not only illuminates the current state of LLM-powered agent recommender systems but also charts critical challenges and promising research directions in this transformative field.

1 Introduction

In the era of information explosion, recommender systems [Wu *et al.*, 2022] have become an indispensable component of digital platforms, helping users navigate through massive amounts of content across e-commerce, social media, and entertainment domains. While traditional recommendation approaches [He *et al.*, 2017] have achieved considerable success in providing personalized suggestions through analyzing user preferences and historical behaviors, they still face significant challenges in real-world applications, such as limited understanding of complex user intents, insufficient interaction capabilities, and the inability to provide interpretable recommendations [Zhu *et al.*, 2024b].

Recent advancements in Large Language Models (LLMs) [Achiam *et al.*, 2023] have sparked increasing interest in leveraging LLM-powered agents [Wang *et al.*, 2024a] to address the aforementioned challenges in recommender systems. The integration of LLM-powered agents into recommender systems offers several compelling advantages over traditional approaches [Zhu *et al.*, 2024b]. First, LLM agents can understand complex user preferences and generate contextual recommendations through their sophisticated reasoning capabilities, enabling more nuanced decision-making beyond simple feature-based matching. Second, their natural language interaction abilities facilitate multi-turn conversations that proactively explore user interests and provide interpretable explanations, enhancing both recommendation accuracy and user experience. Third, these agents revolutionize user behavior simulation by generating more realistic user profiles that incorporate emotional states and temporal dynamics, enabling more effective system evaluation. Furthermore, the pre-trained knowledge and strong generalization capabilities of LLMs facilitate better knowledge transfer across domains, addressing persistent challenges such as cold-start [Shu *et al.*, 2024] with minimal additional training.

In this survey, we present a comprehensive review of LLM-powered agents for recommender systems. First, we introduce the background of traditional recommender systems and discuss their limitations in understanding complex user intents, interaction capabilities, and interpretability. We then systematically examine how LLM-powered agents address these challenges through three main paradigms: recommender-oriented (e.g., [Wang *et al.*, 2024b; Wang *et al.*, 2024c]), interaction-oriented (e.g., [Zeng *et al.*, 2024; Friedman *et al.*, 2023]), and simulation-oriented (e.g., [Yoon *et al.*, 2024; Guo *et al.*, 2024]) approaches. Following that, we propose a unified agent architecture consisting of four core modules (Profile [Cai *et al.*, 2024; Zhang *et al.*, 2024c], Memory [Shi *et al.*, 2024; Fang *et al.*, 2024], Planning [Wang *et al.*, 2023b; Shi *et al.*, 2024], and Action [Zhu *et al.*, 2024a; Zhao *et al.*, 2024]) and analyze how existing methods implement these

components. Furthermore, we compile comprehensive comparisons of datasets (including Amazon series, MovieLens, Steam, etc.) and evaluation methodologies, encompassing both standard recommendation metrics and novel evaluation approaches. Finally, we explore several promising future directions in this field.

- We propose a systematic categorization of LLM-powered recommender agents, identifying three fundamental paradigms: recommender-oriented, interaction-oriented, and simulation-oriented approaches. This taxonomy provides a structured framework for understanding current research.

- We utilize a unified architectural framework for analyzing LLM-powered agent recommender, decomposing them into four essential modules: Profile Construction, Memory Management, Strategic Planning, and Action Execution. Through this framework, we systematically examine how existing methods integrate and implement these components.

- We provide a comprehensive comparative analysis of existing methods, benchmark datasets, and evaluation methodologies, encompassing both traditional recommendation metrics and emerging evaluation approaches specifically designed for LLM-powered agent recommender.

2 Background

2.1 Traditional Recommendation

In conventional recommendation systems, the problem is typically formulated over a user space $\mathcal{U} = [u_1, u_2, ..., u_m]$, an item space $\mathcal{I} = [i_1, i_2, ..., i_n]$, and their interaction matrix $\mathcal{D} \in \mathbb{R}^{m \times n}$. The fundamental goal is to learn a preference function $p : \mathcal{U} \times \mathcal{I} \to \mathbb{R}$ that predicts user preferences:

$$\min_{\theta} \sum_{(u,i) \in \mathcal{D}} \mathcal{L}(p_\theta(u, i), y_{u,i}) , \qquad (1)$$

where $p_\theta(u, i)$ represents the predicted preference and $y_{u,i}$ denotes the ground truth interaction. While various approaches have been proposed, from matrix factorization [Hu et al., 2008] to deep learning [He et al., 2017], these traditional methods face several inherent limitations. First, they struggle to understand complex user intents beyond numerical interactions. Second, they lack the ability to engage in meaningful interactions to explore user preferences. Third, their recommendations often appear as a "black box" without clear explanations for users.

2.2 LLM as Agent

Large Language Model (LLM) as an agent is an emerging research direction that has garnered significant attention [Park et al., 2023]. By transcending the traditional static prompt-response paradigm, it establishes a dynamic decision-making framework [Patil et al., 2023] capable of systematically decomposing complex tasks into manageable components. A typical LLM-powered agent architecture integrates four fundamental modules [Wang et al., 2024a]: (1) the Profile module,

which constructs and maintains comprehensive user feature representations; (2) the Memory module, which orchestrates historical interactions and preserves contextual information for systematic experience accumulation; (3) the Planning module, which formulates strategic policies through sophisticated task decomposition and multi-objective optimization; and (4) the Action module, which executes decisions and facilitates environment interaction. The emergence of pioneering works such as ReAct [Yao et al., 2023], Toolformer [Schick et al., 2023], and HuggingGPT [Shen et al., 2024] has significantly advanced this field.

2.3 LLM Agents for Recommendation

In LLM-powered agent for recommender systems, we formulate the recommendation process through an agent-centric framework. Let $a \in \mathcal{A}$ denote an agent equipped with a set of functional modules $\mathcal{F} = \mathcal{F}_1, \mathcal{F}_2, ..., \mathcal{F}_K$, where each module $\mathcal{F}_k$ represents a specific capability. The recommendation process for a user u can be formally expressed as:

$$\hat{\mathbf{y}}_u = f(\mathcal{F}_k(X_u)), k = 1 \cdots K , \qquad (2)$$

where $X_u \in \mathcal{X}$ represents the input space containing user-specific information (e.g., interaction history, contextual features), and $\hat{\mathbf{y}}_u \in \mathbb{R}^N$ denotes the predicted preference distribution over the item space. The integration function $f : \mathcal{F}_k(X_u) \to \mathbb{R}^N$ synthesizes module outputs to generate final recommendations. Building upon the previously introduced four functional module (Profile, Memory, Planning, and Action), this formulation provides a flexible framework that can accommodate various LLM-powered agent recommendation approaches. These modules operate in a closed-loop framework, where interaction data continuously enriches user profiles and system memory, informing planning strategies that ultimately manifest as personalized recommendations through action execution and feedback collection.

3 Methods

In this section, we sort out existing LLM-powered agent recommendation works based on the overall objective of the method and the agent components of different methods.

3.1 Method Objective

In Table 1, we classify method objectives of existing methods into three categories: recommender-oriented approaches, interaction-oriented methods, and simulation-oriented methods. The illustrations of categories are shown in Figure 1.

(1) **Recommender-oriented** approaches focus on developing intelligent recommendation equipped with enhanced planning, reasoning, memory, and tool-using capabilities. In these approaches, LLMs leverage users' historical behaviors to generate direct recommendation decisions. For instance, as shown in Figure 1, when a user demonstrates recent engagement with technology news and AI-related content, the system might strategically recommend: "Here are 5 articles about latest large language model breakthroughs, 3 introductory articles about machine learning basics, and 2 popular science pieces about AI's impact on society." This paradigm demonstrates how agents can effectively combine their core capabilities to deliver direct item recommendations.

Figure 1: Illustration of Different Method Objectives. We classify existing methods into the following three categories: (1) Recommender-oriented method; (2) Interaction-oriented method; (3) Simulation-oriented method.
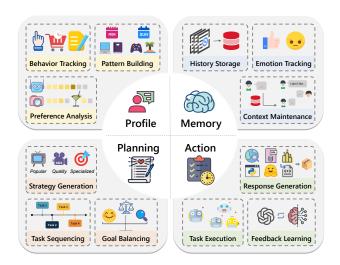

Figure 2: Illustration of Agent Components and Corresponding Functions.

Representative works in this direction include Rec-Mind [Wang *et al.*, 2024b], which develops a unified LLM agent with comprehensive capabilities to generate recommendations directly through LLM outputs. MACRec, which introduces an agent-collaboration mechanism that orchestrates different types of agents to provide personalized recommendations [Wang *et al.*, 2024c].

(2) Interaction-oriented methods focus on enabling natural language interaction and enhancing recommendation interpretability through conversational engagement. These approaches utilize LLMs to conduct human-like dialogues or explanation while making recommendations. For example, as shown in Figure 1, an LLM might respond to a user query with: "I noticed that you like science fiction movies, especially after watching The Descent and Star Trek recently. Considering this preference, I would like to recommend Space Odyssey 2001, a classic film that also explores profound themes about human and alien civilizations. What do you think?" Such interactive recommendations showcase the agent's ability to not only track user preferences but also articulate recommendations in a conversational manner, explaining the reasoning behind suggestions.

AutoConcierge [Zeng *et al.*, 2024] uses natural language conversations to understand user needs and collect user preferences, and uses LLM to understand and generate language, ultimately providing explainable personalized restaurant recommendations. RAH [Shu *et al.*, 2024] is a human-computer interaction recommendation framework based on LLM agents. It realizes personalized recommendations and user intent understanding through the ResSys-Assistant-Human tripartite interaction and the Learn-Act-Critic loop mechanism.

(3) Simulation-oriented methods aim to authentically replicate user behaviors and preferences through sophisticated simulation techniques. These approaches leverage LLMs to generate realistic user responses to recommendations. For instance, when simulating user feedback, an LLM might generate: "As a user who is keen to explore new music, I will click on this new song that combines jazz and electronic elements because it matches my interest in experimental music while maintaining the rhythmic style that I like." These methods focus on using agents to simulate user behaviors and item characteristics in RSs.

Agent4Rec [Zhang *et al.*, 2024a] utilizes LLM-empowered generative agents as user simulators to model authentic interactions between users and recommender systems, aiming to replicate and evaluate realistic user behaviors in recommendation environments. AgentCF [Zhang *et al.*, 2024c] models both users and items as LLM-powered agents that autonomously interact and collaboratively learn from each other to simulate authentic user-item interactions in recommender systems. UserSimulator proposes [Yoon *et al.*, 2024] an evaluation protocol to assess LLMs as generative user simulators in conversational recommendation through five tasks to measure how closely these simulators can emulate authentic user behaviors.

3.2 Agent Components

The LLM-based agent recommendation architecture consists of four main modules: Profile Module, Memory Module, Planning Module, and Action Module. Figure 2 illustrates the core components of the architecture and corresponding functions.

(1) Profile Module is a fundamental component that constructs and maintains dynamic representations of users and items in recommender systems. Through continuous analysis of historical interactions, it captures temporal and contextual patterns in user behavior. For example, when the system observes that a user often browses technology news on weekday

Category	Methods	Profile Module	Memory Module	Planning Module	Action Module
Recommender-oriented Method	RAH [Shu *et al.*, 2024]	×	✓	✓	✓
	ToolRec [Zhao *et al.*, 2024]	×	✓	×	✓
	PMS [Thakkar and Yadav, 2024a]	✓	×	×	✓
	DRDT [Wang *et al.*, 2023b]	×	×	✓	×
	BiLLP [Shi *et al.*, 2024]	×	✓	✓	✓
	RecMind [Wang *et al.*, 2024b]	×	✓	✓	✓
	MACRec [Wang *et al.*, 2024c]	✓	×	✓	✓
Interaction-oriented Method	AutoConcierge [Zeng *et al.*, 2024]	×	✓	✓	✓
	MACRS [Fang *et al.*, 2024]	✓	✓	✓	✓
	RecLLM [Friedman *et al.*, 2023]	✓	✓	×	✓
	InteRecAgent [Huang *et al.*, 2023]	✓	✓	✓	✓
	MAS [Thakkar and Yadav, 2024b]	✓	✓	✓	✓
	H-MACRS [Nie *et al.*, 2024]	✓	✓	×	✓
	Rec4Agentverse [Zhang *et al.*, 2024b]	✓	×	✓	×
Simulation-oriented Method	KGLA [Guo *et al.*, 2024]	✓	✓	×	✓
	CSHI [Zhu *et al.*, 2024a]	✓	✓	×	✓
	SUBER [Corecco *et al.*, 2024]	✓	✓	×	×
	LUSIM [Zhang *et al.*, 2024d]	✓	✓	×	×
	FLOW [Cai *et al.*, 2024]	✓	✓	×	✓
	Agent4Rec [Zhang *et al.*, 2024a]	✓	✓	×	✓
	AgentCF [Zhang *et al.*, 2024c]	✓	✓	×	✓
	UserSimulator [Yoon *et al.*, 2024]	✓	×	×	✓
	RecAgent [Wang *et al.*, 2023a]	✓	✓	×	✓

Table 1: Comparison of Different LLM-powered Agent Recommendation Methods.

mornings and likes to watch travel content on weekends, the Profile Module will build a user profile of "focusing on technology news on weekdays and preferring leisure content on weekends". This adaptive profiling approach integrates behavioral patterns, user preferences, and external knowledge to enable highly personalized recommendations.

The profile module in Agent4Rec [Zhang *et al.*, 2024a] incorporates dual components: quantifiable social traits (activity, conformity, and diversity) and personalized preferences extracted via LLM, enabling a comprehensive simulation of user characteristics. MACRec [Wang *et al.*, 2024c] incorporates a user and item analyst, which play a crucial role in understanding user preferences and item characteristics. AgentCF [Zhang *et al.*, 2024c] constructs natural language-based user profiles to capture dynamic user preferences and item profiles to represent item characteristics and potential adopters' preferences, enabling personalized agent-based collaborative filtering.

(2) Memory Module serves as a contextual brain that manages and leverages historical interactions and experiences to enhance recommendation quality. It maintains a structured repository of past interactions, emotional responses, and conversational context to enable more informed decisions. For example, in a restaurant recommendation scenario, when a user comments "that Sichuan restaurant was too spicy last time", the Memory Module retrieves the specific restaurant reference from historical interactions and incorporates this

preference signal into future recommendations, helping avoid overly spicy options. Through this continuous accumulation and utilization of experiential knowledge, the module enables more personalized and context-aware recommendations that reflect users' past experiences and preferences.

RecAgent [Wang *et al.*, 2023a] comprises three hierarchical levels: sensory memory, short-term memory, and long-term memory. The sensory memory processes environmental inputs, while short-term memory serves as an intermediate layer that can be transformed into long-term memory through repetitive reinforcement. Long-term memory stores crucial reusable information and facilitates self-reflection and knowledge generalization. Agent4Rec [Zhang *et al.*, 2024a] consists of factual memory (recording interactive behaviors) and emotional memory (capturing psychological states), stored in both natural language and vector representations, and managed through three mechanisms: retrieval, writing, and reflection.

(3) Planning Module outputs intelligent recommendation strategies by designing multi-step action plans that balance immediate user satisfaction with long-term engagement goals. It dynamically formulates recommendation trajectories through careful strategy generation and task sequencing. For example, in video recommendation, the system might construct a strategic plan: "first recommend a popular video to establish user interest, and then gradually introduce niche but high-quality related content, while maintaining the diversity of genres, and

ultimately achieve the goal of both satisfying user interest and expanding horizons". Through this planning approach, the module optimizes resource allocation and adapts recommendation sequences to achieve both user engagement and item discovery.

BiLLP [Shi *et al.*, 2024] planning mechanism employs a hierarchical structure with two levels: macro-learning (Planner and Reflector LLMs) generates high-level strategic plans and guidelines from experience, while micro-learning (Actor-Critic) translates these plans into specific recommendations. MACRS [Fang *et al.*, 2024] uses a multi-agent planning system where a Planner Agent coordinates three Responder Agents (Ask, Recommend, Chat) through multi-step reasoning. The system adjusts its dialogue strategy through a feedback mechanism, enabling reflective planning based on user interactions.

(4) Action Module serves as the execution engine that transforms decisions into concrete recommendations through systematic interaction with various system components. For example, in an e-commerce scenario, when receiving the directive "recommend entry-level camera for new user" from the Planning Module, the Action Module executes a coordinated sequence: analyzing purchase patterns of similar users, querying the product database with specific price and feature constraints, generating targeted recommendations, and capturing user feedback. This execution enables the system to deliver contextually appropriate recommendations while continuously learning from interaction outcomes.

RecAgent [Wang *et al.*, 2023a] orchestrates naturalistic agent interactions within recommender systems and social environments through a unified prompting framework, incorporating six action modalities (encompassing search, browse, click, pagination, chat, and broadcast functionalities). InteRecAgent [Huang *et al.*, 2023] action module integrates three core tools (information querying, item retrieval, and item ranking) while leveraging a Candidate Bus for sequential tool communication, enabling an end-to-end interactive process from user queries to final recommendations.

4 Datasets and Evaluations

In this section, we report the datasets and evaluation metrics used by various methods. The dataset information comes from the original source or paper.

4.1 Datasets

Traditional Recommendation Dataset In Table 2, we list several traditional recommendation datasets for evaluating model performance. These datasets provide comprehensive interaction data from various platform, including user-item interactions, timestamps, and review text, enabling the assessment of recommendation models. Several state-of-the-art methods have demonstrated their effectiveness using these datasets.

For instance, the "Books" dataset (10.3M users, 4.4M items) from **Amazon Review data** [McAuley *et al.*, 2015] has been used to evaluate Agent4Rec [Zhang *et al.*, 2024a] and BiLLP [Shi *et al.*, 2024] performance on large-scale tasks, while the "Video Games" dataset (2.8M users, 137.2K items)

has validated DRDT [Wang *et al.*, 2023b] and RAH [Shu *et al.*, 2024] capabilities. The "Beauty" dataset (632K users, 112.6K items) has been utilized by IntcRecAgent [Huang *et al.*, 2023] and DRDT [Wang *et al.*, 2023b] to demonstrate their proficiency in recommendation. These diverse applications underscore the datasets' crucial role in advancing LLM-powered agent recommender systems and providing a foundation for evaluating various of algorithms.

The **MovieLens datasets**, introduced by [Harper and Konstan, 2015], represent another crucial benchmark for evaluating LLM-powered agents recommenders, offering different scales of movie rating data from the MovieLens platform. These datasets range from MovieLens-100K (0.9K users, 1.6K items) to MovieLens-20M (138.5K users, 27.3K items), providing researchers with flexibility in testing their methods across different data scales. Various state-of-the-art approaches have utilized these datasets: FLOW [Cai *et al.*, 2024] and MACRS [Fang *et al.*, 2024] have been validated on the smaller MovieLens-100K dataset, while Agent4Rec [Zhang *et al.*, 2024a], DRDT [Wang *et al.*, 2023b], and MACRS [Fang *et al.*, 2024] have demonstrated their capabilities on MovieLens-1M. The larger variants like MovieLens-10M and MovieLens-20M have been employed by InteRecAgent [Huang *et al.*, 2023] and RecAgent [Yoon *et al.*, 2024] respectively, showcasing the scalability of their approaches. This hierarchical structure of MovieLens datasets makes them particularly valuable for systematically evaluating recommendation algorithms at different scales.

The **Steam**, **Lastfm**, **Anime**, and **Yelp** datasets provide diverse domain-specific evaluation scenarios for LLM-powered agent recommender systems. The Steam dataset, introduced by [Kang and McAuley, 2018], contains 3.7M interactions between 334.7K users and 13K gaming items, and has been extensively used by methods such as Agent4Rec [Zhang *et al.*, 2024a], BiLLP [Shi *et al.*, 2024], FLOW [Cai *et al.*, 2024], and InteRecAgent [Huang *et al.*, 2023] to validate their effectiveness in game recommendation. The Lastfm dataset [Cantador *et al.*, 2011], focusing on music recommendation, comprises 73.5K interactions from 1.2K users on 4.6K music items, and has been specifically utilized by FLOW [Cai *et al.*, 2024] to demonstrate its capabilities in the music domain. Additionally, the Yelp dataset, containing 316.3K interactions between 30.4K users and 20.4K items, has been employed by RecMind [Wang *et al.*, 2024b] to evaluate its performance in recommendations. These domain-specific datasets offer unique evaluation opportunities in specialized recommendation contexts.

Conversational Recommendation Dataset In addition to the above traditional recommendation datasets, some works [Zhu *et al.*, 2024a] evaluate the model performance on conversational datasets. In Table 2, we list three widely-adopted datasets: **ReDial** [Li *et al.*, 2018], **Reddit** [He *et al.*, 2023], and **OpenDialKG** [Moon *et al.*, 2019]. The ReDial dataset comprises 11348 multi-turn dialogues involving 6925 movies, where participants engage in seeker-recommender interactions. The Reddit dataset is derived from movie recommendation discussions within Reddit communities, where users post recommendation requests and receive responses

Category	Datasets	Reference	Users	Items	Interactions	Conversations	Turns	Methods
Traditional Recommendation Dataset	Books	[McAuley *et al.*, 2015]	10.3M	4.4M	29.5M	-	-	Agent4Rec, BiLLP, RAH, SUBER
	CDs and Vinyl		1.8M	701.7K	4.8M	-	-	AgentCF, KGLA, ToolRec
	Video Games		2.8M	137.2K	4.6M	-	-	DRDT, RAH, LUSIM
	Beauty		632.0K	112.6K	701.5K	-	-	InteRecAgent, DRDT, Rec-Mind
	Clothing		22.6M	7.2M	66.0M	-	-	DRDT
	Movies		6.5M	747.8K	17.3M	-	-	RAH, LUSIM
	Office Products		7.6M	710.4K	12.8M	-	-	AgentCF
	Music		101.0K	70.5K	130.4K	-	-	LUSIM
	Movielens-100K	[Harper and Konstan, 2015]	0.9K	1.6K	100K	-	-	FLOW, MACRS, SUBER
	Movielens-1M		6K	3.7K	1.0M	-	-	Agent4Rec, RecAgent, DRDT, MACRS, ToolRec
	Movielens-10M		69.9K	10.6K	10M	-	-	InteRecAgent
	Movielens-20M		138.5K	27.3K	20M	-	-	MACRS, UserSimulator
	Steam	[Kang and McAuley, 2018]	334.7K	13K	3.7M	-	-	Agent4Rec, BiLLP, FLOW, InteRecAgent
	Lastfm	[Cantador *et al.*, 2011]	1.2K	4.6K	73.5K	-	-	FLOW
	Yelp	https://www.yelp.com/dataset	30.4K	20.4K	316.3K	-	-	RecMind, ToolRec, LUSIM
	Anime	https://www.kaggle.com/datasets	73.5K	12.2K	1.05M	-	-	LUSIM
Conversational Recommendation Dataset	ReDial	[Li *et al.*, 2018]	0.9K	51.6K	-	10K	-	UserSimulator, CSHI
	Reddit	[He *et al.*, 2023]	36.2K	51.2K	-	634.4K	1.6M	UserSimulator
	OpenDialKG	[Moon *et al.*, 2019]	-	-	-	15.6K	91.2K	CSHI

Table 2: Summary of Used Experimental Datasets.

with movie suggestions, often accompanied by explanatory rationales. This extensive dataset encompasses 634392 conversations, 1669720 dialogue turns, 36247 users, and 51203 movies. CSHI [Zhu *et al.*, 2024a] employs ReDial (movie domain, including 10006 dialogues) and OpenDialKG (multiple domains, including 13802 dialogues) for performance evaluation. UserSimulator [Yoon *et al.*, 2024] evaluates on the Redial and Reddit datasets in a variety of ways, including behavior simulation and memory module believability, etc. These authentic human-human conversations serve as crucial benchmarks for assessing the model capabilities of LLM-powered agents recommender systems.

It is worth mentioning that considering the agent recommender system based on LLMs, it is necessary to frequently call LLMs or APIs when the model is running. In order to save resources and time, some methods sample data from the original dataset for performance evaluation. For instance, AgentCF [Zhang *et al.*, 2024c] randomly samples two subsets (one dense and one sparse), with each subset containing 100 users. DRDT [Wang *et al.*, 2023b] randomly samples 200 users from each dataset and uses the target items along with 19 randomly sampled items as the candidate item set.

4.2 Evaluation

In Table 3, we summary the evaluation metrics used by recent representative methods.

Standard Recommendation Metrics Most existing methods employ standard recommendation evaluation metrics to assess model performance. The commonly utilized metrics including Normalized Discounted Cumulative Gain (NDCG@K), Recall@K and Hit Ratio@K (HR@K), etc. For instance, AgentCF [Zhang *et al.*, 2024c] evaluates its performance using NDCG@K and Recall@K on the MovieLens-1M dataset. Similarly, DRDT [Wang *et al.*, 2023b] con-

ducts comprehensive evaluations using Recall@10,20 and NDCG@10,20 across multiple datasets including ML-1M, Games, and Luxury datasets. Hit Ratio@K (HR@K) is another crucial metric for evaluating recommendation performance. RecMind [Wang *et al.*, 2024b] employ that for evaluating the recommendation tasks on Amazon Reviews (Beauty) and Yelp datasets.

Language Generation Quality Some methods [Wang *et al.*, 2024b] consider the evaluation of language generation quality (e.g., recommendation explanation generation, review summarization), which primarily rely on BLEU and ROUGE metrics. BLEU measures the precision of generated text against references, while ROUGE evaluates recall-based similarity, enabling comprehensive assessment of language generation capabilities in recommendation scenarios. PMS [Thakkar and Yadav, 2024a] utilizes the ROUGE to evaluate the quality of its generated textual recommendations.

Reinforcement Learning Metrics In evaluating LLM-powered agent recommender systems for long-term engagement, BiLLP [Shi *et al.*, 2024] employs three key metrics adopted from reinforcement learning: trajectory length, average single-round reward, and cumulative trajectory reward. Similarly, LUSIM [Zhang *et al.*, 2024d] uses the total reward to reflect the overall user engagement during the entire interaction process, and the average reward to represent the average quality of a single recommendation. These metrics are to evaluate both immediate recommendation quality and long-term engagement effectiveness.

Conversational Efficiency Metrics Recent research has introduced more comprehensive metrics to evaluate the efficiency of conversational interactions in recommender systems. For instance, MACRS [Fang *et al.*, 2024] employs key interaction-focused metrics such as Success Rate (proportion of successful recommendations) and Average Turn (AT) (num-

Category	Metrics	Methods
Standard Recommendation	NDCG@K, Recall@K, HR@K, Hit@K, MRR, Acc, F1-Score, MAP	DRDT, RecMind, InteRecAgent, RAH, MACRS, PMS, Agent4Rec, AgentCF, KGLA, FLOW, CSHI, ToolRec, SUBER
	RMSE, MAE, MSE	RecMind
Language Generation Quality	BLEU, ROUGE	RecMind, PMS
Reinforcement Learning	Rewards	LUSIM, BiLLP, SUBER
Conversational Efficiency	Average Turn (AT), Success Rate (SR)	InteRecAgent, MACRS, CSHI
Custom Indicators	Proactivity, Economy, Explainability, Correctness, Consistency, Efficiency	AutoConcierge
	Simulated user behaviors believability, Agent memory believability	RecAgent

Table 3: Summary of Used Evaluation Metrics.

ber of interaction rounds needed to reach a recommendation) per session. These metrics assess how effectively the system can understand user preferences and deliver accurate recommendations while minimizing the number of interaction turns.

Custom Indicators Beyond conventional metrics, some methods [Yoon *et al.*, 2024] propose customized evaluation frameworks. AutoConcierge [Zeng *et al.*, 2024] presents six evaluation metrics for task-driven conversational agents: proactivity, economy, explainability, correctness, consistency, and efficiency. RecAgent [Wang *et al.*, 2023a] proposes simulated user behaviors believability and Agent memory believability, to assess the credibility of LLM-simulated user interactions and memory mechanism effectiveness. These metrics assess system engagement, dialogue efficiency, answer interpretability, response accuracy, requirement fulfillment, and response time, respectively.

In all, these metrics prioritize a holistic understanding of conversational performance, emphasizing balance between efficient recommendation delivery, and maintaining high-quality dialogue throughout the recommendation process.

5 Related Research Fields

LLM-powered Recommender Systems In recent years, recommender systems based on Large Language Models (LLMs) have attracted widespread attention. Such systems make full use of the powerful language understanding and generation capabilities of LLMs, bringing a new paradigm to traditional recommender systems. Most existing methods are primarily designed for rating prediction [Bao *et al.*, 2023] and sequential recommendation [Hou *et al.*, 2024; Shao *et al.*, 2024; Zheng *et al.*, 2024]. CoLLM [Zhang *et al.*, 2023] captures and maps the collaborative information through external traditional models, forming collaborative embeddings used by LLMs. LlamaRec [Yue *et al.*, 2023] fine-tunes Llama-2-7b for list-wise ranking of the pre-selected items. However, these methods would face significant limitations: the inability to simulate authentic user behaviors for enhanced personalization, the lack of effective memory mechanisms for long-term context awareness, and the rigid pipeline structure that prevents flexible task decomposition and seamless integration with external tools.

Conversational Recommender Systems Conversational recommender systems (CRS) have emerged as a significant research direction in recent years [Jannach *et al.*, 2021], which are similar to the LLM-powered agent recommender systems. However, traditional methods [Lei *et al.*, 2020] have two main drawbacks: attribute-based approaches are limited by rigid dialogue patterns, while generation-based methods suffer from restricted knowledge and poor generalization capabilities of small language models.

6 Future Directions

Optimization of System Architecture The integration between traditional recommendation methods and LLMs remains insufficient, with challenges in multi-agent collaboration and system interpretability. Future developments should explore flexible architectural designs, enhance agent cooperation efficiency, while ensuring transparency in recommendation.

Refinement of Evaluation Framework There is a notable absence of unified and comprehensive evaluation standards for accurately measuring dialogue quality and recommendation effectiveness. Future research necessitates the establishment of robust evaluation frameworks, development of novel performance metrics, and consideration of privacy and security concerns in practical applications.

Security Recommender System [Ning *et al.*, 2024] reveals the vulnerability of LLM-empowered recommender systems to adversarial attacks. In future, the researchers could develop robust adversarial detection methods, investigate multi-agent defensive architectures, and integrating domain-specific security knowledge into defense mechanisms.

7 Conclusion

The integration of LLM-powered agents into recommender systems has emerged as a significant advancement in recent years. In this survey, we systematically categorize existing approaches into three paradigms: recommender-oriented, interaction-oriented, and simulation-oriented. We comprehensively analyze these methods through a unified four-module architecture and review current datasets and evaluation methodologies. Finally, we identify three promising directions for future research.

References

[Achiam *et al.*, 2023] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

[Bao *et al.*, 2023] Keqin Bao, Jizhi Zhang, Yang Zhang, Wenjie Wang, Fuli Feng, and Xiangnan He. Tallrec: An effective and efficient tuning framework to align large language model with recommendation. In *Recsys*, pages 1007–1014, 2023.

[Cai *et al.*, 2024] Shihao Cai, Jizhi Zhang, Keqin Bao, Chongming Gao, and Fuli Feng. Flow: A feedback loop framework for simultaneously enhancing recommendation and user agents. *arXiv preprint arXiv:2410.20027*, 2024.

[Cantador *et al.*, 2011] Iván Cantador, Peter Brusilovsky, and Tsvi Kuflik. Second workshop on information heterogeneity and fusion in recommender systems (hetrec2011). In *Recsys*, pages 387–388, 2011.

[Corecco *et al.*, 2024] Nathan Corecco, Giorgio Piatti, Luca A Lanzendörfer, Flint Xiaofeng Fan, and Roger Wattenhofer. An llm-based recommender system environment. *arXiv preprint arXiv:2406.01631*, 2024.

[Fang *et al.*, 2024] Jiabao Fang, Shen Gao, Pengjie Ren, Xiuying Chen, Suzan Verberne, and Zhaochun Ren. A multi-agent conversational recommender system. *arXiv preprint arXiv:2402.01135*, 2024.

[Friedman *et al.*, 2023] Luke Friedman, Sameer Ahuja, David Allen, Zhenning Tan, Hakim Sidahmed, Changbo Long, Jun Xie, Gabriel Schubiner, Ajay Patel, et al. Leveraging large language models in conversational recommender systems. *arXiv preprint arXiv:2305.07961*, 2023.

[Guo *et al.*, 2024] Taicheng Guo, Chaochun Liu, Hai Wang, Varun Mannam, Fang Wang, Xin Chen, Xiangliang Zhang, and Chandan K Reddy. Knowledge graph enhanced language agents for recommendation. *arXiv preprint arXiv:2410.19627*, 2024.

[Harper and Konstan, 2015] F Maxwell Harper and Joseph A Konstan. The movielens datasets: History and context. *ACM TIIS*, 5(4):1–19, 2015.

[He *et al.*, 2017] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. Neural collaborative filtering. In *The WebConf*, pages 173–182, 2017.

[He *et al.*, 2023] Zhankui He, Zhouhang Xie, Rahul Jha, Harald Steck, Dawen Liang, Yesu Feng, Bodhisattwa Prasad Majumder, Nathan Kallus, and Julian McAuley. Large language models as zero-shot conversational recommenders. In *CIKM*, pages 720–730, 2023.

[Hou *et al.*, 2024] Yupeng Hou, Junjie Zhang, Zihan Lin, Hongyu Lu, Ruobing Xie, Julian McAuley, and Wayne Xin Zhao. Large language models are zero-shot rankers for recommender systems. In *ECIR*, pages 364–381, 2024.

[Hu *et al.*, 2008] Yifan Hu, Yehuda Koren, and Chris Volinsky. Collaborative filtering for implicit feedback datasets. In *ICDM*, pages 263–272, 2008.

[Huang *et al.*, 2023] Xu Huang, Jianxun Lian, Yuxuan Lei, Jing Yao, Defu Lian, and Xing Xie. Recommender ai agent: Integrating large language models for interactive recommendations. *arXiv preprint arXiv:2308.16505*, 2023.

[Jannach *et al.*, 2021] Dietmar Jannach, Ahtsham Manzoor, Wanling Cai, and Li Chen. A survey on conversational recommender systems. *CSUR*, 54(5):1–36, 2021.

[Kang and McAuley, 2018] Wang-Cheng Kang and Julian McAuley. Self-attentive sequential recommendation. In *ICDM*, pages 197–206. IEEE, 2018.

[Lei *et al.*, 2020] Wenqiang Lei, Gangyi Zhang, Xiangnan He, Yisong Miao, Xiang Wang, Liang Chen, and Tat-Seng Chua. Interactive path reasoning on graph for conversational recommendation. In *KDD*, pages 2073–2083, 2020.

[Li *et al.*, 2018] Raymond Li, Samira Ebrahimi Kahou, Hannes Schulz, Vincent Michalski, Laurent Charlin, and Chris Pal. Towards deep conversational recommendations. *NuerIPS*, 31, 2018.

[McAuley *et al.*, 2015] Julian McAuley, Christopher Targett, Qinfeng Shi, and Anton Van Den Hengel. Image-based recommendations on styles and substitutes. In *SIGIR*, pages 43–52, 2015.

[Moon *et al.*, 2019] Seungwhan Moon, Pararth Shah, Anuj Kumar, and Rajen Subba. Opendialkg: Explainable conversational reasoning with attention-based walks over knowledge graphs. In *ACL*, pages 845–854, 2019.

[Nie *et al.*, 2024] Guangtao Nie, Rong Zhi, Xiaofan Yan, Yufan Du, Xiangyang Zhang, Jianwei Chen, Mi Zhou, Hongshen Chen, Tianhao Li, Ziguang Cheng, et al. A hybrid multi-agent conversational recommender system with llm and search engine in e-commerce. In *Recsys*, pages 745–747, 2024.

[Ning *et al.*, 2024] Liang-bo Ning, Shijie Wang, Wenqi Fan, Qing Li, Xin Xu, Hao Chen, and Feiran Huang. Cheatagent: Attacking llm-empowered recommender systems via llm agent. In *KDD*, pages 2284–2295, 2024.

[Park *et al.*, 2023] Joon Sung Park, Joseph O'Brien, Carrie Jun Cai, Meredith Ringel Morris, Percy Liang, and Michael S Bernstein. Generative agents: Interactive simulacra of human behavior. In *AASUIST*, pages 1–22, 2023.

[Patil *et al.*, 2023] Shishir G Patil, Tianjun Zhang, Xin Wang, and Joseph E Gonzalez. Gorilla: Large language model connected with massive apis. *arXiv preprint arXiv:2305.15334*, 2023.

[Schick *et al.*, 2023] Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Eric Hambro, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. Toolformer: Language models can teach themselves to use tools. In *NuerIPS*, volume 36, 2023.

[Shao *et al.*, 2024] Minglai Shao, Hua Huang, Qiyao Peng, and Hongtao Liu. Ulmrec: User-centric large language model for sequential recommendation. *arXiv preprint arXiv:2412.05543*, 2024.

[Shen *et al.*, 2024] Yongliang Shen, Kaitao Song, Xu Tan, Dongsheng Li, Weiming Lu, and Yueting Zhuang. Hugginggpt: Solving ai tasks with chatgpt and its friends in hugging face. 36, 2024.

[Shi *et al.*, 2024] Wentao Shi, Xiangnan He, Yang Zhang, Chongming Gao, Xinyue Li, Jizhi Zhang, Qifan Wang, and Fuli Feng. Large language models are learnable planners for long-term recommendation. In *SIGIR*, pages 1893–1903, 2024.

[Shu *et al.*, 2024] Yubo Shu, Haonan Zhang, Hansu Gu, Peng Zhang, Tun Lu, Dongsheng Li, and Ning Gu. Rah! recsys–assistant–human: A human-centered recommendation framework with llm agents. *IEEE TCSS*, 2024.

[Thakkar and Yadav, 2024a] Param Thakkar and Anushka Yadav. Personalized recommendation systems using multimodal, autonomous, multi agent systems. *arXiv preprint arXiv:2410.19855*, 2024.

[Thakkar and Yadav, 2024b] Param Thakkar and Anushka Yadav. Personalized recommendation systems using multimodal, autonomous, multi agent systems. *arXiv preprint arXiv:2410.19855*, 2024.

[Wang *et al.*, 2023a] Lei Wang, Jingsen Zhang, Hao Yang, Zhiyuan Chen, Jiakai Tang, Zeyu Zhang, Xu Chen, Yankai Lin, Ruihua Song, Wayne Xin Zhao, et al. User behavior simulation with large language model based agents. *arXiv preprint arXiv:2306.02552*, 2023.

[Wang *et al.*, 2023b] Yu Wang, Zhiwei Liu, Jianguo Zhang, Weiran Yao, Shelby Heinecke, and Philip S Yu. Drdt: Dynamic reflection with divergent thinking for llm-based sequential recommendation. *arXiv preprint arXiv:2312.11336*, 2023.

[Wang *et al.*, 2024a] Lei Wang, Chen Ma, Xueyang Feng, Zeyu Zhang, Hao Yang, Jingsen Zhang, Zhiyuan Chen, Jiakai Tang, Xu Chen, Yankai Lin, et al. A survey on large language model based autonomous agents. *Frontiers of Computer Science*, 18(6):186345, 2024.

[Wang *et al.*, 2024b] Yancheng Wang, Ziyan Jiang, Zheng Chen, Fan Yang, Yingxue Zhou, Eunah Cho, Xing Fan, Yanbin Lu, Xiaojiang Huang, and Yingzhen Yang. Recmind: Large language model powered agent for recommendation. In *Findings of NAACL*, pages 4351–4364, 2024.

[Wang *et al.*, 2024c] Zhefan Wang, Yuanqing Yu, Wendi Zheng, Weizhi Ma, and Min Zhang. Macrec: A multi-agent collaboration framework for recommendation. In *SIGIR*, pages 2760–2764, 2024.

[Wu *et al.*, 2022] Le Wu, Xiangnan He, Xiang Wang, Kun Zhang, and Meng Wang. A survey on accuracy-oriented neural recommendation: From collaborative filtering to information-rich recommendation. *IEEE TKDE*, 35(5):4425–4445, 2022.

[Yao *et al.*, 2023] Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik R Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models. In *ICLR*, 2023.

[Yoon *et al.*, 2024] Se-eun Yoon, Zhankui He, Jessica Maria Echterhoff, and Julian McAuley. Evaluating large language models as generative user simulators for conversational recommendation. *arXiv preprint arXiv:2403.09738*, 2024.

[Yue *et al.*, 2023] Zhenrui Yue, Sara Rabhi, Gabriel de Souza Pereira Moreira, Dong Wang, and Even Oldridge. Llamarec: Two-stage recommendation using large language models for ranking. *arXiv preprint arXiv:2311.02089*, 2023.

[Zeng *et al.*, 2024] Yankai Zeng, Abhiramon Rajasekharan, Parth Padalkar, Kinjal Basu, Joaquín Arias, and Gopal Gupta. Automated interactive domain-specific conversational agents that understand human dialogs. In *ISPADL*, pages 204–222, 2024.

[Zhang *et al.*, 2023] Yang Zhang, Fuli Feng, Jizhi Zhang, Keqin Bao, Qifan Wang, and Xiangnan He. Collm: Integrating collaborative embeddings into large language models for recommendation. *arXiv preprint arXiv:2310.19488*, 2023.

[Zhang *et al.*, 2024a] An Zhang, Yuxin Chen, Leheng Sheng, Xiang Wang, and Tat-Seng Chua. On generative agents in recommendation. In *SIGIR*, pages 1807–1817, 2024.

[Zhang *et al.*, 2024b] Jizhi Zhang, Keqin Bao, Wenjie Wang, Yang Zhang, Wentao Shi, Wanhong Xu, Fuli Feng, and Tat-Seng Chua. Prospect personalized recommendation on large language model-based agent platform. *arXiv preprint arXiv:2402.18240*, 2024.

[Zhang *et al.*, 2024c] Junjie Zhang, Yupeng Hou, Ruobing Xie, Wenqi Sun, Julian McAuley, Wayne Xin Zhao, Leyu Lin, and Ji-Rong Wen. Agentcf: Collaborative learning with autonomous language agents for recommender systems. In *The WebConf*, pages 3679–3689, 2024.

[Zhang *et al.*, 2024d] Zijian Zhang, Shuchang Liu, Ziru Liu, Rui Zhong, Qingpeng Cai, Xiangyu Zhao, Chunxu Zhang, Qidong Liu, and Peng Jiang. Llm-powered user simulator for recommender system. *arXiv preprint arXiv:2412.16984*, 2024.

[Zhao *et al.*, 2024] Yuyue Zhao, Jiancan Wu, Xiang Wang, Wei Tang, Dingxian Wang, and Maarten De Rijke. Let me do it for you: Towards llm empowered recommendation via tool learning. In *SIGIR*, pages 1796–1806, 2024.

[Zheng *et al.*, 2024] Bowen Zheng, Yupeng Hou, Hongyu Lu, Yu Chen, Wayne Xin Zhao, Ming Chen, and Ji-Rong Wen. Adapting large language models by integrating collaborative semantics for recommendation. In *ICDE*, pages 1435–1448. IEEE, 2024.

[Zhu *et al.*, 2024a] Lixi Zhu, Xiaowen Huang, and Jitao Sang. A llm-based controllable, scalable, human-involved user simulator framework for conversational recommender systems. *arXiv preprint arXiv:2405.08035*, 2024.

[Zhu *et al.*, 2024b] Xi Zhu, Yu Wang, Hang Gao, Wujiang Xu, Chen Wang, Zhiwei Liu, Kun Wang, Mingyu Jin, Linsey Pang, Qingsong Wen, et al. Recommender systems meet large language model agents: A survey. *SSRN 5062105*, 2024.