Multi-Agent Video Recommenders: Evolution, Patterns, and Open Challenges

Srivaths Ranganathan[1], **Abhishek Dharmaratnakar**[2], **Anushree Sinha**[3], **Debanshu Das**[4]

[1]Google LLC, Mountain View, USA
[2]Google LLC, Mountain View, USA
[3]Google LLC, Mountain View, USA
[4]Google LLC, Mountain View, USA

Abstract

Video recommender systems are among the most popular and impactful applications of AI, shaping content consumption and influencing culture for billions of users. Traditional single-model recommenders, which optimize static engagement metrics, are increasingly limited in addressing the dynamic requirements of modern platforms. In response, multi-agent architectures are redefining how video recommender systems serve, learn, and adapt to both users and datasets. These agent-based systems coordinate specialized agents responsible for video understanding, reasoning, memory, and feedback, to provide precise, explainable recommendations.

In this survey, we trace the evolution of multi-agent video recommendation systems (MAVRS). We combine ideas from multi-agent recommender systems, foundation models, and conversational AI, culminating in the emerging field of large language model (LLM)-powered MAVRS. We present a taxonomy of collaborative patterns and analyze coordination mechanisms across diverse video domains, ranging from short-form clips to educational platforms. We discuss representative frameworks, including early multi-agent reinforcement learning (MARL) systems such as MMRF and recent LLM-driven architectures like MACRec and Agent4Rec, to illustrate these patterns. We also outline open challenges in scalability, multimodal understanding, incentive alignment, and identify research directions such as hybrid reinforcement learning–LLM systems, lifelong personalization and self-improving recommender systems.

1. Introduction and Motivation

Recommender systems (RSs) have become essential for navigating the vast and growing landscape of video on the internet (Liebman, Saar-Tsechansky, and Stone 2015; Adomavicius and Tuzhilin 2005; Ricci, Rokach, and Shapira 2011). They curate personalized feeds, improve user satisfaction, and support the attention economy across platforms for short-form entertainment, music streaming, live broadcasts, and educational media. The large-scale, high-impact nature of modern video recommenders makes them a perfect testing ground for developing and validating LLM-powered multi-agent systems.

Accepted at the AAAI 2026 Bridge Program on Advancing LLM-Based Multi-Agent Collaboration (WMAC 2026).

Conventional RS pipelines, whether collaborative filtering (Koren, Bell, and Volinsky 2009; Rendle 2010), deep sequential models (Kang and McAuley 2018; Sun et al. 2019), or reinforcement-learning optimizers (Mnih et al. 2015; Sutton and Barto 2018), operate largely as *single-agent systems*, optimizing one global objective (e.g., click-through rate or watch time). This paradigm not only neglects competing goals, such as diversity, fairness, and explainability (Zhang and Chen 2020; Burke 2017), but also hinders the system from adapting to the dynamic and complex nature of real-world environments, including heterogeneous content, evolving user intent, and complex feedback loops. (Quadrana, Cremonesi, and Jannach 2018; He et al. 2017).

Recent progress in multi-agent learning has introduced decentralized and cooperative paradigms that decompose the recommendation process into interacting roles. Each agent can specialize in tasks, such as perception, reasoning, or feedback integration, jointly optimizing a shared objective through communication and coordination (Wang et al. 2024a, 2025). These developments reveal that a multi-agent design can solve more complex user problems, increasing recommendation quality and user engagement (Boadana et al. 2025).

Concurrently, the emergence of foundation models (FMs) [large language and multimodal models trained on vast corpora] has transformed how recommender systems can represent, reason, and interact (Vaswani et al. 2017; Devlin et al. 2019; Brown et al. 2020). FMs enable zero-shot generalization (He et al. 2023; Ranganathan et al. 2025), natural-language interfaces, and cross-modal reasoning over text, vision, and audio. When coupled with multi-agent coordination, they form the basis of agentic recommender systems which autonomously plan, reflect, use tools and coordinate with other agents to achieve their goals. (He et al. 2020; Wang et al. 2025).

Despite this rapid progress, the field lacks a unified taxonomy that bridges classical multi-agent reinforcement learning with these emerging foundation-model paradigms across diverse video ecosystems (Wu et al. 2023; Zhang, Yang, and Basar 2021). Prior surveys have focused either on Multi-Agent RL or on foundation models in traditional recommendation systems or collaboration in generic multi-agent systems, leaving a gap in understanding how these streams converge in modern recommender systems (Zhou et al. 2024a).

Overall, this work aims to build that bridge for the domain of multi-agent video recommendation systems (MAVRS), outlining a pathway toward self-improving, transparent, and trustworthy next-generation video recommenders. Although this paper focuses on "video" recommenders, some of the underlying principles can be generalized to other recommendation domains.

2. Background and Related Work

Before the advent of multi-agent and LLM-driven frameworks, the field of recommender systems was dominated by two primary paradigms: collaborative filtering and content-based filtering. Collaborative filtering (CF) operates on the principle of homophily, identifying users with similar taste profiles to make recommendations based on what analogous users have enjoyed (Ricci, Rokach, and Shapira 2011). Content-based (CB) methods, in contrast, focus on the intrinsic properties of items and recommend content with features similar to those a user has previously rated positively (Adomavicius and Tuzhilin 2005; Koren, Bell, and Volinsky 2009). While often effective, these classical approaches face challenges such as the "cold start" problem for new users or items, data sparsity in user-item interaction matrices, and a limited ability to capture the dynamic, multi-faceted nature of user intent (Burke 2017). These challenges paved the way for more complex, decentralized models, which form the basis of modern multi-agent systems (Quadrana, Cremonesi, and Jannach 2018; He et al. 2017; Sun et al. 2019; Zhang et al. 2019).

Multi-Agent Recommender Systems

Early multi-agent recommender systems (MARS) emerged from distributed AI research, where the goal was to decompose recommendation subtasks among cooperative software entities (Wooldridge 2009; Selmi, Brahmi, and Gammoudi 2014). Selmi *et al.* (2014) identified four canonical roles: *interface* agents that interact with users, *filtering* agents that match items to preferences, *learning* agents that update profiles, and *mediator* agents that resolve conflicts across heterogeneous sources. Subsequent systems incorporated negotiation, trust modeling, and content aggregation to enhance autonomy and scalability (Burke 2017). Although these designs improved modularity, they relied heavily on symbolic reasoning and rule-based communication, limiting adaptability in large-scale, dynamic video environments. The success of deep reinforcement learning (DRL)—notably the Deep Q-Network (DQN) (Mnih et al. 2015)—catalyzed a wave of research towards optimizing multi-agent recommender systems using DRL (Sutton and Barto 2018; Liebman, Saar-Tsechansky, and Stone 2015). In MARL, multiple agents learn coordinated policies through shared or partially shared rewards. Model-based methods such as MMRF optimize heterogeneous feedback signals (e.g., watch-time, like-rate, dwell-time) using attention-based message passing among agents, yielding stable off-policy learning (Wang et al. 2025).

Foundation-Model-Powered Recommendation

Foundation models (FMs)—large language and multimodal transformers—have redefined how recommender systems can represent and reason about content (Vaswani et al. 2017; Devlin et al. 2019; Brown et al. 2020; Radford et al. 2021; Alayrac et al. 2022). Large Language Models (LLMs) provide enhanced generalization abilities, having trained on extensive datasets, allowing them to understand complex patterns and handle new items or user trends effectively (Chowdhery et al. 2023; Touvron et al. 2023). They offer improved explanation and reasoning capabilities by providing more comprehensive and context-aware justifications (Ouyang et al. 2022; Zhang and Chen 2020). Additionally, LLMs enhance personalization and interactivity through their natural language processing features, enabling dynamic adaptation to user feedback and preferences (Chen, Yu, and Huang 2024). They can also allow users to have more fine-tuned control over the system's understanding of user preferences and, subsequently, the recommended content (Boadana et al. 2025).

LLMs have been integrated into RS through three main paradigms: (i) *feature-based*, using FMs as embedding extractors for user and item representations (Wang et al. 2024a); (ii) *generative*, treating recommendation as text or sequence generation by prompting or fine-tuning (Brown et al. 2020; Devlin et al. 2019); and (iii) *agentic*, where the LLM serves as the core of autonomous reasoning that plans, memorizes, and interacts through natural language (Boadana et al. 2025).

Agentic Frameworks

Recent studies combine multi-agent coordination with LLM reasoning to create conversational and collaborative recommenders. An LLM-based recommender agent is an autonomous entity designed to perceive its environment, make decisions, and take actions within a recommendation scenario (Chen, Yu, and Huang 2024).

MACRec and its extension MACRS organize LLM agents into hierarchical roles—manager, analyst, searcher, reflector, and interpreter—to perform sequential and dialogue-based tasks (Wang et al. 2024a). EmotionRec and MusicAgent further incorporate multimodal affect detection, enabling personalized music and video recommendation grounded in user emotion and context (Boadana et al. 2025; Yu et al. 2023). These systems demonstrate that emotional awareness and cooperative reasoning can significantly enhance engagement and trust (Wang et al. 2025).

3. Collaborative Multi-Agent Video Recommender Patterns

The collaborative interactions between LLM agents in video recommendation can be categorized into distinct architectural patterns. This taxonomy classifies systems according to the primary mechanism of agent interaction and the overarching goal of the collaboration, revealing how different structures are engineered to solve specific problems. The following sections detail prominent architectures, each illustrated with a key example from recent research (Wang et al. 2025).

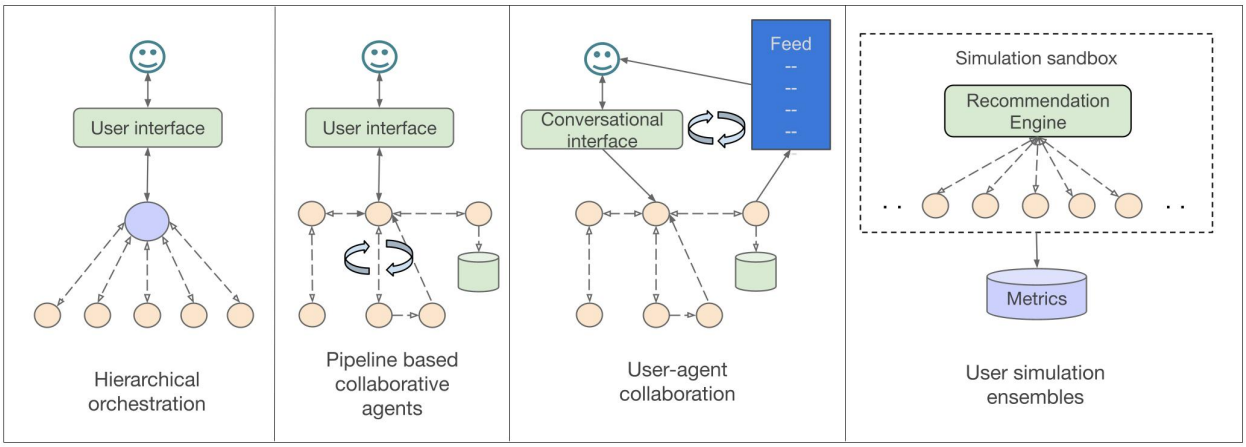

Figure 1: Illustration of Multi-agent Video Recommender patterns highlighting an example for each pattern in Section 3.

3.1. Hierarchical Orchestration

This architecture employs a central coordinating agent that directs the actions and integrates the outputs of specialized, subordinate agents to achieve a unified objective. The collaboration pattern is explicitly top-down, with the coordinating agent orchestrating the contributions of the agentic group. Subordinate agents may operate in two primary modes: (1) collaboratively, to jointly identify an optimal recommendation, or (2) *competitively*, proposing distinct recommendations from which the coordinating agent selects based on user signals or other optimization criteria (Rahwan et al. 2019; Wang et al. 2021).

A prominent example of this model is the **Model-based Multi-agent Ranking Framework (MMRF)**, (Zhou et al. 2024b) designed to maximize user WatchTime on a short-video platform. In MMRF, a main agent is dedicated to the primary objective (WatchTime) and is supported by auxiliary agents, each tasked with maximizing a secondary user interaction signal (e.g., Follow, Like, Comment). Coordination is achieved via an "Attentive Collaboration Mechanism," which permits the main agent to dynamically weigh and integrate salient information from the auxiliary agents. This hierarchical structure allows the system to optimize for a primary metric while strategically leveraging correlated signals from secondary user preferences.

The **MMAgentRec** system (Xiao 2025), applied in the tourism domain, presents a conceptual variation. It prompts a single LLM to simulate multiple expert personas from diverse domains (e.g., natural sciences, social sciences, humanities), which then provide interdisciplinary advice on a user's request. This framework also incorporates a "reflection mechanism," enabling the LLM to self-critique its outputs and refine its decision-making (Ouyang et al. 2022). This approach leverages the LLM's latent knowledge by structuring its reasoning process as an internal, collaborative dialogue among simulated experts (Boadana et al. 2025).

This architectural pattern can be generalized to multiple, distinct agents, each parameterized with specific prompts or inputs to optimize for different objectives. In a video RS context, this could be implemented as specialized agents recommending content from different domains (e.g., News, Education, Music) or optimizing for divergent engagement goals (e.g., long-term user value vs. short-term engagement) (Chen, Wang, and Chen 2023; Wang et al. 2025).

3.2. Pipeline-based Modular Collaboration

In this architectural pattern, agents operate sequentially, forming a processing pipeline where each agent executes a distinct, specialized task. The output of one agent serves as the direct input for the next, establishing a modular workflow that decomposes a complex problem into manageable stages. This pattern is analogous to traditional, non-agentic industry systems where distinct engineering teams manage separate data processing pipelines (e.g., video processing and indexing, user history summarization, model training) that write intermediate outputs to offline databases (Zhou et al. 2024a; Adomavicius and Tuzhilin 2005; He et al. 2017; Da Silva, Marcolino et al. 2023).

The **VRAgent-R1** system demonstrates this approach, utilizing a two-stage pipeline to enhance video recommendation performance:

1. **Item Perception (IP) Agent:** This initial agent processes raw, multimodal video content. It employs a "human-like progressive thinking" process to move beyond surface-level features, generating an enhanced semantic summary that captures latent, recommendation-relevant semantics (Radford et al. 2021; Alayrac et al. 2022; Li et al. 2023).
2. **User Simulation (US) Agent:** The semantic summary from the IP Agent enriches the base recommender model's item representations. The US Agent leverages this enhanced understanding to simulate user decisions. This agent's feedback is integrated into a reinforcement learning (RL) loop, with rewards for predicting the next video watched by the user and for providing Chain of

Thought reasoning of whether a user would like a specific video. The resulting learned policy is better aligned with human preferences, and subsequently generates higher-quality recommendations (Mnih et al. 2015; Sutton and Barto 2018).

In contrast to the two-stage VRAgent-R1, the authors of **MACRec** propose a conversational recommender system with an alternative task decomposition (Wang et al. 2024b):

- **Manager:** Assigns sub-tasks to other agents, aggregates their responses, and reasons about the task status to generate a final response to the user or instantiate new sub-agents.
- **Reflector:** Evaluates the Manager's proposed response and provides critical feedback for improvement. The Manager uses this feedback to decide whether to share the current recommendation with the user or iterate further (Ouyang et al. 2022).
- **User/Item Analyst:** Provides a nuanced analysis of both user preferences and item content. This role is analogous to the combined functions of the IP and US agents in VRAgent-R1.
- **Searcher:** Executes search queries and summarizes the results for the Manager. This two-stage process (search-then-summarize) optimizes token consumption for the Manager agent (Boadana et al. 2025).
- **Task Interpreter:** Interfaces with the user, converting natural language queries into structured task descriptions for the Manager. It also maintains the conversational state and history across multiple Manager calls (Fang et al. 2024; Huang et al. 2025a).

3.3. User-Agent Collaboration

In this architecture, multiple agents collaborate internally to power a single, user-facing conversational interface (within a broader recommendation surface) where the primary objective is not to provide recommendations, but to empower the end-user with direct, intuitive control over their recommendation feed, thereby enhancing their "sense of agency" (Floridi and Cowls 2019).

TKGPT (Niu, Vishnuvardhan, and Punnam 2025) is a system designed around this principle. It functions as an LLM-enhanced chatbot that allows users to modify their TikTok "For You" page through natural language. This is achieved through a partnership between two internal assistants:

1. The **Recommender Assistant** interprets the user's conversational requests to generate relevant keywords for video topics.
2. The **Sorting Assistant** uses the LLM to assign weights to these keywords, which determine the *proportion* of videos for each topic in the next batch of 32 videos. These videos are then shuffled and presented to the user.

This collaboration translates a user's natural language intent into concrete algorithmic adjustments via a proportional allocation and batch-based update mechanism, creating a direct and transparent control interface (Huang et al. 2025a; Fang et al. 2024).

3.4. User Simulation Agent Ensembles

This architecture uses agents not as the core recommender, but as a simulated population of users. The goal is to generate high-fidelity synthetic interaction data, which can be used to evaluate system performance offline, train other models, or study complex user behavior phenomena without the cost and risk of live A/B testing (Wang et al. 2025; Rahwan et al. 2019).

Agent4Rec (Zhang et al. 2024a) is the primary example of this pattern, creating a simulator with thousands of LLM-empowered generative agents (Wang et al. 2025). Each agent is initialized from real-world datasets with a detailed profile, including unique tastes and social traits like activity (interaction frequency) and conformity (alignment with popular sentiment). The central goal is to achieve "agent alignment" by ensuring simulated behaviors are faithful to those of real humans, allowing the ensemble to replicate effects like the "filter bubble" (Zhang et al. 2024c). The **US Agent** from VRAgent-R1 also serves as a simulation agent. These two systems exemplify different philosophies for achieving alignment: Agent4Rec relies on rich, static profiling initialized from real data, whereas VRAgent-R1's US Agent uses a dynamic, in-loop training method—Reinforcement Learning with Group Relative Policy Optimization (GRPO)—to continuously align its behavior with real user decisions (Chen et al. 2025).

This simulation pattern can be used to create a sandbox for testing multi-agent systems' insights on social norms and governance. For example, Agent4Rec's modeling of user ensembles allows researchers to prototype various agent incentive formulations and observe emergent behaviors (like filter bubbles) without real-world risk.

4. Agent-centric Evaluation

Evaluating multi-agent recommender systems (MARS) differs fundamentally from classical single-model recommenders because multiple agents interact, negotiate, and learn concurrently (Dafoe et al. 2021; Zhang et al. 2024c). Standard metrics such as Precision@K and NDCG remain necessary to measure the quality of the recommendations (Adomavicius and Tuzhilin 2005; He et al. 2017) but are insufficient to capture coordination, reasoning quality, and emergent behaviors of the agentic framework itself (Huang et al. 2025a; Zhang et al. 2024b). Multi-agent RS may also perform better for nuanced or niche recommendations which are often in the tail-end of frequency in most evaluation datasets, but can influence a user's subjective evaluation of a RS (Quadrana, Cremonesi, and Jannach 2018; Burke 2017).

A comprehensive evaluation must therefore be multi-dimensional, assessing not only the final output but also the internal processes of the agents (Zhou et al. 2024a; Zhang et al. 2024c). We propose five key dimensions for a holistic, agent-centric evaluation.

4.1. Task-Specific Quality

Table 1: Evaluation of collaborative multi-agent video recommender architectures. Metrics emphasize coordination, user alignment, and computational feasibility.

Pattern	Primary Evaluation Focus	Representative Metrics	Critical Failure Points & Risks
Hierarchical Orchestration (e.g., MMRF, MMAgentRec)	*Orchestration Effectiveness:* How well does the central agent integrate diverse sub-goals to optimize the primary system objective?	Main objective metric (e.g., WatchTime), contribution weights (from attentive mechanism), system-wide latency.	*Coordinator Bottleneck:* The central agent becomes a single point of failure. *Conflicting Goals:* Auxiliary agents may work at cross-purposes, harming the main objective.
Pipeline-based Modular (e.g., VRAgent-R1, MACRec)	*End-to-End Task Quality:* How well does the final output perform after passing through all sequential stages?	Quality of intermediate outputs, error propagation rate, end-to-end latency.	*Compounding Errors and Brittleness:* An error in an early agent (e.g., IP Agent) can degrade the entire chain.
User-Agent Collaboration (e.g., TKGPT)	*User-Perceived Agency:* Does the user feel in control and satisfied with the system's response to their natural language commands?	User satisfaction (SUS scores), task success rate (from user studies), latency from command to feed update.	*Misinterpretation:* The system may misunderstand the user's (often ambiguous) intent and make drastic, undesirable changes to recommendations.
User Simulation Ensemble (e.g., Agent4Rec)	*Behavioral Fidelity:* How accurately does the simulated agent population replicate the statistical properties of real human users?	KL divergence (or similar) between simulated and real interaction distributions; replication of known macro-effects (e.g., filter bubbles).	*Lack of Generalization:* Agents overfit to initialization data and fail to model novel behaviors. *Prohibitive Cost:* High computational overhead for running thousands of LLM agents.

This dimension evaluates the performance of an individual agent on its specialized sub-task, separate from the final recommendation (Zhang et al. 2024c; Ouyang et al. 2022).

- **For Perception Agents** (e.g., the IP Agent in VRAgent-R1): Evaluation can involve comparing the agent-generated representation/summary for a sample of videos against human-generated summaries or ground-truth labels using metrics like ROUGE, BERTScore, or emotion-based recognition signals (Radford et al. 2021; Alayrac et al. 2022; Li et al. 2023; Chaugule et al. 2016).
- **For Reasoning Agents** (e.g., the "reflection mechanism" in MMAgentRec): Evaluation is often qualitative, assessing the logical coherence, factuality, and self-correction capability of the agent's internal monologue or "scratchpad" (Ouyang et al. 2022).
- **For Specialized Recommenders** (e.g., the auxiliary agents in MMRF): These can be evaluated on their own proxy metrics (e.g., can the 'Like' agent predict 'Likes' with high precision?) .

4.2. Coordination & Collaboration Efficiency

This dimension assesses the *interaction* between agents, focusing on the overhead and effectiveness of their collaboration.

- **Communication Overhead:** This is a critical metric for LLM-based systems, measured in the number of tokens, messages, or API calls exchanged between agents to reach a decision. The "Searcher" agent in MACRec is an example of a design that explicitly optimizes this (Huang et al. 2025a; Zhang et al. 2024c).
- **Latency:** The end-to-end time from user request to final recommendation. This is vital for real-time video feeds and includes the cumulative processing and communication time of all agents in the chain (Dafoe et al. 2021; Huang et al. 2025b).
- **Contribution Alignment:** In hierarchical systems like MMRF, this measures whether the auxiliary agents' contributions (e.g., 'Follow' signal) are weighted appropriately and genuinely improve the main agent's primary objective ('WatchTime').

4.3. System-Level & Emergent Properties

This dimension evaluates the macro-behavior of the entire system, particularly its stability and adaptability (Zhang et al. 2024c; Fang et al. 2024).

- **Robustness & Fault Tolerance:** This tests how the system handles the failure of a single agent. Does a pipeline-based system collapse (a "brittle" failure), or can a hierarchical system's coordinator route around the failed agent (Fang et al. 2024; Dafoe et al. 2021)?
- **Adaptability:** This measures how quickly the agent ensemble can adapt to new items, new user interests, or a shift in the data distribution. This is a key goal for systems using RL (like VRAgent-R1) and "lifelong personalization" (Mnih et al. 2015; Sutton and Barto 2018; Chen, Yu, and Huang 2024).
- **Emergent Behavior Accuracy:** For user simulation ensembles like Agent4Rec, this is the primary evaluation. It involves measuring the statistical divergence (e.g., KL divergence) between the simulated interaction data and real user data (Zhang et al. 2024c,b).

4.4. Human-Alignment & User-Centric Metrics

This dimension moves beyond offline metrics to measure the system's impact on the end-user experience, which is often the primary goal (Zhang et al. 2024b,c; Dafoe et al. 2021; Chaugule et al. 2016).

- **Controllability & Agency:** For systems like TKGPT, the core metric is the user's "sense of agency." This is measured via user studies, assessing whether users feel their natural language commands are correctly interpreted and

lead to a satisfying change in their feed (Zhang et al. 2024b,c).

- **Explainability:** A MARS architecture should naturally provide better explainability (Zhang and Chen 2020; Zhang et al. 2019). Evaluation can involve user studies where participants rate the quality of explanations generated by the system (e.g., "The 'Education' agent suggested this video, and the 'Sorting' agent prioritized it because you asked for 'deep dives'") (Zhang et al. 2024b; Dafoe et al. 2021).
- **Trustworthiness:** This is a longitudinal user-study metric measuring whether users trust the system's recommendations and explanations over time (Zhang et al. 2024c; Floridi and Cowls 2019).
- **Fairness:** The quality of reasoning agents and user simulation agents strongly affects bias in the recommendations for specific slices or users or content (Burke 2017; Mehrabi et al. 2021). Standard fairness metrics that measure equal exposure for items, such as Jain's Index or Gini Index, and metrics based on user group disparity (like Equalized Odds or Demographic Parity) can be used to measure end-to-end fairness (Wang et al. 2023; Zhang et al. 2024b).

4.5. Scalability & Economic Viability

This practical dimension assesses the cost of deploying and maintaining the MARS (Shleifer, Nguyen, and Liu 2023; Chen, Zhou, and Yu 2024; Zhang et al. 2024c).

- **Inference Cost and Latency:** For LLM-driven agents, this is the total token cost per user request or per recommendation batch and the end-to-end latency for the coordinating agents to generate a recommendation (Shleifer, Nguyen, and Liu 2023; Zhang et al. 2024c).
- **Training & Alignment Cost:** For systems using RL (VRAgent-R1) or large-scale simulation (Agent4Rec), this measures the computational resources (GPU hours, real-user data) required to train or align the agents before they produce high-fidelity results (Wu et al. 2024; Zhang et al. 2024c).

5. Challenges and Open Problems

Despite the rapid progress in LLM-powered multi-agent recommenders, deploying MAVRS at industry scale presents significant challenges, limiting their current utility and trustworthiness (Zhou et al. 2024a; Chen et al. 2025).

5.1 Computational Cost and Scalability

The reliance on large language models (LLMs) as the cognitive core for agents introduces significant computational and financial overhead. Architectures like Agent4Rec, which simulate *thousands* of agents, are prohibitively expensive for most research labs and impractical for real-time training or inference in production RS (Shleifer, Nguyen, and Liu 2023). Lightweight, "distilled" agent models or more efficient token-sharing mechanisms might offer a path forward to widespread adoption (Chen, Zhou, and Yu 2024; Zhou et al. 2024a).

5.2 Multimodal Grounding and Reasoning

Video is an inherently dense medium packed with informationa cross modalities: visual, audio, textual and temporal. Current agents, especially those built on text-centric LLMs, struggle to "ground" their reasoning in this rich data. While systems like VRAgent-R1 employ an *Item Perception (IP) Agent* to generate semantic summaries, this is often a lossy compression (Li et al. 2023). The challenge lies in enabling agents to perform deep, cross-modal reasoning directly on video streams, moving beyond metadata and text summaries to cohesively understanding the content of the video (Alayrac et al. 2022; Radford et al. 2021; He et al. 2023; Huang et al. 2025a).

5.3 Evaluation

As discussed in the previous section, evaluating the performance of complex, collaborative agent systems is an open problem. Offline metrics (e.g., nDCG, MRR) may not capture the subjective benefits of context-aware, conversational recommendation (Adomavicius and Tuzhilin 2005; He et al. 2017). Furthermore, user simulation ensembles (Agent4Rec, VRAgent-R1) face an alignment problem: ensuring that synthetic agent behavior is a high-fidelity proxy for real human behavior, including irrationality, conformity, and drift (Chen et al. 2025; Rahwan et al. 2019). Without robust validation, it is difficult to trust simulation-based findings or offline training (Zhang et al. 2024b).

5.4 Controllability and Trustworthiness

As agents become more autonomous, ensuring they are controllable, robust, and aligned with human values becomes essential (Floridi and Cowls 2019; Zhang et al. 2024b; Huang et al. 2025a). In hierarchical systems (MMRF), a subordinate agent could diverge and optimize its secondary metric at the expense of the primary goal (Chen et al. 2025). In conversational systems (TKGPT), the translation of user intent into algorithmic action must be transparent and faithful (Li et al. 2023; Zhang and Chen 2020). Agents could also fail in a silent, opaque manner, causing errors to propagate through other downstream agents (Rahwan et al. 2019; Ouyang et al. 2022).

5.5 Incentive Alignment

In multi-agent systems, agents must be *incentivized* to collaborate effectively (Rahwan et al. 2019). In current recommenders, this is implicit (e.g., optimizing a shared goal). However, as systems grow in complexity, agents with different objectives (e.g., user WatchTime vs. user Likes in MMRF) may enter into conflict. A key challenge is to design explicit coordination mechanisms, potentially borrowing from computational economics (e.g., auctions, contract theory) (Zhang et al. 2024c; Ostrom 1990). These mechanisms can help the high-level agent ensure subordinate agents

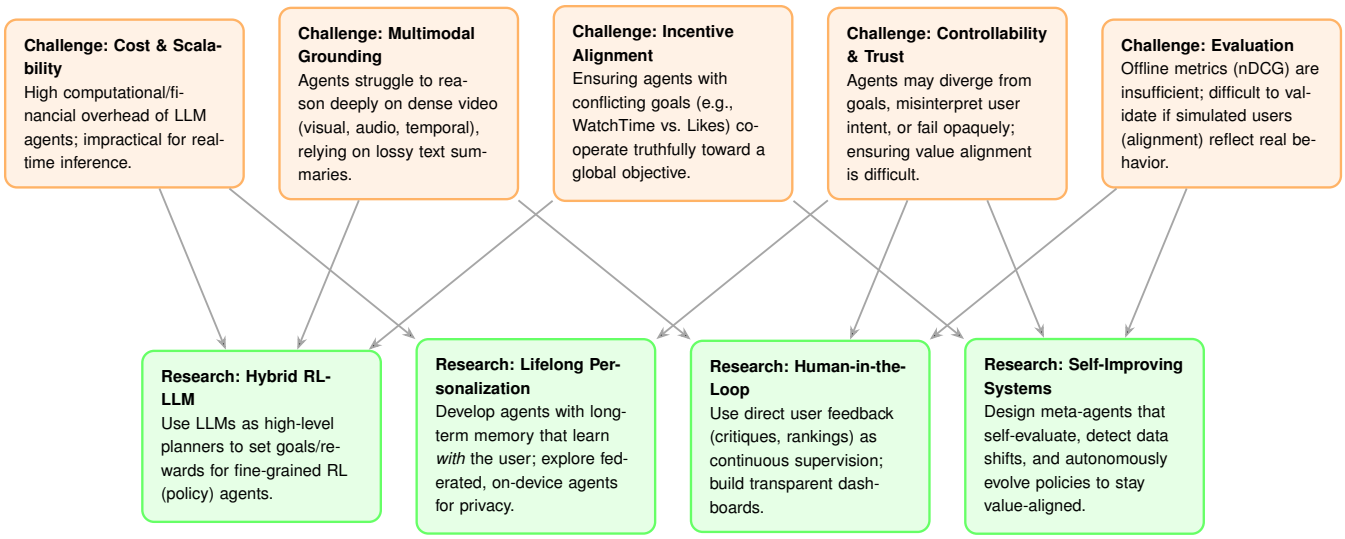

Figure 2: Challenges and Future Research Directions for Multi-Agent Video Recommendation Systems (MAVRS).

cooperate truthfully and robustly toward the global system objective, even under uncertainty or conflicting signals (Huang et al. 2025a; Zhang et al. 2024b). However, unlike computational economics, incentives in LLM-based agents are configured via natural language, which allows room for the underlying LLM to interpret the prompt in ways that differ from what the developer intended. (Yang et al. 2020)

6. Future Directions

Addressing the challenges above requires unifying algorithmic efficiency, realistic evaluation, and human alignment (Fang et al. 2024; Zhang et al. 2024b). Future research should treat multi-agent recommendation as a socio-technical system integrating cognition, collaboration, and ethics (Floridi and Cowls 2019; Rahwan et al. 2019).

These challenges also highlight specific directions for future research, focusing on the development of more intelligent, adaptive, and human-centric systems.

6.1. Hybrid RL-LLM Architectures

A promising frontier is the deeper integration of Reinforcement Learning (RL) and LLMs. LLMs excel at high-level reasoning, planning, and understanding user intent (as seen in TKGPT or the *Manager* MACRec), while RL excels at fine-grained policy optimization in dynamic environments (as seen in VRAgent-R1). Future systems may use an LLM as a ”planner” to set high-level goals or generate reward-shaping functions for a subordinate RL agent, creating a hybrid system that is both context-aware and adaptive to user feedback (Sutton and Barto 2018; Mnih et al. 2015; Zhang et al. 2024c). These emerging “planner–executor” hybrid systems show promise for scaling such coordination while maintaining explainablity (Garnelo and Shanahan 2019; Li et al. 2023).

6.2. Lifelong Personalization and Agent Memory

Current models largely operate on a session- or user-profile-level memory. The next step is lifelong personalization, where agents build and maintain a dynamic, long-term memory of user preferences and evolving interests. This involves moving beyond static profiles (Agent4Rec) to models where agents can reason over their interaction history, self-correct past assumptions, and proactively adapt to a user’s long-term personal journey, effectively *learning with* the user. This requires new designs for maintaining a summarized version of long-term user preference history (Chen, Yu, and Huang 2024; Li, Wang, and Xu 2024; Wang, Huang, and Wu 2025). A promising research area here is Federated Collaboration, which applies federated learning principles to the multi-agent paradigm. A local ”User Profile Agent,” co-located with the user (such as on the device), could perform deep, lifelong personalization using raw interaction data that never leaves the device. The local agent can interact with online RS agents while optimizing for privacy and user well-being (Shleifer, Nguyen, and Liu 2023; Huang et al. 2025a).

6.3. Human-in-the-Loop Validation

Long-term trust depends on user participation (Burke 2017; Zhang and Chen 2020). Crowdsourced or platform-integrated feedback, where users critique and rank recommendations, can serve as continuous supervision (Huang et al. 2025a; Fang et al. 2024). Interactive dashboards visualizing reasoning and fairness trade-offs will enhance transparency and literacy among users and regulators (Zhang et al. 2024b; Floridi and Cowls 2019). In the long term, we can derive these signals directly using optimized multimodal affect detection (e.g., facial expression or tone analysis) to enhance personalization (Chaugule et al. 2016).

6.4. Toward Self-Improving Recommenders

The next frontier is self-governing ecosystems where agents perceive, reason, and evolve collaboratively (Fang et al. 2024; Wang et al. 2025). Such multi-agent architectures should enable a meta-agent to evaluate reasoning quality, detect distributional shifts, and autonomously propose schema or policy updates (Chen, Yu, and Huang 2024; Huang et al. 2025a). The system should understand cause and effect and evolve its strategies to achieve better outcomes than optimizing for short-term objectives like watch time (Peters, Janzing, and Schölkopf 2017; Schölkopf et al. 2021). By self-reflecting to continuously optimizing the behavior and incentives of the modular internal agents, these multi-agent systems can evolve from content delivery tools into recommenders that are closely aligned with human values (Floridi and Cowls 2019; Zhang et al. 2024b).

References

Adomavicius, G.; and Tuzhilin, A. 2005. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering*, 17(6): 734–749.

Alayrac, J.-B.; Donahue, J.; Luc, P.; Miech, A.; Barr, I.; Hasson, Y.; et al. 2022. Flamingo: a visual language model for few-shot learning. *Advances in Neural Information Processing Systems*, 35: 23716–23730.

Boadana, R. C.; da Costa Junior, A. G.; Rios, R.; and da Silva, F. S. 2025. LLM-based intelligent agents for music recommendation: A comparison with classical content-based filtering. *arXiv preprint arXiv:2508.11671*.

Brown, T.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J. D.; Dhariwal, P.; et al. 2020. Language models are few-shot learners. *Advances in Neural Information Processing Systems*, 33: 1877–1901.

Burke, R. 2017. Multisided fairness for recommendation. *ACM Transactions on Recommender Systems*, 1(1): 1–32.

Chaugule, V.; Abhishek, D.; Vijayakumar, A.; Ramteke, P. B.; and Koolagudi, S. G. 2016. Product Review Based on Optimized Facial Expression Detection. In *Proceedings of the Ninth International Conference on Contemporary Computing (IC3)*, 1–6. IEEE.

Chen, B.; Yu, T.; and Huang, C. 2024. Lifelong personalization with LLM-based agentic recommenders. *arXiv preprint arXiv:2408.11567*.

Chen, C.; Wang, S.; and Chen, L. 2023. Multi-Objective Recommendation: Theory, Methods, and Applications. *IEEE Transactions on Knowledge and Data Engineering*.

Chen, K.; Zhou, D.; and Yu, T. 2024. Efficient foundation model fine-tuning for large-scale recommender systems. *arXiv preprint arXiv:2406.00132*.

Chen, S.; Chen, B.; Yu, C.; Luo, Y.; Ouyang, Y.; Lei, C.; Zhuo, C.; Zang, L.; and Wang, Y. 2025. VRAgent-R1: Boosting video recommendation with MLLM-based agents via reinforcement learning. *arXiv preprint arXiv:2507.02626*.

Chowdhery, A.; Narang, S.; Devlin, J.; Bosma, M.; Mishra, G.; Roberts, A.; Barham, P.; Chung, H. W.; Sutton, C.; Gehrmann, S.; et al. 2023. Palm: Scaling language modeling with pathways. *Journal of Machine Learning Research*, 24(240): 1–113.

Da Silva, F.; Marcolino, L.; et al. 2023. A Survey on Multi-Agent Reinforcement Learning: From Decentralized to Hierarchical Architectures. *IEEE Transactions on Artificial Intelligence*.

Dafoe, A.; Bachrach, Y.; Hadfield, G.; Horvitz, E.; Larson, K.; and Graepel, T. 2021. Cooperative AI: Machines must learn to find common ground. *Nature*, 593(7857): 33–36.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*, 4171–4186.

Fang, J.; Gao, S.; Ren, P.; Chen, X.; Verberne, S.; and Ren, Z. 2024. A multi-agent conversational recommender system. *arXiv preprint arXiv:2402.01135*.

Floridi, L.; and Cowls, J. 2019. Establishing the rules for building trustworthy AI. *Nature Machine Intelligence*, 1(6): 261–262.

Garnelo, M.; and Shanahan, M. 2019. Reconciling deep learning with symbolic artificial intelligence: representing objects and relations. *Current Opinion in Behavioral Sciences*, 29: 17–23.

He, X.; Liao, L.; Zhang, H.; Nie, L.; Hu, X.; and Chua, T.-S. 2017. Neural collaborative filtering. In *Proceedings of the 26th International Conference on World Wide Web (WWW)*, 173–182.

He, Z.; Chen, X.; Zhang, H.; Ma, W.; and Zhang, M. 2020. Multi-Module Cooperation for Recommendation via Reinforcement Learning. In *Proceedings of the 14th ACM Conference on Recommender Systems (RecSys)*, 160–169. ACM.

He, Z.; Xie, Z.; Jha, R.; Steck, H.; Liang, D.; Feng, Y.; Majumder, B. P.; Kallus, N.; and McAuley, J. 2023. Large language models as zero-shot conversational recommenders. In *Proceedings of the 32nd ACM international conference on information and knowledge management*, 720–730.

Huang, C.; Huang, H.; Yu, T.; Xie, K.; Wu, J.; Zhang, S.; McAuley, J.; Jannach, D.; and Yao, L. 2025a. A survey of foundation model-powered recommender systems: From feature-based, generative to agentic paradigms. *IEEE Transactions on Knowledge and Data Engineering*.

Huang, C.; Wu, J.; Xia, Y.; Yu, Z.; Wang, R.; Yu, T.; Zhang, R.; Rossi, R. A.; Kveton, B.; Zhou, D.; McAuley, J.; and Yao, L. 2025b. Towards agentic recommender systems in the era of multimodal large language models. *arXiv preprint arXiv:2503.16734*.

Kang, W.-C.; and McAuley, J. 2018. Self-attentive sequential recommendation. In *Proceedings of the 2018 IEEE International Conference on Web Search and Data Mining (WSDM)*, 197–206.

Koren, Y.; Bell, R.; and Volinsky, C. 2009. Matrix factorization techniques for recommender systems. *Computer*, 42(8): 30–37.

Li, J.; Li, D.; Savarese, S.; and Hoi, S. 2023. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, 19730–19742. PMLR.

Li, X.; Wang, R.; and Xu, J. 2024. P4LM: Policy learning with pretrained language models for recommender adaptation. *arXiv preprint arXiv:2403.09145*.

Liebman, E.; Saar-Tsechansky, M.; and Stone, P. 2015. DJ-MC: A reinforcement-learning agent for music playlist recommendation. In *Proceedings of the 14th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, 591–598. Istanbul, Turkey.

Mehrabi, N.; Morstatter, F.; Saxena, N.; Lerman, K.; and Galstyan, A. 2021. A survey on bias and fairness in machine learning. *ACM Computing Surveys*, 54(6): 1–35.

Mnih, V.; Kavukcuoglu, K.; Silver, D.; Rusu, A. A.; et al. 2015. Human-level control through deep reinforcement learning. *Nature*, 518(7540): 529–533.

Niu, S.; Vishnuvardhan, D.; and Punnam, V. S. R. 2025. Chat with the 'For You'Algorithm: An LLM-Enhanced Chatbot for Controlling Video Recommendation Flow. In *Proceedings of the 7th ACM Conference on Conversational User Interfaces*, 1–16.

Ostrom, E. 1990. *Governing the Commons: The Evolution of Institutions for Collective Action*. Cambridge University Press.

Ouyang, L.; Wu, J.; Jiang, X.; Almeida, D.; Wainwright, C.; Mishkin, P.; Zhang, C.; Agarwal, S.; Slama, K.; Ray, A.; et al. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35: 27730–27744.

Peters, J.; Janzing, D.; and Schölkopf, B. 2017. *Elements of Causal Inference: Foundations and Learning Algorithms*. Cambridge, MA: MIT Press. ISBN 9780262037310.

Quadrana, M.; Cremonesi, P.; and Jannach, D. 2018. Sequence-aware recommender systems. *ACM Computing Surveys*, 51(4): 1–36.

Radford, A.; Kim, J. W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; et al. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, 8748–8763. PmLR.

Rahwan, I.; Cebrian, M.; Obradovich, N.; Bongard, J.; Bonnefon, J.-F.; Breazeal, C.; Crandall, J. W.; Christakis, N. A.; Couzin, I. D.; Jackson, M. O.; et al. 2019. Machine behaviour. *Nature*, 568(7753): 477–486.

Ranganathan, S.; Lo, C.; Cunha, B.; Khani, N.; Wei, L.; Nath, A.; Andrews, S.; Varady, G.; Song, Y.; Klingenhoefer, J.; et al. 2025. Zero-shot Cross-domain Knowledge Distillation: A Case study on YouTube Music. In *Proceedings of the Nineteenth ACM Conference on Recommender Systems*, 1122–1125.

Rendle, S. 2010. Factorization machines. In *Proceedings of the IEEE International Conference on Data Mining (ICDM)*, 995–1000.

Ricci, F.; Rokach, L.; and Shapira, B. 2011. *Recommender Systems Handbook*. Springer.

Schölkopf, B.; Locatello, F.; Bauer, S.; Ke, N. R.; Kalchbrenner, N.; Goyal, A.; and Bengio, Y. 2021. Toward Causal Representation Learning. *Proceedings of the IEEE*, 109(5): 612–634.

Selmi, A.; Brahmi, Z.; and Gammoudi, M. 2014. Multi-agent recommender system: State of the art. In *Proceedings of the 16th international conference on information and communications security*.

Shleifer, S.; Nguyen, T.; and Liu, P. 2023. The cost of inference for large models and recommender deployment. *arXiv preprint arXiv:2312.07110*.

Sun, F.; Liu, J.; Wu, J.; Pei, C.; Lin, X.; Ou, W.; and Jiang, P. 2019. BERT4Rec: Sequential recommendation with bidirectional encoder representations from transformer. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management (CIKM)*, 1441–1450.

Sutton, R. S.; and Barto, A. G. 2018. Reinforcement learning: An introduction. *MIT Press*.

Touvron, H.; Lavril, T.; Izacard, G.; et al. 2023. LLaMA: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, L.; and Polosukhin, I. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems (NeurIPS)*, 5998–6008.

Wang, H.; Zhang, F.; Xie, X.; and Guo, M. 2021. Dueling Bandit Gradient Descent for Recommender Systems. *IEEE Transactions on Knowledge and Data Engineering*, 33(5): 2183–2195.

Wang, Q.; Huang, Z.; Jia, R.; Debevec, P.; and Yu, N. 2025. MAViS: A multi-agent framework for long-sequence video storytelling. *arXiv preprint arXiv:2508.08487*.

Wang, R.; Huang, C.; and Wu, J. 2025. Rec-R1: Towards reinforcement-tuned recommender agents. *arXiv preprint arXiv:2501.08765*.

Wang, W.; Lin, X.; Feng, F.; He, X.; and Chua, T.-S. 2024a. Generative recommendation: Towards next-generation recommender paradigm. *ACM Transactions on Recommender Systems*, 1(1): 1–25.

Wang, Y.; Ma, W.; Zhang, M.; Liu, Y.; and Ma, S. 2023. A survey on the fairness of recommender systems. *ACM Transactions on Information Systems*, 41(3): 1–43.

Wang, Z.; Yu, Y.; Zheng, W.; Ma, W.; and Zhang, M. 2024b. Macrec: A multi-agent collaboration framework for recommendation. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2760–2764.

Wooldridge, M. 2009. *An introduction to multiagent systems*. John Wiley & Sons.

Wu, J.; Huang, C.; Yu, T.; and Yao, L. 2023. A survey on multimodal recommender systems: Taxonomy, challenges and future directions. *arXiv preprint arXiv:2304.03516*.

Wu, J.; Li, Y.; Zhao, J.; and Tang, J. 2024. The Economics of Agent-Based AI Systems: Cost, Efficiency, and Market Dynamics. *ACM Transactions on Recommender Systems*, 2(3): 1–25. Examines cost-efficiency tradeoffs and scaling economics in multi-agent and LLM-driven recommender systems.

Xiao, X. 2025. MMAgentRec, a personalized multi-modal recommendation agent with large language model. *Scientific Reports*, 15(1): 12062.

Yang, J.; Li, A.; Farajtabar, M.; Sunehag, P.; Hughes, E.; and Zha, H. 2020. Learning to incentivize other learning agents. *Advances in Neural Information Processing Systems*, 33: 15208–15219.

Yu, D.; Song, K.; Lu, P.; He, T.; Tan, X.; Ye, W.; Zhang, S.; and Bian, J. 2023. MusicAgent: An AI agent for music understanding and generation with large language models. *arXiv preprint arXiv:2310.11954*.

Zhang, A.; Chen, Y.; Sheng, L.; Wang, X.; and Chua, T.-S. 2024a. On generative agents in recommendation. In *Proceedings of the 47th international ACM SIGIR conference on research and development in Information Retrieval*, 1807–1817.

Zhang, K.; Yang, Z.; and Basar, T. 2021. Multi-Agent Reinforcement Learning: A Selective Overview of Theories and Algorithms. *IEEE Transactions on Artificial Intelligence*, 2(2): 320–340.

Zhang, S.; Huang, C.; Yu, T.; and Yao, L. 2024b. Trust and transparency in agentic recommender systems. *arXiv preprint arXiv:2409.12021*.

Zhang, S.; Yao, L.; Sun, A.; and Tay, Y. 2019. Deep learning based recommender system: A survey and new perspectives. *ACM Computing Surveys*, 52(1): 1–38.

Zhang, Y.; and Chen, X. 2020. Explainable recommendation: A survey and new perspectives. *Foundations and Trends in Information Retrieval*, 14(1): 1–101.

Zhang, Y.; Chen, X.; Wang, S.; and Chen, L. 2024c. Generative Agents for Recommender Systems: Challenges and Opportunities. *ACM Transactions on Recommender Systems*.

Zhou, K.; Zhang, S.; Yu, T.; and Yao, L. 2024a. A survey on large language model applications in recommender systems. *arXiv preprint arXiv:2402.05120*.

Zhou, P.; Xu, X.; Hu, L.; Li, H.; and Jiang, P. 2024b. A Model-based Multi-Agent Personalized Short-Video Recommender System. *arXiv preprint arXiv:2405.01847*.