A Survey of Large Language Model for Recommendation System

1 Abstract

The rapid advancement of artificial intelligence, particularly through large language models (LLMs), has significantly reshaped the landscape of recommendation systems by addressing longstanding challenges such as data sparsity, cold-start problems, and limited explainability. This survey paper explores the integration of LLMs into recommendation pipelines, focusing on their role in enhancing user preference modeling, sequential recommendations, and system explainability. A comprehensive taxonomy is presented, categorizing LLM-based frameworks into three tiers: representing and understanding, scheming and utilizing, and industrial deployment, each highlighting distinct aspects of LLM integration. Key advancements include hybrid approaches combining explicit and implicit preferences, strategies for mitigating cold-start issues, and innovations in explainability and fairness through causal and counterfactual analysis. Additionally, the survey examines the fusion of LLMs with graph neural networks and multimodal data processing, emphasizing innovations like token alignment and unified latent spaces. The survey contributes a structured overview of methodologies, identifies critical challenges such as scalability and interpretability, and synthesizes findings from recent studies to guide future research directions. By bridging theoretical foundations with practical applications, this work aims to facilitate the development of more personalized, scalable, and user-centric recommendation systems.

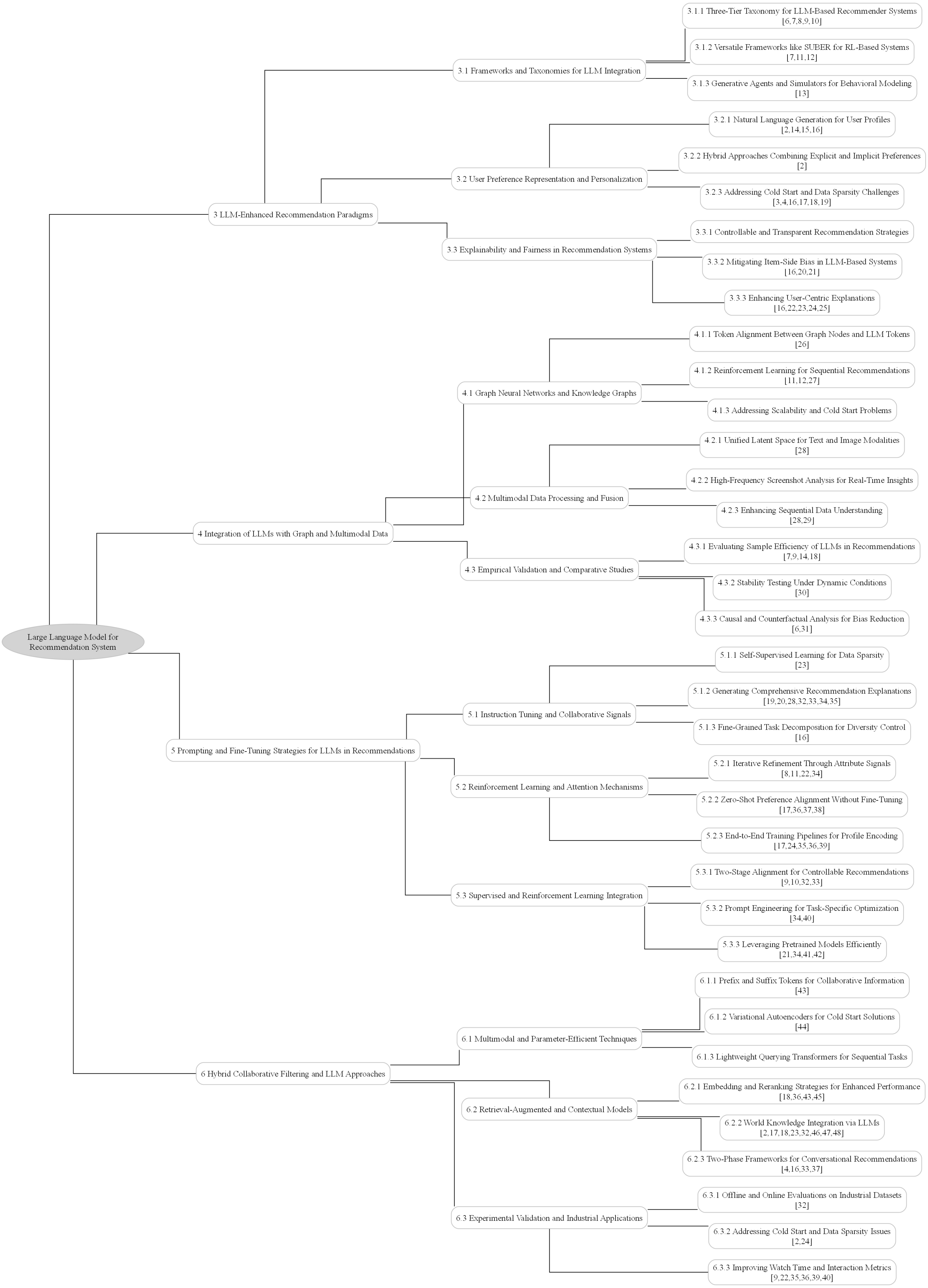
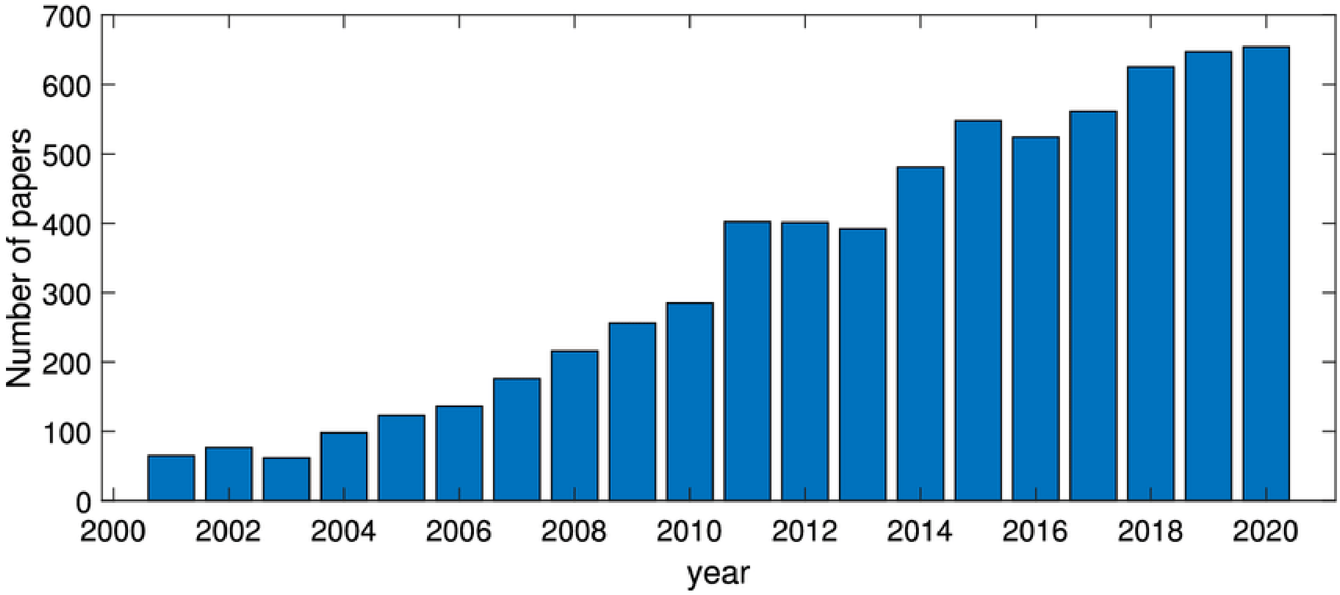

Large Language Model for Recommendation System
3 LLM-Enhanced Recommendation Paradigms
3.1 Frameworks and Taxonomies for LLM Integration
3.1.1 Three-Tier Taxonomy for LLM-Based Recommender Systems [6,7,8,9,10]
3.1.2 Versatile Frameworks like SUBER for RL-Based Systems [7,11,12]
3.1.3 Generative Agents and Simulators for Behavioral Modeling [13]
3.2 User Preference Representation and Personalization
3.2.1 Natural Language Generation for User Profiles [2,14,15,16]
3.2.2 Hybrid Approaches Combining Explicit and Implicit Preferences [2]
3.2.3 Addressing Cold Start and Data Sparsity Challenges [3,4,16,17,18,19]
3.3 Explainability and Fairness in Recommendation Systems
3.3.1 Controllable and Transparent Recommendation Strategies
3.3.2 Mitigating Item-Side Bias in LLM-Based Systems [16,20,21]
3.3.3 Enhancing User-Centric Explanations [16,22,23,24,25]
4 Integration of LLMs with Graph and Multimodal Data
4.1 Graph Neural Networks and Knowledge Graphs
4.1.1 Token Alignment Between Graph Nodes and LLM Tokens [26]
4.1.2 Reinforcement Learning for Sequential Recommendations [11,12,27]
4.1.3 Addressing Scalability and Cold Start Problems
4.2 Multimodal Data Processing and Fusion
4.2.1 Unified Latent Space for Text and Image Modalities [28]
4.2.2 High-Frequency Screenshot Analysis for Real-Time Insights
4.2.3 Enhancing Sequential Data Understanding [28,29]
4.3 Empirical Validation and Comparative Studies
4.3.1 Evaluating Sample Efficiency of LLMs in Recommendations [7,9,14,18]
4.3.2 Stability Testing Under Dynamic Conditions [30]
4.3.3 Causal and Counterfactual Analysis for Bias Reduction [6,31]
5 Prompting and Fine-Tuning Strategies for LLMs in Recommendations
5.1 Instruction Tuning and Collaborative Signals
5.1.1 Self-Supervised Learning for Data Sparsity [23]
5.1.2 Generating Comprehensive Recommendation Explanations [19,20,28,32,33,34,35]
5.1.3 Fine-Grained Task Decomposition for Diversity Control [16]
5.2 Reinforcement Learning and Attention Mechanisms
5.2.1 Iterative Refinement Through Attribute Signals [8,11,22,34]
5.2.2 Zero-Shot Preference Alignment Without Fine-Tuning [17,36,37,38]
5.2.3 End-to-End Training Pipelines for Profile Encoding [17,24,35,36,39]
5.3 Supervised and Reinforcement Learning Integration
5.3.1 Two-Stage Alignment for Controllable Recommendations [9,10,32,33]
5.3.2 Prompt Engineering for Task-Specific Optimization [34,40]
5.3.3 Leveraging Pretrained Models Efficiently [21,34,41,42]
6 Hybrid Collaborative Filtering and LLM Approaches
6.1 Multimodal and Parameter-Efficient Techniques
6.1.1 Prefix and Suffix Tokens for Collaborative Information [43]
6.1.2 Variational Autoencoders for Cold Start Solutions [44]
6.1.3 Lightweight Querying Transformers for Sequential Tasks
6.2 Retrieval-Augmented and Contextual Models
6.2.1 Embedding and Reranking Strategies for Enhanced Performance [18,36,43,45]
6.2.2 World Knowledge Integration via LLMs [2,17,18,23,32,46,47,48]
6.2.3 Two-Phase Frameworks for Conversational Recommendations [4,16,33,37]
6.3 Experimental Validation and Industrial Applications
6.3.1 Offline and Online Evaluations on Industrial Datasets [32]
6.3.2 Addressing Cold Start and Data Sparsity Issues [2,24]
6.3.3 Improving Watch Time and Interaction Metrics [9,22,35,36,39,40]

Fig 1. The outline of the Large Language Model for Recommendation System

2 Introduction



(Number of publications containing the phrase “recommender systems” up to 2020)

The rapid advancement of artificial intelligence has significantly transformed the landscape of recommendation systems, which play a pivotal role in modern digital platforms [2]. Traditional methods primarily relied on collaborative filtering and content-based approaches to predict user preferences and suggest relevant items [3]. However, these techniques often encountered limitations such as data sparsity, cold-start problems, and difficulties in capturing nuanced user-item relationships. The emergence of large language models (LLMs) has introduced new possibilities for addressing these challenges by leveraging their deep semantic understanding and generative capabilities [4]. This evolution underscores the growing importance of integrating advanced AI models into recommendation pipelines to create more personalized, explainable, and scalable solutions [5].

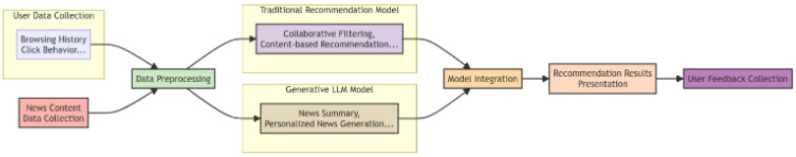
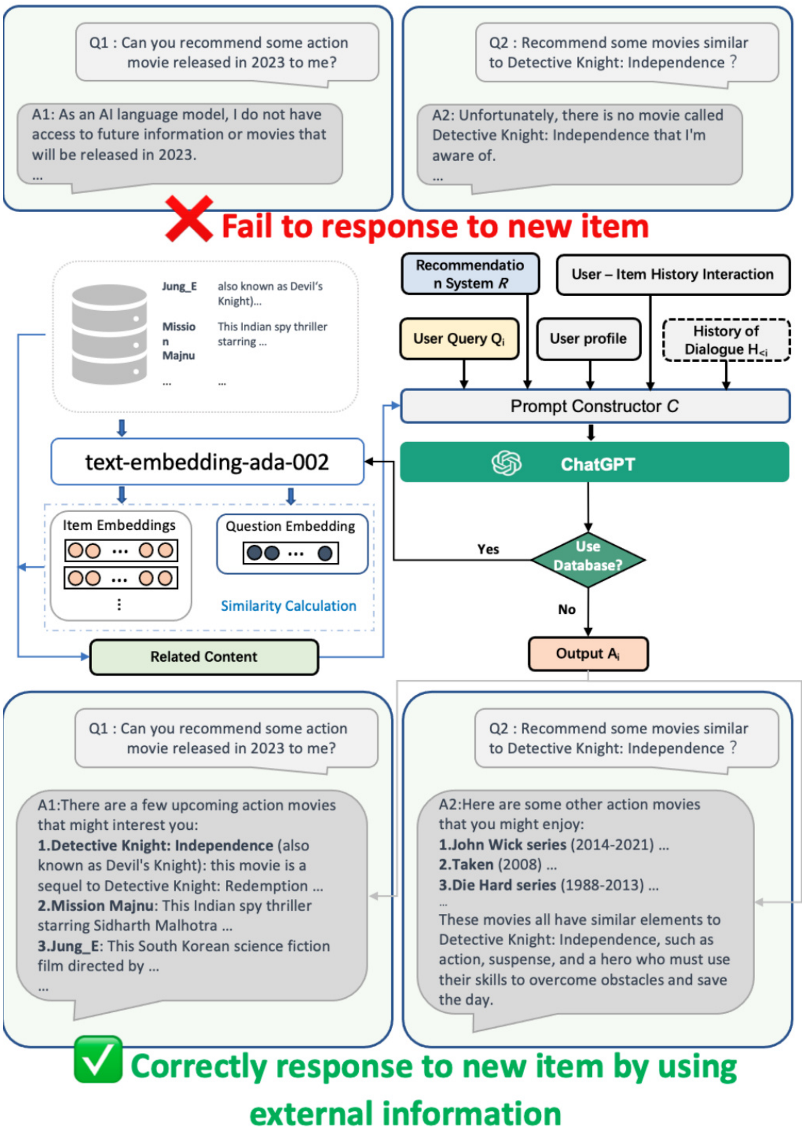


Fig 2: Chart from 'A review of methods using large language models in news recommendation systems'



Q1 : Can you recommend some action movie released in 2023 to me?

A1: As an AI language model, I do not have access to future information or movies that will be released in 2023. ...

Q2 : Recommend some movies similar to Detective Knight: Independence ?

A2: Unfortunately, there is no movie called Detective Knight: Independence that I'm aware of. ...

Q1 : Can you recommend some action movie released in 2023 to me?

A1:There are a few upcoming action movies that might interest you:
1.Detective Knight: Independence (also known as Devil's Knight): this movie is a sequel to Detective Knight: Redemption ...
2.Mission Majnu: This Indian spy thriller starring Sidharth Malhotra ...
3.Jung_E: This South Korean science fiction film directed by ...
...

Q2 : Recommend some movies similar to Detective Knight: Independence ?

A2:Here are some other action movies that you might enjoy:
1.John Wick series (2014-2021) ...
2.Taken (2008) ...
3.Die Hard series (1988-2013) ...
...
These movies all have similar elements to Detective Knight: Independence, such as action, suspense, and a hero who must use their skills to overcome obstacles and save the day.

Fail to response to new item
Jung_E
also known as Devil's Knight)...
Mission Majnu
This Indian spy thriller starring ...
...
text-embedding-ada-002
Item Embeddings
Question Embedding
Similarity Calculation
Related Content
Recommendation System R
User – Item History Interaction
User Query Q_i
User profile
History of Dialogue $H_{<i}$
Prompt Constructor C
ChatGPT
Use Database?
Yes
No
Output A_i
Correctly response to new item by using external information

Fig 3: Chart from 'Chat rec towards interactive and explainable llms augmented recommender system'

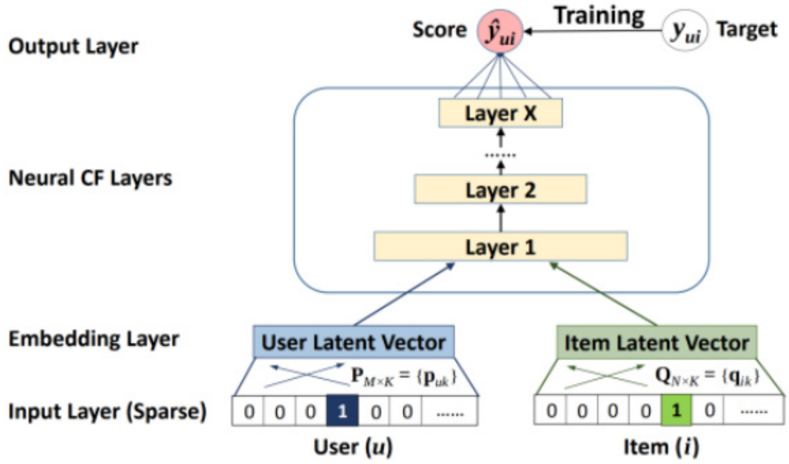


Fig 4: Chart from 'Enhanced recommendation combining collaborative filtering and large language models'

This survey paper focuses on the integration of large language models into recommendation systems, exploring how LLMs are reshaping paradigms in user preference modeling, sequential recommendation, and explainability [6]. The survey begins by presenting a comprehensive taxonomy of LLM-enhanced recommendation frameworks, categorizing them into three tiers: representing and understanding, scheming and utilizing, and industrial deploying [7]. Each tier highlights distinct aspects of LLM integration, from semantic interpretation of user preferences to practical deployment considerations in real-world applications. The paper also delves into versatile frameworks like SUBER for reinforcement learning-based systems and examines the role of generative agents in behavioral modeling, providing a detailed analysis of their methodological contributions and challenges.

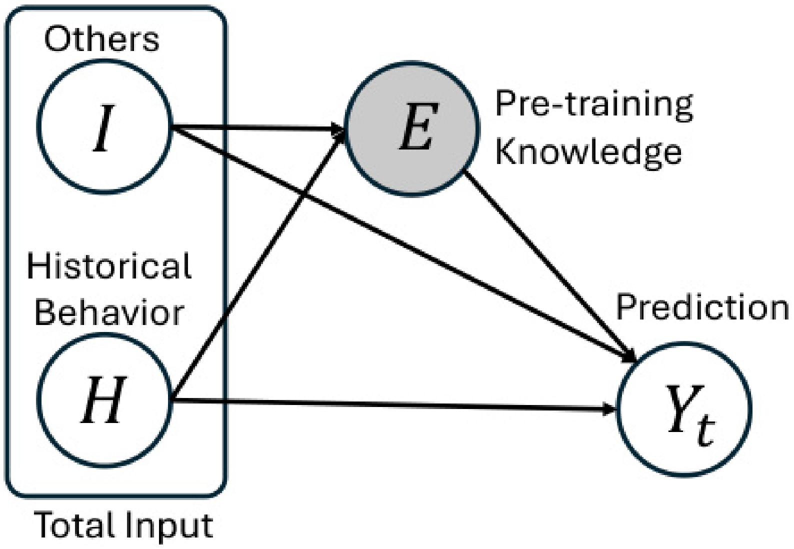


Fig 5: Chart from 'Causality enhanced behavior sequence modeling in llms for personalized recommendation'

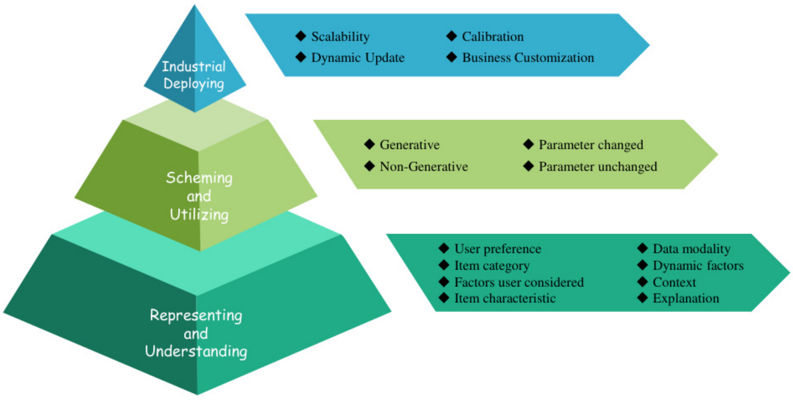


Fig 6: Chart from 'Towards next generation llm based recommender systems a survey and beyond'

The subsequent sections of this survey examine key advancements in user preference representation and personalization, emphasizing the role of natural language generation in creating dynamic user profiles. Hybrid approaches combining explicit and implicit preferences are discussed, alongside strategies for mitigating cold-start and data sparsity issues through LLMs. The survey further explores the critical dimensions of explainability and fairness in recommendation systems, analyzing controllable strategies, item-side bias mitigation, and user-centric explanations. Additionally, the integration of LLMs with graph neural networks and multimodal data processing is addressed, highlighting innovations in token alignment, high-frequency screenshot analysis, and unified latent spaces for text and image modalities.

This survey makes several significant contributions to the field of LLM-based recommendation systems [7]. First, it provides a structured overview of existing methodologies and taxonomies, offering researchers and practitioners a clear framework for understanding the current state of the art. Second, it identifies and analyzes key challenges, such as scalability, interpretability, and computational efficiency, while proposing potential solutions grounded in empirical evidence. Third, the survey synthesizes findings from recent studies, presenting insights into the effectiveness of hybrid collaborative filtering techniques, parameter-efficient approaches, and retrieval-augmented models. By bridging theoretical foundations with practical applications, this work aims to guide future research directions and facilitate the development of more robust and user-centric recommendation systems [7].

3 LLM-Enhanced Recommendation Paradigms

3.1 Frameworks and Taxonomies for LLM Integration

3.1.1 Three-Tier Taxonomy for LLM-Based Recommender Systems

The three-tier taxonomy for LLM-based recommender systems provides a structured framework to categorize and analyze the integration of Large Language Models (LLMs) in recommendation pipelines [6]. The first tier, representing and understanding, focuses on how LLMs interpret and model user preferences and item characteristics. Unlike traditional methods reliant on explicit feedback or sparse interaction data, LLMs leverage their deep semantic understanding to infer latent user interests from textual descriptions, reviews, or contextual cues [8]. This capability allows for richer user profiling and more nuanced item representations, addressing challenges such as cold-start scenarios and sparsity in user-item interactions. By encoding both user and item information into a shared embedding space, LLMs facilitate cross-domain recommendations and enhance the diversity of suggested content [8].

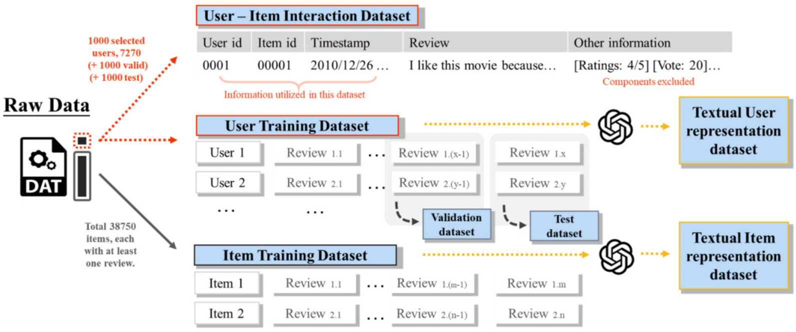


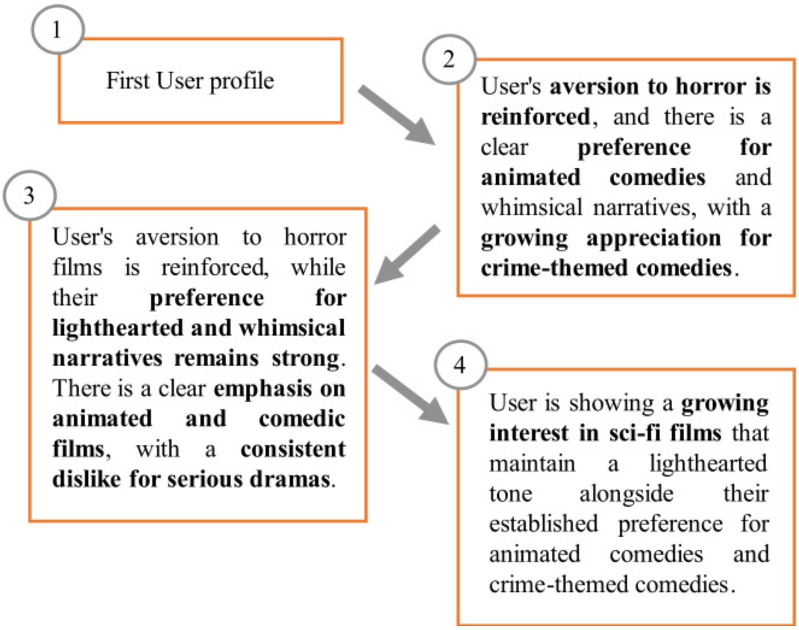
Fig 7: Chart from 'Empowering few shot recommender systems with large language models enhanced representations'

The second tier, scheming and utilizing, delves into the operational mechanisms through which LLMs generate recommendations [9]. This involves leveraging generative capabilities to produce tailored suggestions, explanations, or conversational responses that align with user expectations. For instance, LLMs can dynamically adapt their outputs based on multi-turn dialogues, enabling interactive and explainable recommendation experiences. Additionally, this tier highlights the dual role of LLMs as both decision-makers and interpreters, where they not only predict suitable items but also justify recommendations by synthesizing domain-specific knowledge. Such an approach bridges the gap between discriminative and generative recommendation paradigms, offering a balance between accuracy and interpretability.

The final tier, industrial deploying, examines the practical considerations and challenges of integrating LLMs into real-world recommender systems [10]. While LLMs exhibit remarkable potential, their deployment demands careful attention to scalability, latency, and cost-efficiency. Techniques such as fine-tuning open-source models or employing prompt-based strategies for closed-source LLMs are explored to optimize performance within industrial constraints. Furthermore, this tier emphasizes the importance of evaluation frameworks that assess LLM-based systems across dimensions like personalization, diversity, and user satisfaction. By systematically addressing these aspects, the three-tier taxonomy serves as a comprehensive guide for researchers and practitioners aiming to harness the transformative capabilities of LLMs in recommender systems [7].

3.1.2 Versatile Frameworks like SUBER for RL-Based Systems

The integration of versatile frameworks like SUBER into reinforcement learning (RL)-based systems marks a significant advancement in the field of recommendation systems [11]. These frameworks leverage the power of Large Language Models (LLMs) to enhance decision-making processes, enabling more dynamic and context-aware recommendations [7]. SUBER, as an exemplar of such frameworks, facilitates the efficient utilization of LLMs by providing a structured environment where RL agents can operate. This is achieved through a modular architecture that supports various stages of the recommendation process, from understanding user preferences to executing real-time adjustments based on feedback. By embedding LLMs within this framework, systems can simulate complex user behaviors and interactions, thereby improving the robustness and adaptability of RL models.



1 First User profile

2 User's **aversion to horror is reinforced**, and there is a clear **preference for animated comedies** and whimsical narratives, with a **growing appreciation for crime-themed comedies**.

3 User's aversion to horror films is reinforced, while their **preference for lighthearted and whimsical narratives remains strong**. There is a clear **emphasis on animated and comedic films**, with a **consistent dislike for serious dramas**.

4 User is showing a **growing interest in sci-fi films** that maintain a lighthearted tone alongside their established preference for animated comedies and crime-themed comedies.

Fig 9: Chart from 'Lusifer llm based user simulated feedback environment for online recommender systems'

One of the primary advantages of using frameworks like SUBER is their ability to address the limitations inherent in traditional RL-based systems. Traditional systems often struggle with issues such as slow inference times and the cold-start problem, which hinder their effectiveness in real-world applications. SUBER mitigates these challenges by employing LLMs to generate realistic simulations and synthetic data, allowing for more efficient training and evaluation of RL models. Furthermore, the framework's design enables seamless integration of cross-modal information, enhancing the system's capacity to infer user intent from sparse data. As a result, RL-based recommendation systems powered by SUBER can deliver more personalized and timely suggestions, significantly improving user satisfaction [12].

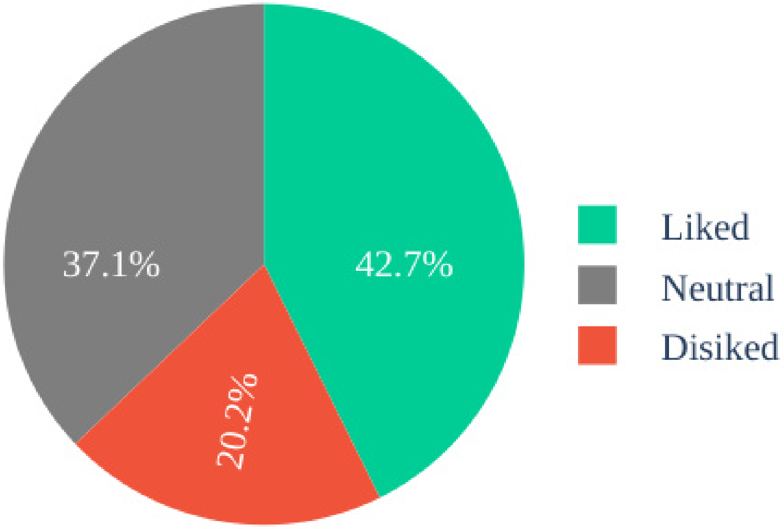


Fig 10: Chart from 'Suber an rl environment with simulated human behavior for recommender systems'

Looking forward, the adoption of versatile frameworks like SUBER is poised to revolutionize RL-based recommendation systems by bridging gaps in scalability, realism, and domain specificity [11]. The flexibility offered by these frameworks allows researchers to experiment with diverse configurations and fine-tune parameters to optimize performance across different scenarios. Moreover, the incorporation of explainable AI components within SUBER ensures that the decision-making processes are transparent and interpretable, fostering trust and acceptance among end-users. As the field continues to evolve, frameworks like SUBER will undoubtedly play a pivotal role in advancing the capabilities of RL-based systems, paving the way for more sophisticated and user-centric recommendation solutions [12].

3.1.3 Generative Agents and Simulators for Behavioral Modeling

Generative agents and simulators have emerged as pivotal tools for behavioral modeling, particularly in the context of recommendation systems [13]. These agents leverage large language models (LLMs) to simulate human-like interactions and decision-making processes, thereby enabling more nuanced user behavior predictions [13]. Unlike traditional models that rely on static datasets, generative agents dynamically adapt to new inputs through techniques such as in-context learning, allowing them to generalize across various tasks without extensive fine-tuning. This adaptability is crucial for capturing the complexity of user preferences and behaviors, which are often influenced by evolving contexts and latent factors.

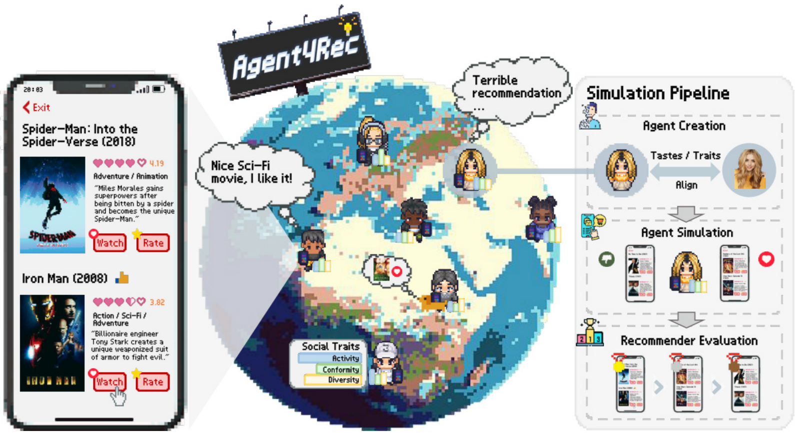

Agent4Rec
20:03
Exit
Spider-Man: Into the Spider-Verse (2018)
4.19
Adventure / Animation
"Miles Morales gains superpowers after being bitten by a spider and becomes the unique Spider-Man."
Watch
Rate
Iron Man (2008)
3.82
Action / Sci-Fi / Adventure
"Billionaire engineer Tony Stark creates a unique weaponized suit of armor to fight evil."
Watch
Rate
Nice Sci-Fi movie, I like it!
Terrible recommendation ...
Social Traits
Activity
Conformity
Diversity
Simulation Pipeline
Agent Creation
Tastes / Traits
Align
Agent Simulation
Recommender Evaluation

Fig 11: Chart from 'On generative agents in recommendation'

The integration of generative agents into recommendation systems introduces several methodological advancements. Firstly, these agents can model high-order dependencies in user interaction sequences, akin to recurrent neural networks (RNNs), but with enhanced semantic understanding due to their foundation in LLMs. Secondly, they facilitate the creation of synthetic yet realistic datasets, which are invaluable for addressing challenges like cold-start problems and data sparsity. By simulating diverse user behaviors, these agents enable the testing and refinement of recommendation algorithms in controlled environments. Furthermore, the modular architecture of generative agents—comprising components like memory, profile management, and interaction handling—allows for scalable and customizable implementations tailored to specific application domains.

Despite their potential, generative agents and simulators face notable challenges, including ensuring realism and domain specificity in simulated behaviors. The auto-regressive nature of LLMs can lead to slower inference times, posing difficulties for real-time applications. Additionally, accurately replicating the intricacies of human reasoning and planning remains an open research question. Nevertheless, ongoing advancements in areas such as causal inference and cross-modal reasoning hold promise for overcoming these limitations. As a result, generative agents are increasingly viewed as a cornerstone for developing adaptive, transparent, and personalized recommendation systems capable of meeting the demands of modern users.

3.2 User Preference Representation and Personalization

3.2.1 Natural Language Generation for User Profiles

Natural Language Generation (NLG) for user profiles represents a transformative approach in recommendation systems, leveraging the advanced capabilities of Large Language Models (LLMs) [14]. These models excel in understanding and generating human-like text, enabling the creation of detailed and dynamic user profiles. By synthesizing user interactions and preferences into coherent textual descriptions, LLMs facilitate a deeper comprehension of user needs [2]. This process not only enhances the granularity of user profiles but also allows for the integration of cross-domain knowledge, enriching the contextual relevance of recommendations. The ability to automatically generate and update these profiles ensures that recommendation systems remain adaptive and personalized, addressing the limitations of static profiling methods.

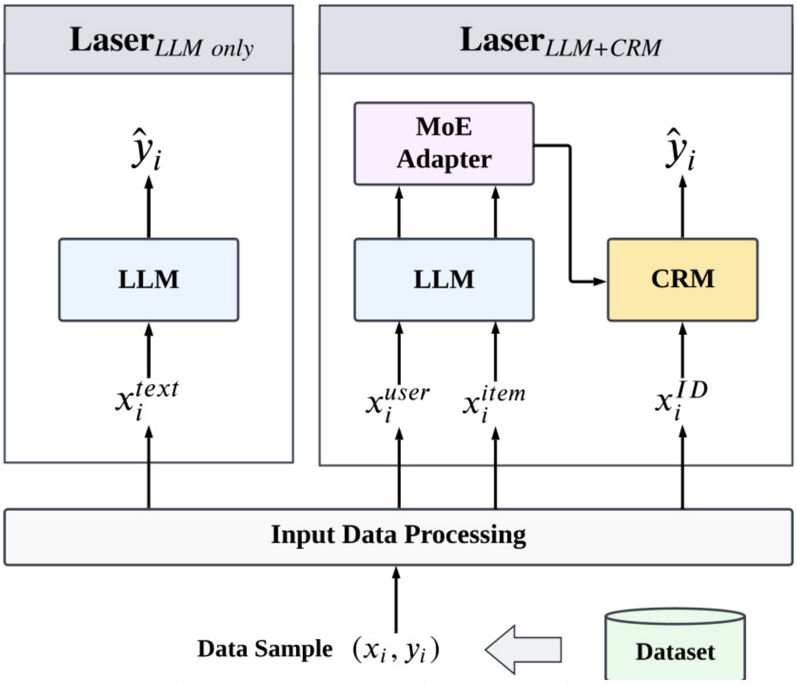


Fig 12: Chart from 'Large language models make sample efficient recommender systems'

Despite the potential of LLMs in NLG for user profiles, several challenges persist in evaluating the quality and effectiveness of generated descriptions [15]. Current research primarily focuses on the technical performance of LLMs, often overlooking the impact of generated profiles on recommendation accuracy and diversity [16]. The interpretability of these profiles remains underexplored, with limited studies assessing how well they capture nuanced user preferences and translate them into actionable recommendations. Moreover, the dynamic nature of user interests necessitates continuous updates to profiles, posing additional complexities in maintaining consistency and relevance over time. Addressing these challenges requires a multidisciplinary approach, combining insights from machine learning, linguistics, and human-computer interaction.

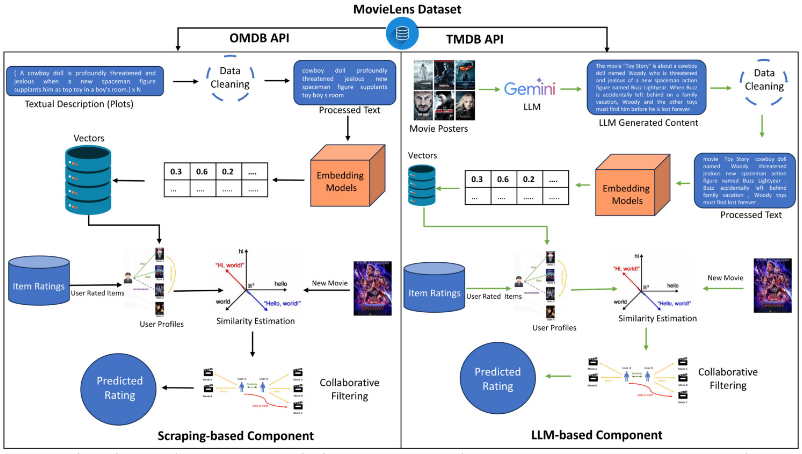

MovieLens Dataset
OMDB API
TMDB API
(A cowboy doll is profoundly threatened and jealous when a new spaceman figure supplants him as top toy in a boy's room.) x N
Textual Description (Plots)
Data Cleaning
cowboy doll profoundly threatened jealous new spaceman figure supplants toy boy s room
Processed Text
Vectors
0.3 0.6 0.2 ....
Embedding Models
Item Ratings
User Rated Items
User Profiles
hi
"Hi, world!"
hello
world
"Hello, world!"
Similarity Estimation
New Movie
Predicted Rating
Collaborative Filtering
Scraping-based Component
Movie Posters
Gemini
LLM
The movie "Toy Story" is about a cowboy doll named Woody who is threatened and jealous of a new spaceman action figure named Buzz Lightyear. When Buzz is accidentally left behind on a family vacation, Woody and the other toys must find him before he is lost forever.
LLM Generated Content
Data Cleaning
movie Toy Story cowboy doll named Woody threatened jealous new spaceman action figure named Buzz Lightyear . Buzz accidentally left behind family vacation , Woody toys must find lost forever .
Processed Text
Vectors
0.3 0.6 0.2 ....
Embedding Models
Item Ratings
User Rated Items
User Profiles
Similarity Estimation
New Movie
Predicted Rating
Collaborative Filtering
LLM-based Component

Fig 13: Chart from 'Llm based cross modality retrieval to improve recommendation performance'

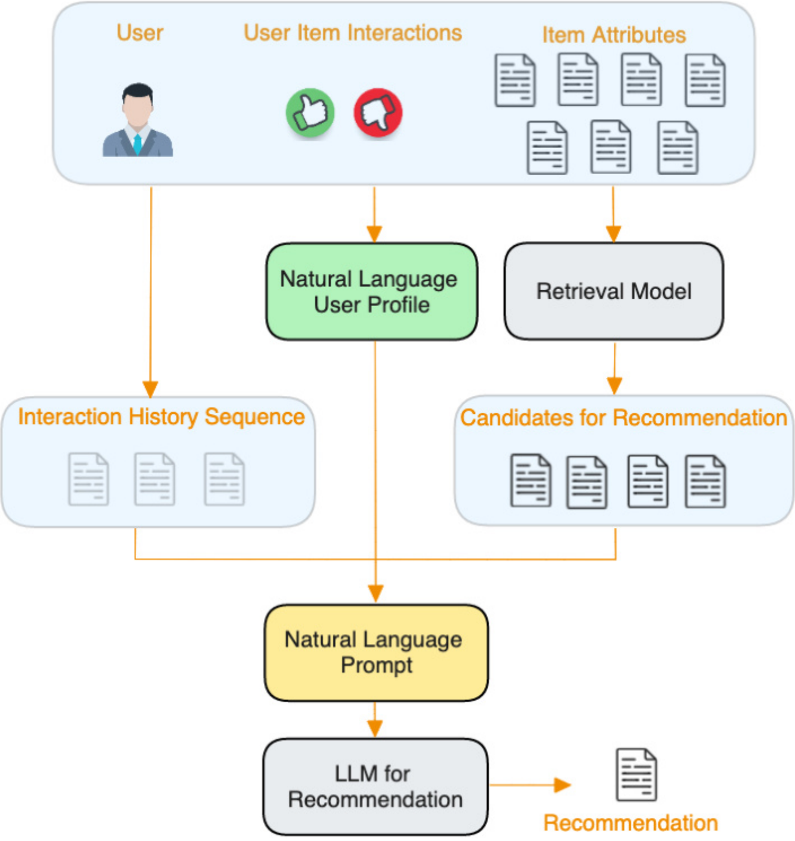


Fig 14: Chart from 'Palr personalization aware llms for recommendation'

The integration of NLG into user profiling also opens new avenues for innovation in recommendation systems. For instance, incorporating Retrieval-Augmented Generation (RAG) can enhance the contextual accuracy of profiles by blending retrieved information with generative capabilities. This hybrid approach ensures that recommendations are not only personalized but also informed by the latest data. Additionally, the interpretability of recommendations can be significantly improved through NLG, as LLMs can provide explanations based on generated profiles, fostering user trust and engagement [15]. As research progresses, exploring the synergy between NLG and other emerging technologies will be crucial in advancing the field, paving the way for more sophisticated and user-centric recommendation paradigms.

3.2.2 Hybrid Approaches Combining Explicit and Implicit Preferences

Hybrid approaches in recommendation systems aim to leverage the strengths of both explicit and implicit user preferences, addressing the limitations inherent in using either method alone. Explicit preferences are typically gathered through direct user input, such as ratings or reviews, offering clear insights into user likes and dislikes. However, obtaining explicit data can be challenging due to user effort and potential biases. Implicit preferences, on the other hand, are derived from user behaviors like clicks, purchases, or browsing history, providing a wealth of data with minimal user intervention but often lacking clarity in user intent. By combining these two types of data, hybrid systems can enhance recommendation accuracy and relevance, ensuring a more comprehensive understanding of user preferences.

The integration of explicit and implicit data within hybrid recommendation systems is achieved through various methodologies. One common approach involves weighting the influence of each data type based on its reliability and relevance to the current recommendation context. For instance, explicit data might be prioritized when available due to its clarity, while implicit data fills in gaps where explicit information is sparse or absent. Advanced techniques may employ machine learning models that dynamically adjust the balance between explicit and implicit inputs, optimizing for performance metrics such as precision, recall, and user satisfaction. These models often utilize sophisticated algorithms capable of discerning patterns and correlations across diverse data sets, thus improving the system's adaptability and robustness.

Despite their advantages, hybrid approaches face challenges, particularly in balancing the trade-offs between data quality and quantity. Explicit data, while valuable, is often limited and may not fully represent a user’s preferences, especially in scenarios with low engagement. Conversely, implicit data, though abundant, can introduce noise and ambiguity, requiring careful preprocessing and interpretation. Additionally, the computational complexity of integrating and processing dual data streams necessitates efficient algorithms and scalable architectures. Addressing these challenges is crucial for realizing the full potential of hybrid recommendation systems, enabling them to deliver personalized, accurate, and timely suggestions that enhance user experience across various domains [2].

3.2.3 Addressing Cold Start and Data Sparsity Challenges

The cold-start problem remains a persistent challenge in recommendation systems, particularly when historical interaction data is scarce or nonexistent. This issue is especially pronounced in domains like news recommendation, where the timeliness and constant influx of new content exacerbate the difficulty of modeling user preferences accurately. Traditional collaborative filtering methods falter under such conditions, as they rely heavily on past interactions to infer recommendations [3]. Meanwhile, content-based filtering, though less dependent on interaction history, struggles with data sparsity and often fails to capture nuanced user interests. Recent advancements in Large Language Models (LLMs) offer promising avenues for mitigating these challenges [4]. By leveraging their extensive pre-training on diverse datasets, LLMs can generalize to unseen items and infer user preferences even with limited interaction data, thus addressing both cold-start and sparsity issues [16].

One of the key strengths of LLMs lies in their ability to perform zero-shot and few-shot learning, enabling them to generate relevant recommendations without requiring extensive prior user-item interactions [17]. For instance, LLMs can utilize textual descriptions of items or users to derive latent representations that inform recommendations, bypassing the need for dense interaction histories [18]. Moreover, their capacity for reasoning and contextual understanding allows these models to dynamically adapt to evolving user interests, a critical feature for domains like news recommendation where user preferences may shift rapidly [19]. However, despite these advantages, the computational demands of deploying LLMs in real-time systems remain a significant barrier. The auto-regressive nature of LLMs often results in slower inference times, which can hinder their applicability in scenarios requiring rapid response.

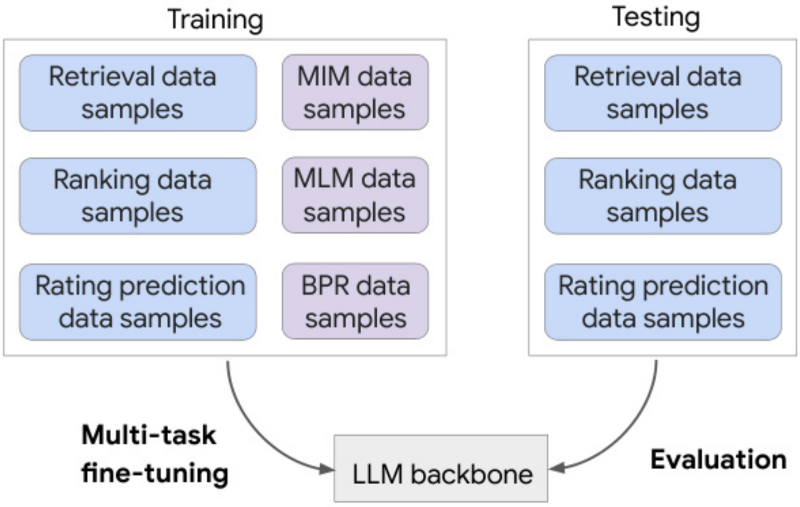


Fig 15: Chart from 'Aligning large language models with recommendation knowledge'

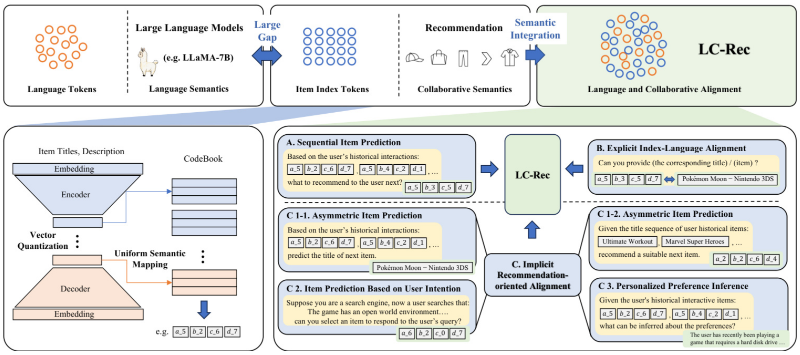

Language Tokens
Large Language Models
(e.g. LLaMA-7B)
Language Semantics
Large Gap
Item Index Tokens
Recommendation
Collaborative Semantics
Semantic Integration
LC-Rec
Language and Collaborative Alignment
Item Titles, Description
Embedding
Encoder
CodeBook
Vector Quantization
Uniform Semantic Mapping
Decoder
Embedding
e.g. a_5 b_2 c_6 d_7
A. Sequential Item Prediction
Based on the user's historical interactions:
a_5 b_2 c_6 d_7 , a_5 b_4 c_2 d_1 , …
what to recommend to the user next?
a_5 b_3 c_5 d_7
LC-Rec
B. Explicit Index-Language Alignment
Can you provide (the corresponding title) / (item) ?
a_5 b_3 c_5 d_7
Pokémon Moon – Nintendo 3DS
C 1-1. Asymmetric Item Prediction
Based on the user's historical interactions:
a_5 b_2 c_6 d_7 , a_5 b_4 c_2 d_1 , …
predict the title of next item.
Pokémon Moon – Nintendo 3DS
C. Implicit Recommendation-oriented Alignment
C 1-2. Asymmetric Item Prediction
Given the title sequence of user historical items:
Ultimate Workout , Marvel Super Heroes , …
recommend a suitable next item.
a_2 b_2 c_6 d_4
C 2. Item Prediction Based on User Intention
Suppose you are a search engine, now a user searches that:
The game has an open world environment….
can you select an item to respond to the user's query?
a_6 b_2 c_0 d_7
C 3. Personalized Preference Inference
Given the user's historical interactive items:
a_5 b_2 c_6 d_7 , a_5 b_4 c_2 d_1 , …
what can be inferred about the preferences?
The user has recently been playing a game that requires a hard disk drive …

Fig 16: Chart from 'Adapting large language models by integrating collaborative semantics for recommendation'

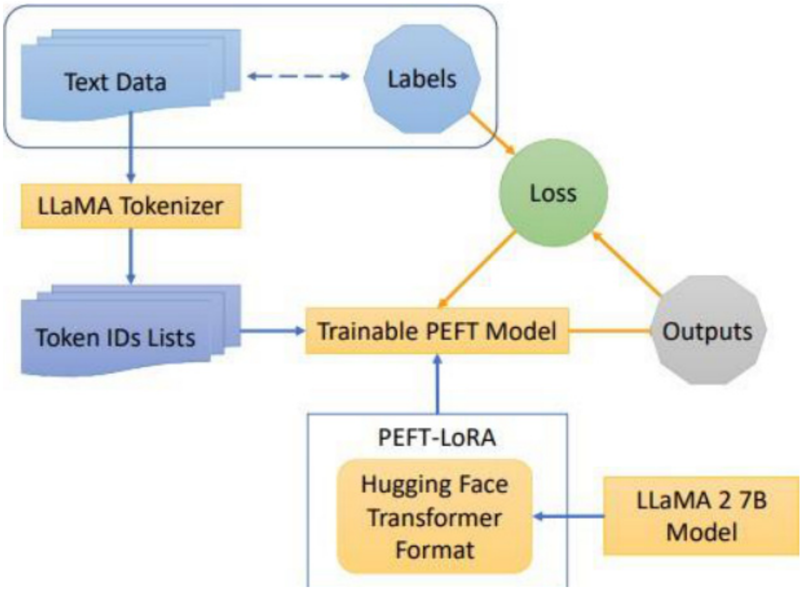


Fig 17: Chart from 'Product recommendation system using large language model llama 2'

To address these limitations, hybrid approaches are increasingly being explored, combining the strengths of LLMs with domain-specific techniques. For example, integrating knowledge graphs with LLMs can bridge the modality gap between structured data and natural language processing, enhancing the model's interpretability and recommendation accuracy [17]. Additionally, strategies such as fine-tuning LLMs on domain-specific corpora or employing prompt-based methods can reduce the computational overhead while maintaining performance. These innovations highlight the potential of LLMs to transform recommendation systems by alleviating cold-start and data sparsity challenges, provided that scalability and efficiency concerns are adequately addressed [4].

3.3 Explainability and Fairness in Recommendation Systems

3.3.1 Controllable and Transparent Recommendation Strategies

Controllable and transparent recommendation strategies are becoming increasingly vital in the design of modern recommender systems. As these systems grow more complex, ensuring that users can comprehend and influence the recommendations they receive is essential for fostering trust and satisfaction. Control in recommendation systems allows users to adjust parameters or directly input preferences, thus tailoring outputs to better match their needs. This contrasts with traditional systems where recommendations are generated based solely on historical data, often leaving users puzzled by the suggestions. Transparency complements controllability by elucidating how recommendations are derived, offering insights into the decision-making process of the algorithm.

The implementation of controllable strategies typically involves integrating interactive elements within the recommendation interface. Users may be provided with sliders, filters, or direct feedback mechanisms to refine results actively. For instance, a user could specify that they prefer items from certain categories or exclude others entirely, guiding the system towards more relevant suggestions. Meanwhile, transparency can be achieved through explainable AI techniques, where the system generates human-readable explanations for each recommendation. These might include highlighting key factors considered by the algorithm or visualizing data patterns leading to specific suggestions, thereby demystifying the underlying processes.

Despite their benefits, developing controllable and transparent recommendation systems presents several challenges. Balancing control with automation requires careful design to avoid overwhelming users with options while maintaining ease of use. Similarly, achieving transparency without compromising performance or exposing proprietary algorithms demands innovative solutions. Moreover, as datasets expand and models grow more intricate, ensuring real-time interactivity and clear explanations becomes computationally demanding. Addressing these issues necessitates ongoing research into efficient algorithms and user-centric design principles, paving the way for future advancements in this domain.

3.3.2 Mitigating Item-Side Bias in LLM-Based Systems

Mitigating item-side bias in LLM-based systems is crucial for ensuring equitable and accurate recommendations. Item-side bias arises when certain items are disproportionately favored or neglected due to inherent characteristics of the model or skewed training data. This can lead to suboptimal user experiences and undermine the perceived fairness of recommendation systems. Addressing this issue requires a careful examination of how LLMs process and prioritize information about items. Techniques such as reweighting, adversarial training, and debiasing through regularization have been explored to counteract these biases. These methods aim to recalibrate the influence of specific features within the model, thereby promoting a more balanced representation of all items.

One promising approach involves leveraging domain-specific knowledge to enhance the understanding of item attributes. By integrating structured data sources like Knowledge Graphs (KGs), LLMs can better contextualize items and mitigate biases stemming from incomplete or unrepresentative training datasets [20]. KGs provide a rich framework that captures relationships between items, enabling the model to discern nuanced distinctions and reduce reliance on superficial patterns. Additionally, fine-tuning LLMs with diverse datasets that encompass a wide array of item types and user interactions can further diminish item-side bias. Such strategies not only improve the accuracy of recommendations but also foster a more inclusive environment where less popular items receive fair consideration [21].

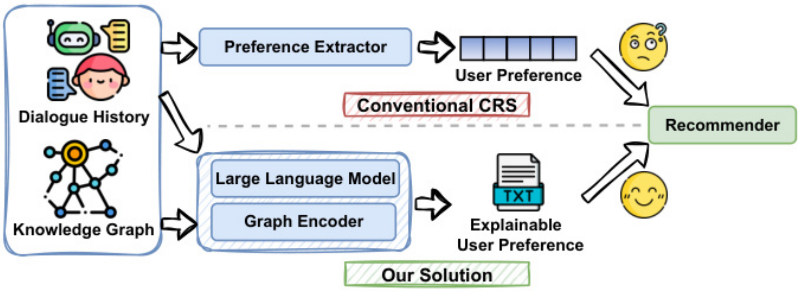
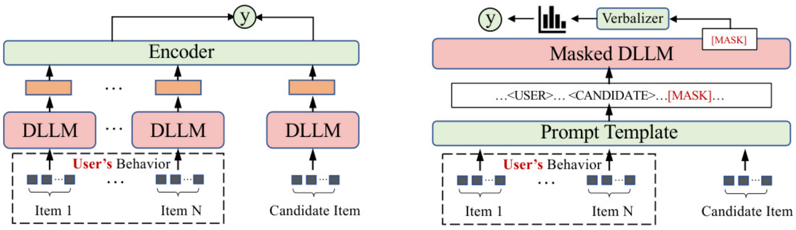


Fig 18: Chart from 'Unveiling user preferences a knowledge graph and llm driven approach for conversational recommendation'



(a) Finetuning DLLM for recommendation (b) Prompt tuning DLLM for recommendation

Fig 19: Chart from 'A survey on large language models for recommendation'

Despite these advancements, challenges remain in fully eradicating item-side bias within LLM-based systems. The dynamic nature of item catalogs, coupled with evolving user preferences, necessitates continuous adaptation and refinement of mitigation techniques. Moreover, the balance between personalization and fairness must be carefully managed to avoid overcorrecting biases, which could inadvertently distort recommendation outcomes. Future research should focus on developing adaptive mechanisms that dynamically adjust to changes in both item availability and user behavior. By doing so, LLM-based systems can achieve a harmonious blend of precision and equity, ultimately enhancing the overall effectiveness and trustworthiness of recommendation engines [16].

3.3.3 Enhancing User-Centric Explanations

Enhancing user-centric explanations in recommendation systems has become a pivotal area of research, driven by the need for transparency and trust in AI-driven decisions. Large Language Models (LLMs) have shown significant potential in this domain by generating natural language explanations that elucidate why certain recommendations are made [22]. These models leverage their extensive training on diverse textual data to produce contextually relevant explanations that align with user preferences and behaviors [23]. By integrating LLMs into recommendation frameworks, systems can offer insights into the decision-making process, thereby enhancing user understanding and acceptance of recommendations [24]. This capability not only improves user satisfaction but also fosters a more interactive engagement with the system.

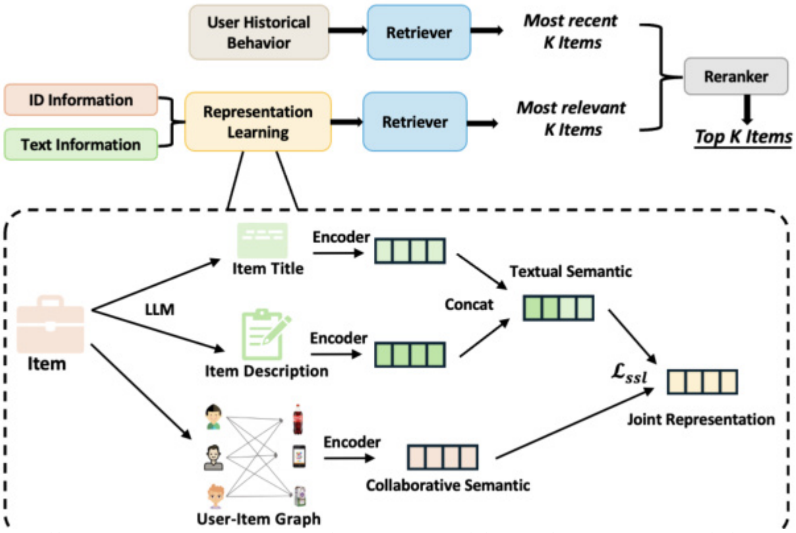


Fig 20: Chart from 'Rallrec improving retrieval augmented large language model recommendation with representation learning'

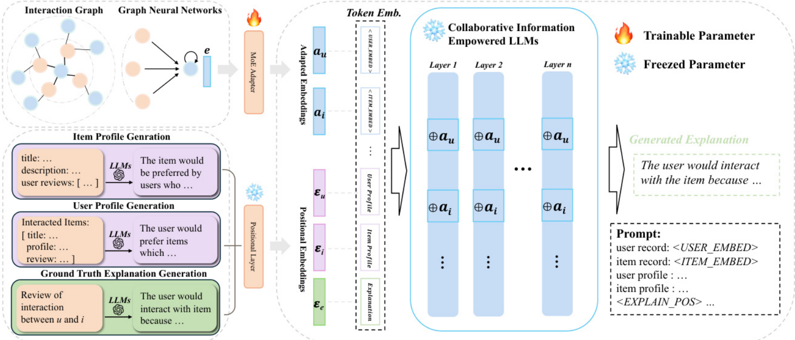
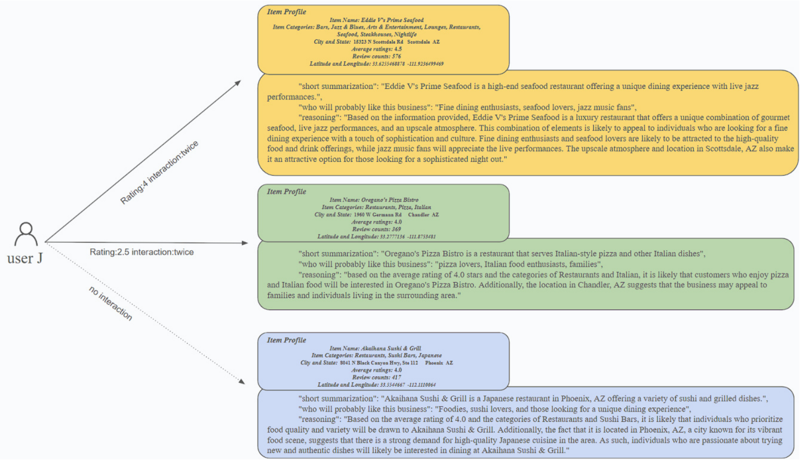


Fig 21: Chart from 'Xrec large language models for explainable recommendation'



Item Profile

Item Name: Eddie V's Prime Seafood
Item Categories: Bars, Jazz & Blues, Arts & Entertainment, Lounges, Restaurants, Seafood, Steakhouses, Nightlife
City and State: 15323 N Scottsdale Rd Scottsdale AZ
Average ratings: 4.5
Review counts: 576
Latitude and Longitude: 33.6255468878 -111.9236499469

"short summarization": "Eddie V's Prime Seafood is a high-end seafood restaurant offering a unique dining experience with live jazz performances.",
"who will probably like this business": "Fine dining enthusiasts, seafood lovers, jazz music fans",
"reasoning": "Based on the information provided, Eddie V's Prime Seafood is a luxury restaurant that offers a unique combination of gourmet seafood, live jazz performances, and an upscale atmosphere. This combination of elements is likely to appeal to individuals who are looking for a fine dining experience with a touch of sophistication and culture. Fine dining enthusiasts and seafood lovers are likely to be attracted to the high-quality food and drink offerings, while jazz music fans will appreciate the live performances. The upscale atmosphere and location in Scottsdale, AZ also make it an attractive option for those looking for a sophisticated night out."

Item Profile

Item Name: Oregano's Pizza Bistro
Item Categories: Restaurants, Pizza, Italian
City and State: 1960 W Germann Rd Chandler AZ
Average ratings: 4.0
Review counts: 369
Latitude and Longitude: 33.2777136 -111.8753481

"short summarization": "Oregano's Pizza Bistro is a restaurant that serves Italian-style pizza and other Italian dishes",
"who will probably like this business": "pizza lovers, Italian food enthusiasts, families",
"reasoning": "based on the average rating of 4.0 stars and the categories of Restaurants and Italian, it is likely that customers who enjoy pizza and Italian food will be interested in Oregano's Pizza Bistro. Additionally, the location in Chandler, AZ suggests that the business may appeal to families and individuals living in the surrounding area."

Item Profile

Item Name: Akaihana Sushi & Grill
Item Categories: Restaurants, Sushi Bars, Japanese
City and State: 8041 N Black Canyon Hwy, Ste 112 Phoenix AZ
Average ratings: 4.0
Review counts: 417
Latitude and Longitude: 33.5544667 -112.1110064

"short summarization": "Akaihana Sushi & Grill is a Japanese restaurant in Phoenix, AZ offering a variety of sushi and grilled dishes.",
"who will probably like this business": "Foodies, sushi lovers, and those looking for a unique dining experience",
"reasoning": "Based on the average rating of 4.0 and the categories of Restaurants and Sushi Bars, it is likely that individuals who prioritize food quality and variety will be drawn to Akaihana Sushi & Grill. Additionally, the fact that it is located in Phoenix, AZ, a city known for its vibrant food scene, suggests that there is a strong demand for high-quality Japanese cuisine in the area. As such, individuals who are passionate about trying new and authentic dishes will likely be interested in dining at Akaihana Sushi & Grill."

Fig 22: Chart from 'A prompting based representation learning method for recommendation with large language models'

Despite these advancements, challenges remain in ensuring that explanations are both accurate and meaningful to users. Current methods often struggle with capturing the dynamic nature of user preferences, which can evolve over time due to various contextual factors. To address this, recent approaches have focused on developing adaptive mechanisms that update user profiles and corresponding explanations in real-time. Techniques such as multi-turn conversational interactions allow systems to refine their understanding of user intent continuously. By employing LLMs to generate explanations during these interactions, systems can provide timely updates that reflect changes in user interests, thus maintaining relevance and enhancing the overall user experience [16].

Furthermore, the integration of LLMs into recommendation systems introduces opportunities for creating more personalized and engaging user experiences [16]. By synthesizing user preferences from ongoing conversations, LLMs can generate tailored explanations that resonate with individual users [16]. This personalization is achieved through a combination of explicit keywords and implicit embeddings that capture nuanced aspects of user preferences [25]. As a result, recommendation systems can deliver explanations that are not only informative but also aligned with the user's cognitive framework. Such user-centric approaches hold the promise of transforming recommendation systems into trusted advisors capable of fostering long-term user engagement and loyalty.

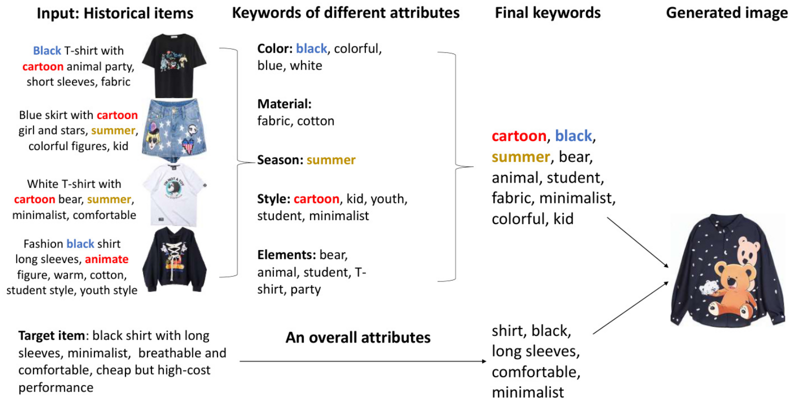

Input: Historical items
Keywords of different attributes
Final keywords
Generated image
Black T-shirt with cartoon animal party, short sleeves, fabric
Blue skirt with cartoon girl and stars, summer, colorful figures, kid
White T-shirt with cartoon bear, summer, minimalist, comfortable
Fashion black shirt long sleeves, animate figure, warm, cotton, student style, youth style
Color: black, colorful, blue, white
Material: fabric, cotton
Season: summer
Style: cartoon, kid, youth, student, minimalist
Elements: bear, animal, student, T-shirt, party
cartoon, black, summer, bear, animal, student, fabric, minimalist, colorful, kid
Target item: black shirt with long sleeves, minimalist, breathable and comfortable, cheap but high-cost performance
An overall attributes
shirt, black, long sleeves, comfortable, minimalist

Fig 23: Chart from 'Pmg personalized multimodal generation with large language models'

4 Integration of LLMs with Graph and Multimodal Data

4.1 Graph Neural Networks and Knowledge Graphs

4.1.1 Token Alignment Between Graph Nodes and LLM Tokens

The alignment between graph nodes and Large Language Model (LLM) tokens represents a pivotal challenge in integrating these two modalities for recommendation systems [26]. Graph-based models typically operate on discrete node identifiers that encapsulate relational structures, while LLMs process continuous token embeddings derived from textual data. This fundamental disparity necessitates a robust alignment mechanism to bridge the semantic gap between structural graph data and linguistic token representations. Effective token alignment ensures that the rich relational information encoded within graph nodes is accurately translated into a format amenable to LLM processing, thus enabling the model to leverage both structural and textual cues for enhanced recommendation accuracy [26].

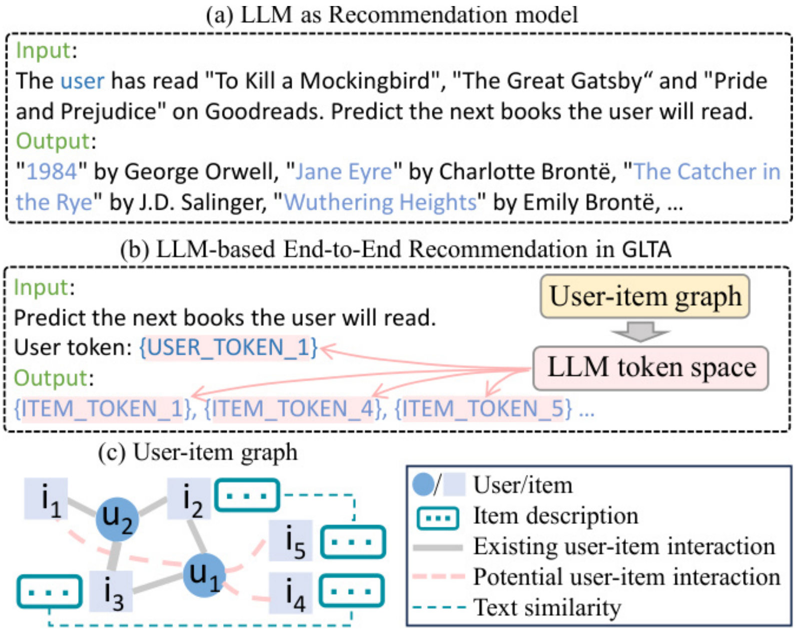

(a) LLM as Recommendation model
Input:
The user has read "To Kill a Mockingbird", "The Great Gatsby" and "Pride and Prejudice" on Goodreads. Predict the next books the user will read.
Output:
"1984" by George Orwell, "Jane Eyre" by Charlotte Brontë, "The Catcher in the Rye" by J.D. Salinger, "Wuthering Heights" by Emily Brontë, ...
(b) LLM-based End-to-End Recommendation in GLTA
Input:
Predict the next books the user will read.
User token: {USER_TOKEN_1}
Output:
{ITEM_TOKEN_1}, {ITEM_TOKEN_4}, {ITEM_TOKEN_5} ...
User-item graph
LLM token space
(c) User-item graph
i1
u2
i2
i5
u1
i3
i4
User/item
Item description
Existing user-item interaction
Potential user-item interaction
Text similarity

Fig 24: Chart from 'Training large recommendation models via graph language token alignment'

To address this challenge, recent approaches have proposed various strategies for aligning graph nodes with LLM tokens. One prominent method involves mapping graph nodes to their corresponding textual descriptions, thereby transforming structural identifiers into semantically meaningful tokens. This process typically employs pre-trained language models to generate contextualized embeddings for each node based on its associated text. Additionally, some frameworks introduce hybrid alignment techniques that integrate both explicit textual information and implicit structural embeddings. By combining these diverse representations, such methods aim to mitigate issues related to semantic misalignment and improve the overall coherence of the integrated model.

Despite these advancements, achieving optimal token alignment remains fraught with complexities. The primary difficulty lies in preserving the nuanced relationships captured by graph structures while translating them into the linear sequence format required by LLMs. Furthermore, scalability concerns arise when dealing with large-scale graphs containing millions of nodes, as the computational overhead of aligning each node can become prohibitive. Future research directions may focus on developing more efficient alignment paradigms that maintain high fidelity to both graph topology and linguistic semantics, potentially through innovative graph embedding techniques or hierarchical tokenization strategies tailored specifically for this dual-modality integration.

4.1.2 Reinforcement Learning for Sequential Recommendations

Reinforcement Learning (RL) has emerged as a powerful paradigm for addressing sequential recommendation challenges, where the goal is to predict a sequence of items that align with a user's evolving preferences [11]. Unlike traditional recommendation approaches that rely on static historical data, RL-based methods dynamically adapt to user interactions over time [12]. In sequential recommendation tasks, each prediction involves forecasting the next item in a sequence, which requires modeling both immediate and long-term user satisfaction [27]. RL frameworks achieve this by optimizing a policy that maximizes cumulative rewards, where rewards are typically defined based on user engagement metrics such as clicks, purchases, or dwell time. This approach enables the system to balance exploration—discovering new user interests—and exploitation—recommending items aligned with known preferences.

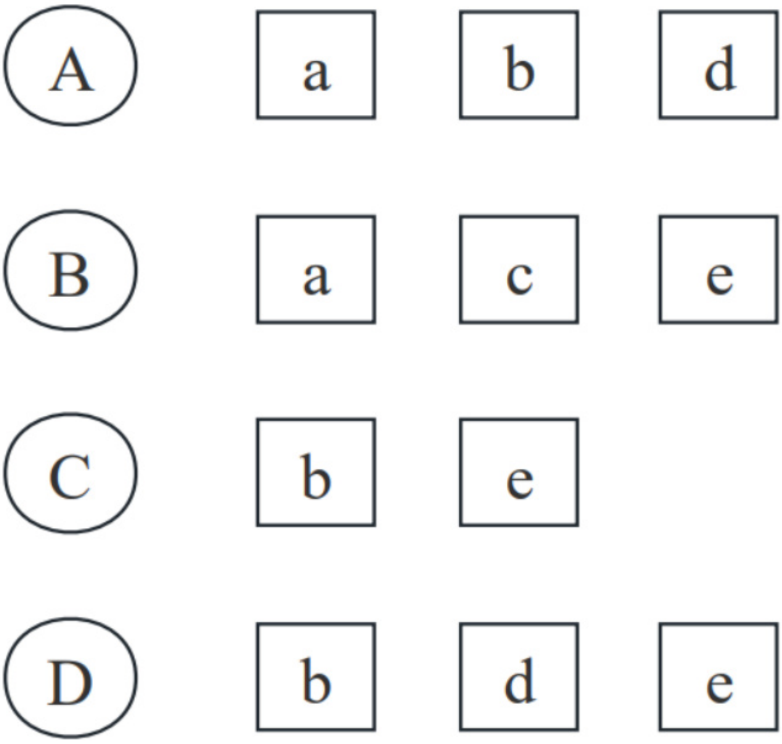


Fig 25: Chart from 'Emerging synergies between large language models and machine learning in ecommerce recommendations'

A key advantage of RL in sequential recommendations lies in its ability to handle uncertainty and evolving dynamics within user behavior. As more tokens or interaction prefixes are established in the sequence, the model's confidence in predicting subsequent items increases, reducing uncertainty. However, uniformly applying RL across all tokens may limit effectiveness, as early predictions often require broader exploration compared to later stages where user intent becomes clearer. To address this, advanced RL algorithms incorporate adaptive mechanisms that adjust the exploration-exploitation tradeoff based on the context. For instance, entropy regularization techniques can be employed to encourage diversity in early predictions while gradually focusing on high-confidence recommendations as the sequence progresses. Such strategies ensure that the recommendation system remains responsive to shifts in user preferences while maintaining coherence in the suggested item sequence.

Despite its potential, integrating RL into sequential recommendation systems presents several technical challenges [12]. One major issue is the sparsity of reward signals, as meaningful feedback from users is often delayed or infrequent. To mitigate this, researchers have explored auxiliary objectives, such as incorporating supervised learning signals derived from historical data, to stabilize training. Additionally, the computational complexity of RL algorithms necessitates efficient architectures, such as graph-based models or transformer variants, to scale effectively across large item catalogs and user bases. By combining RL with these advancements, sequential recommendation systems can achieve superior personalization and adaptability, ultimately enhancing user satisfaction in dynamic online environments [27].

4.1.3 Addressing Scalability and Cold Start Problems

Scalability and cold-start problems remain pivotal challenges in the realm of recommendation systems (RS), particularly when handling vast datasets with sparse interactions. Traditional models often depend heavily on extensive user feedback and exposed samples, which inherently limits their efficacy in scenarios where new items continuously emerge or user interaction data is scarce. This dependency not only hampers the system's ability to scale efficiently but also complicates the integration of new items into the recommendation pool without substantial historical data. Consequently, addressing these limitations necessitates innovative strategies that can operate effectively under constrained information environments while maintaining performance as the system scales.

To mitigate these issues, recent advancements have explored leveraging large language models (LLMs) and other hybrid methodologies to enhance sample efficiency and reduce reliance on extensive historical data. For instance, frameworks like Laser have demonstrated the potential to achieve competitive performance using only a fraction of traditional training samples. By employing few-shot or zero-shot learning techniques, these approaches aim to infer user preferences and item characteristics more adeptly from limited data. Moreover, incorporating external knowledge bases or auxiliary information has shown promise in enriching the contextual understanding of both users and items, thereby improving recommendations even when interaction data is minimal.

Despite these strides, significant hurdles remain, especially concerning the optimal utilization of sequential behavior data and ensuring real-time adaptability. As depicted in various studies, merely integrating historical sequences does not necessarily translate to improved outcomes unless the underlying mechanisms are refined to extract meaningful patterns effectively. Thus, future research should focus on developing more interpretable and dynamic models capable of real-time updates and better alignment with evolving user behaviors and item landscapes. Such enhancements could significantly bolster the scalability and robustness of recommendation systems against cold-start and sparsity challenges.

4.2 Multimodal Data Processing and Fusion

4.2.1 Unified Latent Space for Text and Image Modalities

The unification of latent spaces for text and image modalities represents a pivotal advancement in multimodal learning, particularly within the context of recommendation systems (RS) [28]. By integrating large language models (LLMs) with vision models, this approach facilitates the transformation of diverse data types into a shared embedding space. Such a unified latent space simplifies the complexity inherent in processing heterogeneous data, allowing the model to learn richer cross-modal relationships. The embeddings derived from pre-trained vision models encapsulate visual features, while LLMs contribute contextual understanding, enabling a seamless fusion of textual and visual information. This synergy not only enhances the discriminative power of RS but also addresses challenges such as sparsity and cold-start problems by leveraging complementary modalities.

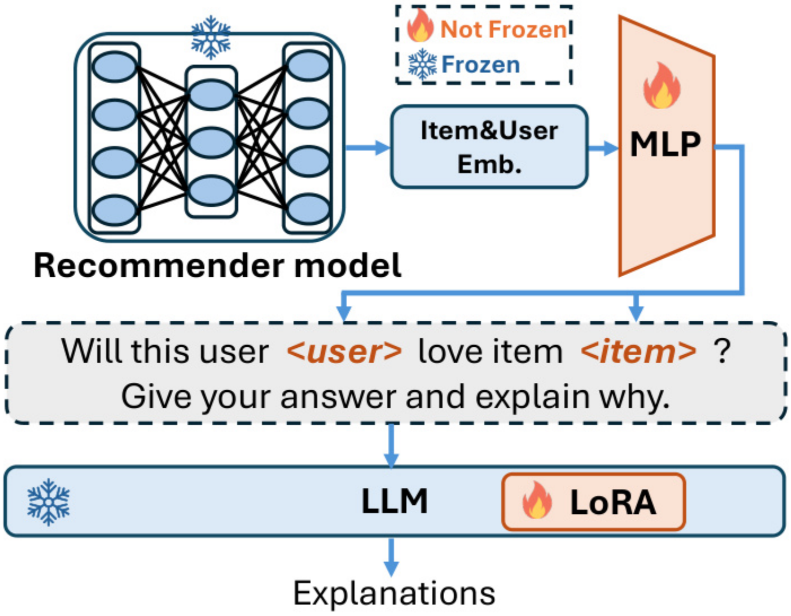

Not Frozen
Frozen
Recommender model
Item&User
Emb.
MLP
Will this user <user> love item <item> ?
Give your answer and explain why.
LLM
LoRA
Explanations

Fig 26: Chart from 'Recexplainer aligning large language models for explaining recommendation models'

A key challenge in constructing a unified latent space lies in aligning the distinct representational formats of text and images. Vision models typically operate on pixel data, generating dense embeddings that capture spatial and semantic features, whereas LLMs process discrete tokens, producing sparse yet highly contextualized representations. Bridging these modalities requires innovative alignment mechanisms, such as token projectors or cross-modal adapters, which map visual embeddings into a format congruent with LLMs. These techniques ensure that the latent space preserves both the semantic richness of language and the descriptive granularity of images. Furthermore, the alignment process is often supported by auxiliary tasks designed to fine-tune the shared space, enhancing its ability to discern subtle inter-modal correlations and improving downstream performance metrics.

The practical implications of a unified latent space extend beyond traditional recommendation paradigms, enabling more personalized and context-aware suggestions. For instance, by incorporating multi-modal data such as product images and user reviews, RS can generate recommendations that are not only statistically accurate but also semantically meaningful. This enriched understanding allows the system to better interpret user intent and preferences, even in scenarios with limited interaction data. Additionally, the unified framework supports advanced applications like open-world image understanding and meme interpretation, where reasoning across modalities is crucial. As research progresses, refining the integration of text and image embeddings within a cohesive latent space will remain a cornerstone for advancing multimodal recommendation systems [28].

4.2.2 High-Frequency Screenshot Analysis for Real-Time Insights

High-frequency screenshot analysis represents a pivotal advancement in extracting real-time insights from user interactions within digital environments. By capturing screenshots at millisecond intervals, this technique enables the observation of nuanced behavioral patterns that are often missed by traditional logging methods. The near-instantaneous capture of visual data provides a granular view of user actions, allowing systems to detect subtle cues such as cursor hesitations, rapid navigation changes, or micro-interactions with interface elements. This capability is particularly beneficial for applications requiring immediate feedback, such as adaptive user interfaces or real-time recommendation engines. The integration of high-frequency screenshot analysis into behavior sequence modeling enhances the precision of predictions and enriches the contextual understanding of user intent.

The effectiveness of high-frequency screenshot analysis is amplified when combined with multi-modal data processing frameworks. Screenshots inherently contain rich visual information that, when processed alongside other modalities like text or interaction logs, creates a unified latent space for more robust model training. For instance, visual features extracted from screenshots can be aligned with embeddings derived from historical exposure and click logs to refine the predictive power of recommendation systems. This approach not only addresses challenges posed by data sparsity but also mitigates the limitations of relying solely on textual or numerical inputs. By leveraging the complementary strengths of different data types, high-frequency screenshot analysis contributes to a more comprehensive representation of user behavior, thereby improving the discriminative capabilities of machine learning models.

Despite its advantages, the implementation of high-frequency screenshot analysis poses technical challenges, particularly in terms of computational efficiency and scalability. Processing and analyzing large volumes of visual data in real time demand significant computational resources, which can be prohibitive in resource-constrained environments. Additionally, the integration of this technique into existing systems requires careful consideration of privacy concerns, as screenshots may inadvertently capture sensitive information. Addressing these challenges necessitates the development of optimized algorithms capable of handling high-throughput data streams while maintaining low latency. Furthermore, advancements in selective data capture and anonymization techniques are essential to ensure compliance with privacy standards without compromising the quality of insights derived from the analysis.

4.2.3 Enhancing Sequential Data Understanding

Enhancing sequential data understanding remains a pivotal challenge in the development of advanced recommender systems (RS). Current models predominantly rely on exposed samples and explicit user feedback, which inherently limits their efficacy in cold-start scenarios. The emergence of new items exacerbates this issue, as these models struggle to generalize from limited interaction data. Furthermore, the sparsity of user interaction samples presents another significant hurdle, often leading to suboptimal performance in click-through rate (CTR) prediction tasks [29]. Addressing these challenges necessitates innovative approaches that can effectively model and extrapolate from sequential data patterns, thereby improving the robustness and adaptability of RS in dynamic real-world environments.

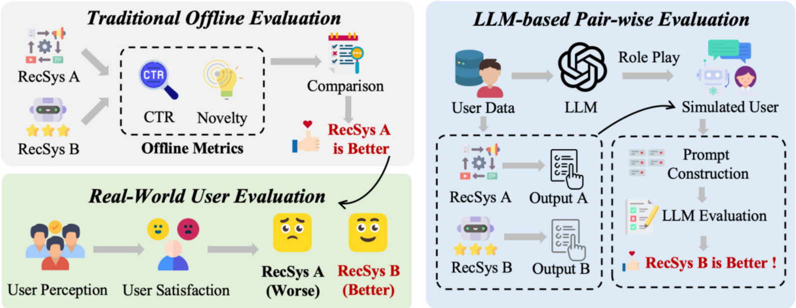


Fig 27: Chart from 'Recsys arena pair wise recommender system evaluation with large language models'

To tackle these limitations, recent advancements have leveraged the complementary strengths of Graph Neural Networks (GNNs) and Large Language Models (LLMs). GNNs excel in capturing higher-order structural information but falter with textual data, whereas LLMs possess superior reasoning capabilities yet lack proficiency in structural comprehension. By integrating these technologies, models like GFM-based RS demonstrate enhanced sequential data understanding through the synergistic exploitation of both textual and structural signals.

This technological complementarity facilitates more nuanced behavior sequence modeling, enabling systems to derive meaningful insights from sparse and heterogeneous interaction data. Such integrative approaches not only improve recommendation accuracy but also enhance the interpretability of model predictions.

Empirical validation of these methodologies has been conducted across multiple real-world datasets, underscoring their effectiveness in enhancing sequential data understanding. These experiments reveal that models incorporating hybrid alignment strategies—melding explicit titles with implicit embeddings—exhibit superior performance in counteracting hallucination issues prevalent in traditional methods [28]. By fine-tuning LLMs on large-scale item ID sequences while preserving textual semantics, these systems achieve a more comprehensive understanding of user behavior trajectories. Consequently, the proposed frameworks yield significant improvements in recommendation precision and generate human-interpretable explanations, marking a substantial advancement in the field of sequential data modeling for recommender systems [28].

4.3 Empirical Validation and Comparative Studies

4.3.1 Evaluating Sample Efficiency of LLMs in Recommendations

The evaluation of sample efficiency in recommendation systems powered by Large Language Models (LLMs) is a critical area of investigation [14]. Sample efficiency refers to the model's ability to achieve high performance with limited training data, an essential trait for practical applications where extensive datasets may not be readily available [14]. LLMs, with their pre-trained knowledge and generalization capabilities, offer promising avenues to improve this aspect [9]. Recent studies indicate that LLM-based recommenders can produce comparable or even superior recommendation distributions with significantly fewer samples than traditional models [7]. This characteristic is particularly advantageous in scenarios like cold-start problems, where user interaction data is sparse. The inherent understanding of language and context within LLMs allows them to leverage minimal data effectively, thus reducing dependency on large historical datasets.

A key factor contributing to the enhanced sample efficiency of LLMs in recommendations lies in their architecture and pre-training [18]. These models are initially trained on vast corpora of text data, enabling them to capture diverse patterns and relationships. When fine-tuned for specific recommendation tasks, they require only a fraction of the domain-specific data to adapt successfully. For instance, frameworks such as Laser have demonstrated that LLMs can match or surpass conventional Collaborative Filtering Methods (CRMs) using substantially reduced training sets [14]. Such efficiency stems from the model's ability to infer user preferences through contextual understanding rather than relying solely on behavioral histories. This reduces the need for extensive interaction logs and enhances the system's scalability across different domains and user bases.

Despite these advancements, challenges remain in fully harnessing the sample efficiency of LLMs for recommendation tasks [18]. One limitation is ensuring that the models maintain personalization while operating with limited data. Personalized responses are crucial in recommendation systems, where individual preferences must be accurately captured and addressed. Moreover, the balance between generalization and task-specific optimization needs careful calibration. Over-reliance on pre-trained knowledge might lead to suboptimal performance if the fine-tuning process does not adequately align with the unique characteristics of the recommendation dataset. Future research should focus on refining techniques that maximize the utilization of LLMs' reasoning capabilities while maintaining high levels of personalization and accuracy in sparse data environments.

4.3.2 Stability Testing Under Dynamic Conditions

Stability testing under dynamic conditions is a critical aspect of evaluating recommender systems (RS), particularly as these systems are increasingly deployed in environments characterized by rapid changes and uncertainties. Dynamic conditions encompass scenarios where user preferences, item availability, and interaction patterns evolve over time. Such volatility necessitates robust stability testing to ensure that RS maintain consistent performance despite fluctuations. Traditional models often falter under these conditions due to their reliance on static datasets and historical user feedback, which may not adequately represent the evolving landscape. Consequently, stability testing must account for temporal dynamics, ensuring that the system adapts effectively without compromising recommendation quality or relevance.

A significant challenge in stability testing under dynamic conditions lies in addressing the sparsity and cold-start issues inherent in real-world applications. The introduction of new items and users into the system creates gaps in the data, making it difficult for conventional RS to generate accurate recommendations. To mitigate this, advanced techniques such as leveraging large language models (LLMs) have been proposed [30]. These models can infer plausible interactions based on contextual and personalized information, thus enhancing stability. However, the efficacy of LLMs is contingent upon the richness of the input data; insufficient context can lead to suboptimal performance. Therefore, stability testing should incorporate mechanisms to evaluate and enhance the model's ability to handle sparse and incomplete data dynamically.

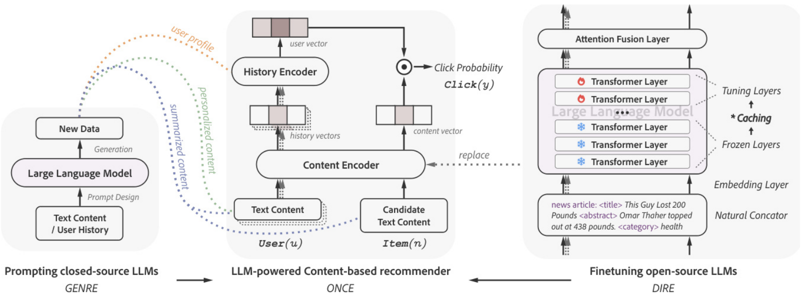


Fig 28: Chart from 'Once boosting content based recommendation with both open and closed source large language models'

Furthermore, stability testing must also consider the causal effects of behavior sequences on prediction accuracy. This involves analyzing how past user interactions influence future recommendations and ensuring that these effects are appropriately emphasized during model tuning. By simulating various dynamic scenarios, researchers can assess the resilience of RS to abrupt changes, such as shifts in user preferences or unexpected item popularity spikes. Effective stability testing frameworks should thus integrate both synthetic and real-world datasets to comprehensively evaluate system robustness. Ultimately, achieving stability under dynamic conditions requires a multifaceted approach that combines innovative modeling techniques with rigorous testing protocols to ensure reliable and adaptive recommender systems.

4.3.3 Causal and Counterfactual Analysis for Bias Reduction

Causal and counterfactual analysis has emerged as a pivotal approach for mitigating biases in machine learning models, particularly within recommender systems. By constructing complementary graphs based on large language models (LLMs), where edges signify purchasing relationships, the framework allows for the modeling of intricate dependencies that traditional methods might overlook. The introduction of a complementary recall module further refines this process by leveraging real exposure click samples to train an Entity-

Entity-Interaction (E-E-I) weight decision model. This model dynamically adjusts edge weights using real-world feedback, ensuring that causal effects are accurately represented. Such an approach not only addresses data sparsity but also reduces inherent biases by emphasizing causality over mere correlation.

Counterfactual reasoning plays a critical role in enhancing the robustness of these models. By adaptively weighting earlier tokens to emphasize their causal impact on predictions, the method ensures that even subtle causal effects are captured without being overshadowed by dominant patterns. This mechanism is particularly effective in scenarios where causal influences diminish toward zero, a situation that often confounds standard training paradigms. The integration of counterfactuals into the fine-tuning process enables the model to simulate alternative outcomes, thereby isolating and quantifying causal relationships [6]. This capability is instrumental in reducing false positives, especially in imbalanced datasets, and improving the overall reliability of recommendations.

The application of causal and counterfactual analysis extends beyond bias reduction to foster explainability and generalizability in recommender systems. By aligning LLMs with rich contextual and personalized information about users and items, the framework generates more credible and semantically interpretable explanations [31]. Experimental evaluations across multiple datasets demonstrate significant improvements in both alignment and explanation generation. Furthermore, the method highlights the limitations of purely statistical imputation strategies, showcasing the superiority of LLM-based approaches in handling subjective tasks. As the field evolves, incorporating causal insights into model design will remain essential for developing systems that are not only accurate but also transparent and equitable.

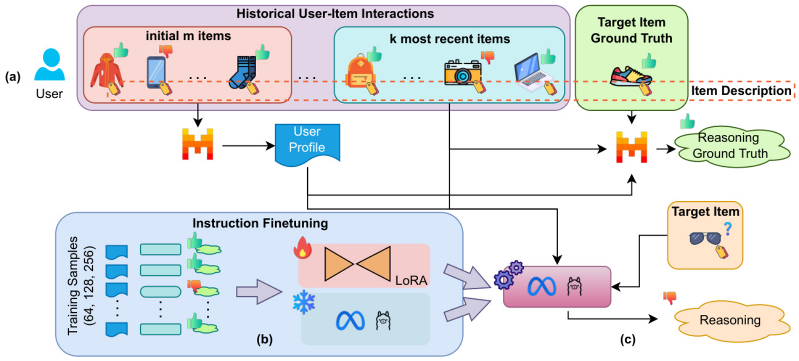


Fig 29: Chart from 'Reasoningrec bridging personalized recommendations and human interpretable explanations through llm reasoning'

5 Prompting and Fine-Tuning Strategies for LLMs in Recommendations

5.1 Instruction Tuning and Collaborative Signals

5.1.1 Self-Supervised Learning for Data Sparsity

Self-supervised learning (SSL) has emerged as a pivotal technique in addressing the pervasive issue of data sparsity within recommender systems [23]. By generating supervisory signals from the data itself, SSL circumvents the reliance on extensive labeled datasets, which are often unavailable or prohibitively expensive to obtain. In scenarios where user-item interactions are sparse, SSL methods extract meaningful representations by designing pretext tasks that exploit the inherent structure of the data. These tasks typically involve predicting missing parts of the input or reconstructing corrupted data, thereby enabling models to learn robust feature embeddings even when interaction data is limited.

In the context of recommendation systems, SSL strategies have been particularly effective when integrated with graph-based models. Graph Neural Networks (GNNs), for instance, leverage self-supervised signals derived from the graph structure to enhance node representations. This is achieved by maximizing mutual information between different views of the graph or by employing contrastive learning techniques that distinguish positive samples from negative ones. Such approaches not only mitigate the impact of sparsity but also improve the generalization capability of the model by capturing higher-order relationships among users and items. Furthermore, SSL can be seamlessly combined with other techniques, such as knowledge distillation or multi-task learning, to further refine the learned representations.

Despite its advantages, the application of SSL in data-sparse environments presents unique challenges. Designing effective pretext tasks that align with the downstream recommendation objective remains a non-trivial problem, as inappropriate tasks may lead to suboptimal representations. Additionally, balancing the trade-off between the complexity of SSL mechanisms and computational efficiency is crucial, especially in large-scale systems. Nevertheless, ongoing advancements in SSL methodologies continue to demonstrate their potential to significantly enhance the performance of recommender systems operating under data-scarce conditions, making them an indispensable tool in modern recommendation pipelines.

5.1.2 Generating Comprehensive Recommendation Explanations

Generating comprehensive recommendation explanations is a pivotal challenge in the development of explainable recommender systems [20]. Large Language Models (LLMs) have emerged as powerful tools for this task, leveraging their extensive semantic understanding to produce human-readable explanations [28]. These models can synthesize diverse information types, including user preferences, item attributes, and contextual data, into coherent narratives that elucidate why a particular recommendation is made [19]. However, the effectiveness of LLMs in generating such explanations hinges on the quality of input representations. Prior approaches often fall short by inadequately capturing personalized features through prompts, leading to suboptimal explanation quality. Addressing this limitation requires innovative strategies to refine feature representations and align them with user-specific needs.

Recent advancements have introduced frameworks like InstructRec, which adopt an instruction-tuning paradigm to enhance the alignment between user needs and model outputs. By allowing users to articulate their preferences in natural language, these systems create a more intuitive interaction model. The design of instruction formats becomes crucial, as it determines how effectively the system interprets user inputs and generates relevant recommendations [32]. Additionally, integrating supplementary information such as item categories and descriptions has shown promise in improving recommendation accuracy. This approach underscores the importance of fine-grained task decomposition, where complex recommendation tasks are broken into manageable sub-tasks, enabling LLMs to better leverage their reasoning capabilities [33].

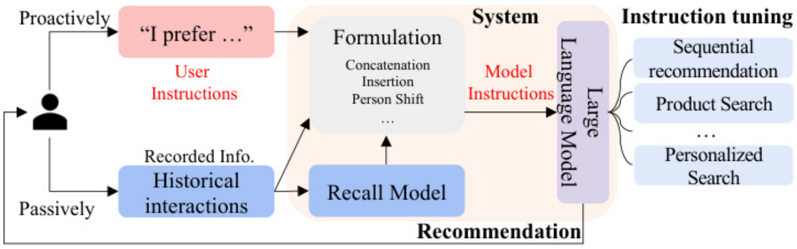


Fig 30: Chart from 'Recommendation as instruction following a large language model empowered recommendation approach'

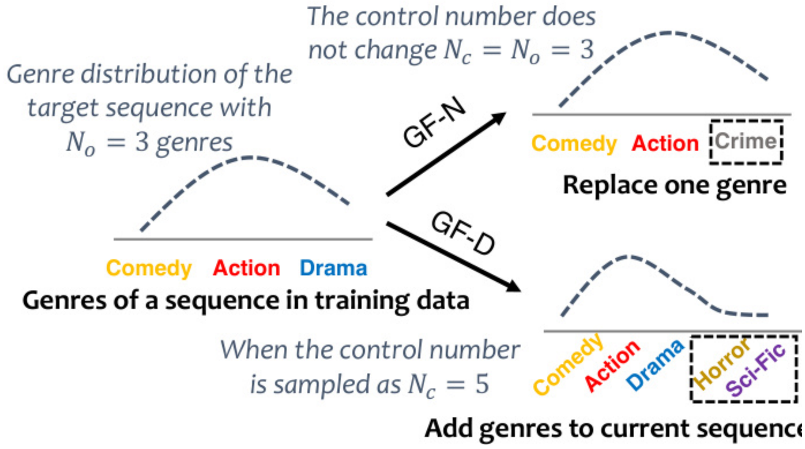

The control number does not change $N_c = N_o = 3$
Genre distribution of the target sequence with $N_o = 3$ genres
GF-N
Comedy Action Crime
Replace one genre
Comedy Action Drama
Genres of a sequence in training data
GF-D
When the control number is sampled as $N_c = 5$
Comedy Action Drama Horror Sci-Fic
Add genres to current sequence

Fig 31: Chart from 'Dlcrec a novel approach for managing diversity in llm based recommender systems'

Despite these advancements, challenges remain in ensuring the robustness and generalizability of explanation generation methods [34]. Noisy or incomplete side information often complicates the extraction of accurate and meaningful features, particularly in data-scarce scenarios. To address this, researchers are exploring techniques to enhance the interpretability of language-based user profiles and improve the alignment of LLM outputs with recommendation logic [35]. Comprehensive evaluations, including ablation studies and robustness analyses, are essential to validate the efficacy of these methods. As the field progresses, developing evaluation metrics that account for transparency, user trust, and engagement will be critical to advancing the state-of-the-art in explainable recommendation systems.

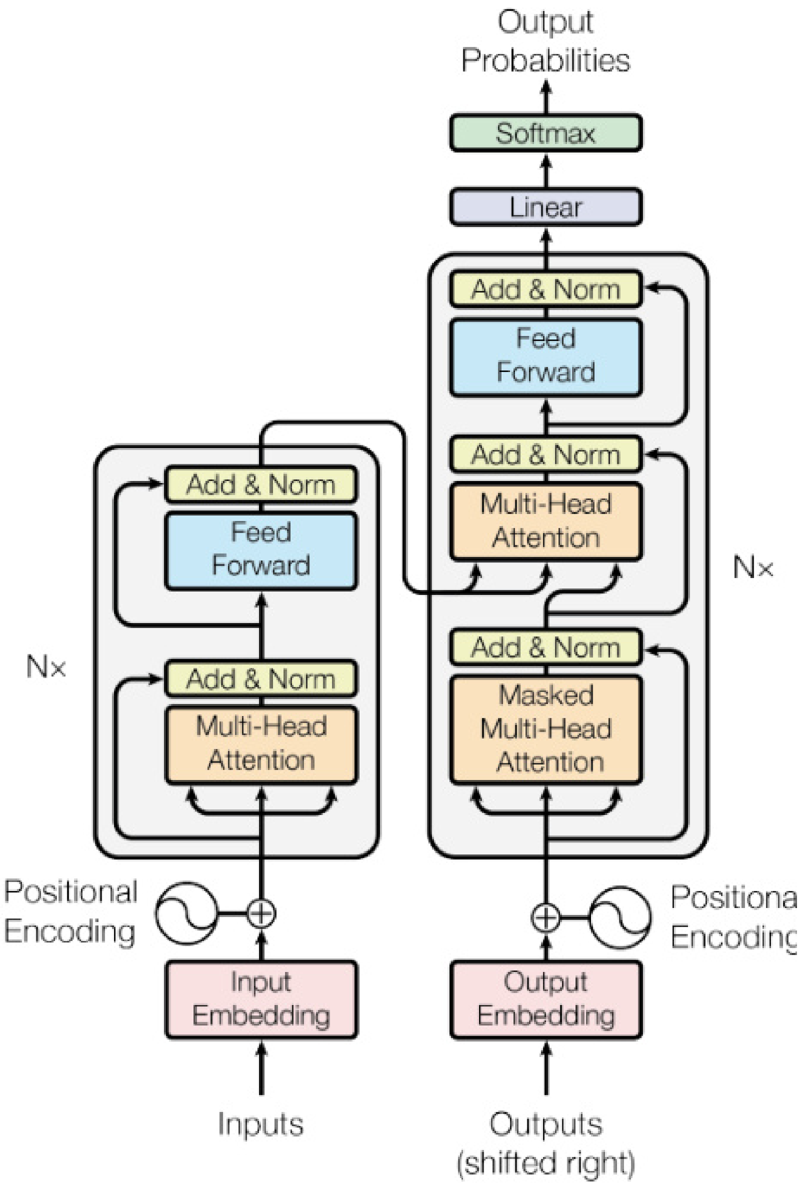
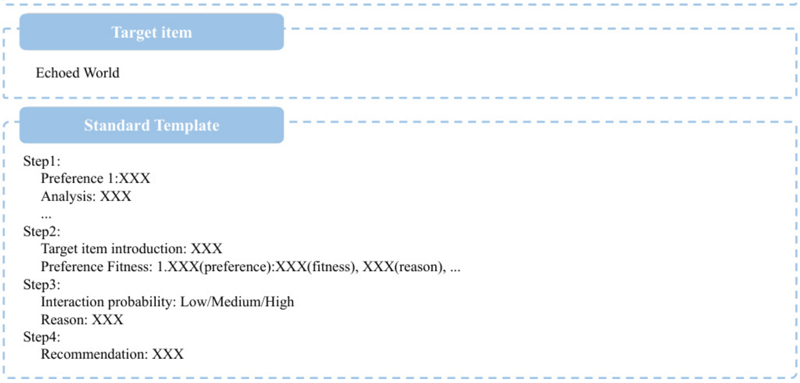


Fig 32: Chart from 'A survey on large language models for personalized and explainable recommendations'



Target item

Echoed World

Standard Template

Step1:
Preference 1:XXX
Analysis: XXX
...
Step2:
Target item introduction: XXX
Preference Fitness: 1.XXX(preference):XXX(fitness), XXX(reason), ...
Step3:
Interaction probability: Low/Medium/High
Reason: XXX
Step4:
Recommendation: XXX

Fig 33: Chart from 'Lane logic alignment of non tuning large language models and online recommendation systems for explainable reason generation'

5.1.3 Fine-Grained Task Decomposition for Diversity Control

Fine-grained task decomposition represents a pivotal strategy in achieving effective diversity control within large language models (LLMs). By breaking down complex tasks into smaller, more manageable subtasks, this approach enables the model to focus on specific aspects of diversity independently. For instance, when generating recommendations, decomposing the task allows the model to address genre, release year, and other attributes separately, ensuring that each facet is adequately represented. This method not only enhances the precision of diversity control but also facilitates the incorporation of detailed attribute-oriented tools, which are crucial for refining candidate item sets based on user preferences. As such, fine-grained decomposition empowers LLMs to deliver more nuanced and varied outputs, aligning closely with user expectations.

The implementation of fine-grained task decomposition often leverages advanced mechanisms like attention-based query-key-value systems to generate collaborative prompts. These prompts capture group-level preferences derived from individual user data, promoting a balanced representation across different demographic or interest groups. Such an architecture contrasts with traditional methods that rely heavily on static, ID-based representations, which struggle in cold-start and zero-shot scenarios. By dynamically adjusting to user interactions, these decomposed tasks allow the model to adapt swiftly to new items or users without extensive retraining. Consequently, the system can maintain high performance even when faced with sparse or incomplete data, demonstrating robust generalization capabilities across diverse application contexts.

Despite its advantages, fine-grained task decomposition introduces certain challenges, particularly concerning computational efficiency and scalability. Decomposing tasks requires additional processing steps, which can increase operational costs and energy consumption—issues already prevalent in LLM deployment. Moreover, ensuring consistency across decomposed subtasks demands meticulous design to avoid fragmented or hallucinatory outputs. To mitigate these limitations, recent research emphasizes treating LLMs as summarization and reasoning engines rather than direct input-output processors, enabling them to handle extensive item pools more effectively [16]. Overall, while fine-grained task decomposition significantly enhances diversity control, addressing its inherent trade-offs remains critical for broader applicability in real-world systems.

5.2 Reinforcement Learning and Attention Mechanisms

5.2.1 Iterative Refinement Through Attribute Signals

Iterative refinement through attribute signals represents a pivotal advancement in recommendation systems, leveraging detailed user and item attributes to enhance model predictions progressively. This process involves the continuous adjustment of recommendations based on feedback loops that utilize attribute-level information, ensuring that each iteration better aligns with user preferences [11]. By incorporating fine-grained attributes such as genre, release year, or actor preferences, models can simulate decision-making processes that closely mimic human reasoning. These iterative refinements are particularly effective in scenarios where initial recommendations may not fully satisfy user expectations, allowing the system to hone in on the most relevant items through successive approximations.

The integration of attribute signals within iterative refinement frameworks often employs sophisticated prompting techniques to guide Large Language Models (LLMs) in generating high-quality recommendations [22]. Prompt tuning, as opposed to full fine-tuning, allows for efficient adaptation of pre-trained models to specific tasks by conditioning them with task-specific prompts [34]. This approach not only reduces computational overhead but also enhances model flexibility by enabling rapid adjustments based on evolving user interactions. The iterative nature of this process ensures that each refinement cycle incorporates updated attribute signals, progressively distilling knowledge from both user behavior and auxiliary recommender models to improve prediction accuracy.

Furthermore, iterative refinement through attribute signals addresses challenges associated with sparse and skewed data distributions commonly encountered in real-world applications. By augmenting training datasets with synthetic examples derived from attribute-based simulations, models gain robustness against rare or unseen scenarios. This methodology significantly enhances the adaptability of recommendation systems, particularly in few-shot learning contexts where traditional models struggle [8]. The continuous interplay between attribute signals and model outputs fosters an environment where recommendations become increasingly personalized and context-aware, ultimately leading to superior user satisfaction and engagement.

5.2.2 Zero-Shot Preference Alignment Without Fine-Tuning

Zero-shot preference alignment without fine-tuning represents a pivotal advancement in leveraging large language models (LLMs) for recommendation systems [36]. Traditional approaches often rely on fine-tuning, which involves adapting a pre-trained model to specific tasks using task-specific datasets. However, this method necessitates significant computational resources and data, limiting its scalability and applicability in scenarios where labeled data is scarce or unavailable. Zero-shot alignment circumvents these limitations by utilizing the inherent knowledge embedded within LLMs, enabling them to comprehend and align with user preferences without additional training [37]. This approach hinges on the model's ability to interpret contextual information from user queries and item descriptions, thereby facilitating accurate recommendations even in the absence of explicit fine-tuning.

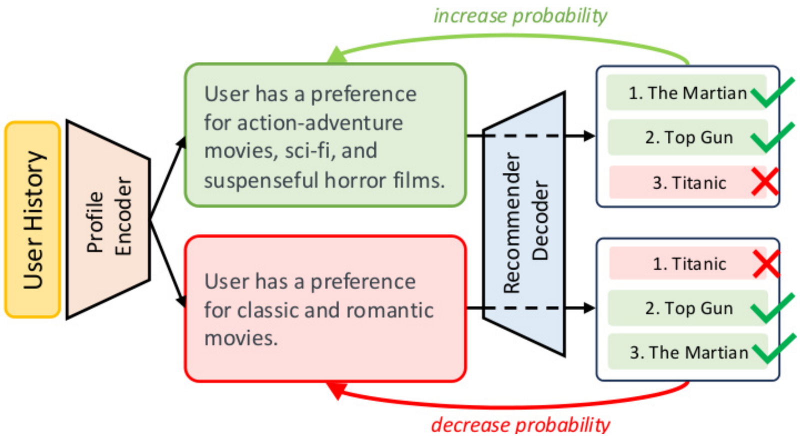


Fig 34: Chart from 'End to end training for recommendation with language based user profiles'

A key strategy in zero-shot preference alignment involves reinforcement learning with tailored reward mechanisms. By designing an alignment stage that incorporates diverse rewards, models can better capture the nuances of controllable recommendation tasks. This method allows the system to adjust its outputs based on feedback, progressively improving its alignment with user preferences. The reinforcement learning framework effectively bridges the gap between the generalized capabilities of LLMs and the specific demands of recommendation tasks, such as rating prediction under zero-shot and few-shot in-context learning settings [17]. Consequently, this approach not only enhances the adaptability of LLMs but also ensures robust performance across various domains without the need for extensive dataset-specific adjustments.

Despite the promising outcomes of zero-shot preference alignment, challenges remain in achieving optimal performance in cold-start scenarios. The dependency on ID-based representations often hinders the model’s ability to generate meaningful recommendations for entirely new items or users [38]. To address this, innovative methods such as attribute-oriented tools and memory strategies have been proposed. These techniques enable the model to explore item facets and leverage intermediate results, refining candidate selections iteratively. Such advancements underscore the potential of zero-shot alignment in enhancing the generalization capabilities of recommendation systems, paving the way for more versatile and user-centric solutions in real-world applications.

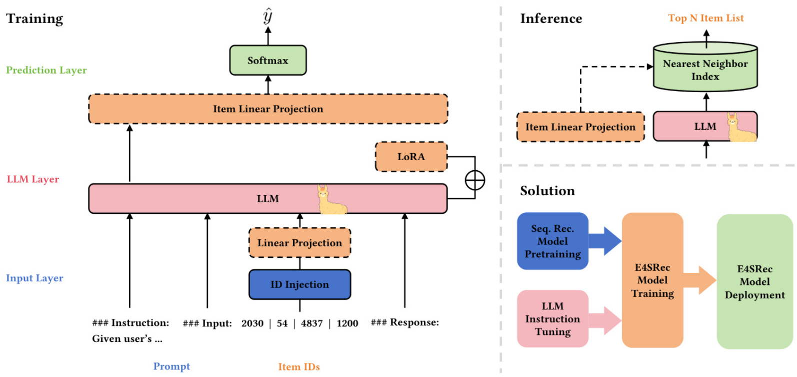


Fig 35: Chart from 'E4srec an elegant effective efficient extensible solution of large language models for sequential recommendation'

5.2.3 End-to-End Training Pipelines for Profile Encoding

End-to-end training pipelines for profile encoding have emerged as a transformative approach in the realm of personalized recommendation systems [36]. These pipelines leverage large language models (LLMs) to generate high-quality user and item profiles by treating intrinsic and extrinsic textual information distinctly during the prompting process [24]. The flexibility of this method allows it to handle diverse textual inputs without necessitating extensive pre-training, thus making it highly adaptable across different domains. By converting traditional recommendation tasks, such as Matrix Factorization and Bayesian Personalized Ranking, into natural language tasks, these pipelines can effectively utilize the capabilities of LLMs [39]. This conversion not only enhances the interpretability of the profiles but also ensures that the downstream performance of the recommender system is maximized.

The architecture of these pipelines often integrates transformer-based models like GPT and BERT, which are renowned for their ability to process sequential data efficiently through parallelization. A key innovation lies in supplementing masked language modeling tasks with fine-grained item correlations derived from users' past interactions [17]. This dual approach enriches the contextual understanding of user preferences, enabling the generation of more accurate and informative profiles. Furthermore, the incorporation of reinforcement learning techniques refines profile quality by mitigating issues related to data noise and over-smoothing, particularly in cold-start scenarios where traditional methods falter. Such advancements underscore the potential of end-to-end pipelines to significantly enhance the efficacy of recommendation systems.

Evaluation of these pipelines reveals their superior performance compared to conventional methods, achieving high accuracy across multiple datasets while maintaining computational efficiency. By employing cost-effective LLMs for prompt exploration, these systems reduce operational costs without compromising prediction accuracy. Additionally, the model-agnostic nature of these pipelines ensures broad applicability, while pre-trained LLMs like GPT-4 can be directly utilized to generate explanations, circumventing the need for additional tuning [35]. This strategic use of advanced LLMs not only simplifies their deployment but also fully leverages their capabilities, addressing challenges posed by closed-source models or limited computational resources. As such, end-to-end training pipelines represent a significant step forward in creating scalable, efficient, and interpretable recommendation systems.

5.3 Supervised and Reinforcement Learning Integration

5.3.1 Two-Stage Alignment for Controllable Recommendations

Two-stage alignment for controllable recommendations represents a significant advancement in the domain of LLM-based recommender systems [33]. This approach is structured to first align the model with large-scale user behavior data, followed by fine-tuning it on specific recommendation tasks [32]. The initial alignment stage leverages reinforcement learning from human feedback (RLHF) techniques, enabling the model to learn generalized user preferences and behaviors. By optimizing the model based on feedback signals from a downstream recommender system, this stage ensures that the LLM captures nuanced patterns in user interactions [9]. The subsequent fine-tuning stage refines the model's ability to generate personalized and context-aware recommendations, thereby enhancing both accuracy and controllability.

A critical aspect of two-stage alignment is its emphasis on reducing formatting errors and improving the diversity of recommendations. Traditional approaches often rely on static prompts or templates, which can lead to suboptimal performance due to oversimplification or misalignment with user preferences. In contrast, the two-stage paradigm dynamically adjusts prompts based on contextual information such as item categories and descriptions. This adaptability not only mitigates biases but also enables fine-grained control over the diversity of recommended items. Experimental evaluations demonstrate that incorporating richer contextual details, such as item titles and descriptions, significantly enhances recommendation quality across diverse datasets.

Furthermore, the two-stage alignment framework addresses challenges associated with scalability and efficiency in LLM-based recommenders [10]. By decoupling the alignment and fine-tuning stages, the approach allows for modular optimization strategies that balance computational costs with performance gains. For instance, cost-effective LLMs can be employed during the alignment phase to reduce API fees and inference times, while high-performance models are reserved for task-specific fine-tuning. This dual-phase strategy not only ensures robustness across various recommendation tasks but also facilitates the integration of emerging techniques, such as tool learning and data augmentation, to further enhance system capabilities. Collectively, these advancements underscore the potential of two-stage alignment as a versatile and effective methodology for controllable recommendations.

5.3.2 Prompt Engineering for Task-Specific Optimization

Prompt engineering has emerged as a pivotal technique for optimizing pretrained language models (PLMs) to achieve task-specific objectives with minimal parameter adjustments. This approach leverages the extensive knowledge embedded within PLMs by conditioning them with carefully designed prompts, enabling more effective utilization of their capabilities, especially in scenarios where downstream data is scarce [34]. Unlike traditional fine-tuning methods that necessitate modifying the entire model's parameters, prompt engineering involves tuning only a small subset of parameters or even just the input representations. This efficiency not only reduces computational overhead but also enhances adaptability across diverse tasks without requiring extensive retraining.

The effectiveness of prompt engineering lies in its ability to guide PLMs toward generating desired outputs through strategic input manipulations. Techniques such as Pretrain, Personalized Prompt, and Predict Paradigm (P5) exemplify this by transforming various tasks into a unified text-to-text format, streamlining the adaptation process. Furthermore, advancements like soft prompt tuning have introduced mechanisms that dynamically adjust prompts based on user preferences and interaction patterns, improving personalization and recommendation accuracy [40]. By integrating external features with collaborative signals, these methods address challenges such as cold-start problems and enhance the system's robustness against sparse data conditions.

Despite its advantages, prompt engineering faces challenges related to generalization and consistency across different tasks and datasets. Empirical studies indicate that no single prompt design universally outperforms others; instead, performance often depends on specific components like categories or descriptions included in the prompt. To mitigate this, reinforcement learning-based frameworks have been proposed to iteratively refine prompts using feedback from downstream tasks. These adaptive strategies ensure that prompts remain relevant and effective, thereby maximizing the utility of PLMs in real-world applications while maintaining flexibility and scalability.

5.3.3 Leveraging Pretrained Models Efficiently

Leveraging pretrained models efficiently has become a cornerstone in advancing various machine learning applications, particularly when labeled data is scarce. Pretrained models, such as large language models (LLMs), possess rich knowledge derived from extensive datasets during their initial training phases [21]. This inherent advantage allows for the effective bridging of gaps between pretraining and downstream objectives, ensuring that the encapsulated knowledge is harnessed optimally [34]. The efficiency of this approach is significantly amplified when only limited downstream data is available, as it mitigates the need for exhaustive data collection and annotation processes. By focusing on prompt engineering, where only a small subset of parameters requires tuning, practitioners can achieve substantial performance improvements with minimal computational overhead.

One notable paradigm illustrating the efficient use of pretrained models is the Pretrain, Personalized Prompt, and Predict Paradigm (P5). This unified text-to-text framework exemplifies how pretrained models can be tailored to diverse tasks without necessitating full fine-tuning. The P5 paradigm capitalizes on two primary strengths: aligning pretraining with specific downstream objectives and minimizing the parameter adjustments needed through prompt engineering. Such an approach not only conserves computational resources but also enhances model adaptability across various applications. Consequently, this method proves particularly advantageous in scenarios where rapid adaptation to new tasks is crucial, enabling models to deliver competent performance even with minimal task-specific data.

Furthermore, the integration of pretrained models within recommendation systems highlights their versatility and robustness [41]. These models exhibit superior reasoning capabilities and vast knowledge bases, facilitating accurate and interpretable recommendations. Their proficiency in generating human-readable explanations fosters user trust and engagement, addressing one of the critical challenges in modern recommendation systems. Additionally, the unified design of LLM-based systems supports multi-tasking, allowing them to excel in zero-shot or few-shot settings [42]. This adaptability underscores the potential of pretrained models to revolutionize fields beyond natural language processing, offering scalable solutions to complex real-world problems.

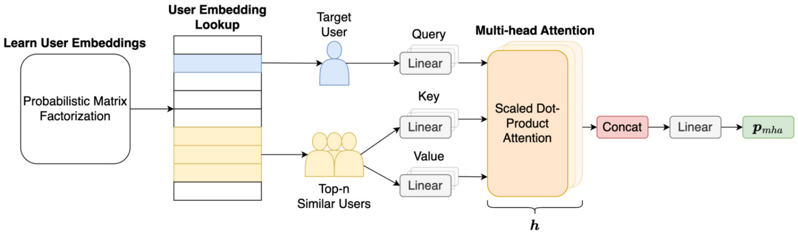


Fig 36: Chart from 'Peapod personalized prompt distillation for generative recommendation'

6 Hybrid Collaborative Filtering and LLM Approaches

6.1 Multimodal and Parameter-Efficient Techniques

6.1.1 Prefix and Suffix Tokens for Collaborative Information

Prefix and suffix tokens have emerged as a pivotal mechanism for incorporating collaborative information into recommendation systems [43]. These tokens serve as markers that delineate specific segments of input data, enabling models to better capture and utilize collaborative signals. In the context of pre-trained language models (PLMs), prefix tokens can be used to encode user-specific or item-specific features at the beginning of an input sequence, while suffix tokens append additional metadata or contextual cues at the end. This structured approach allows PLMs to differentiate between various types of information, such as user preferences, item attributes, or interaction histories, thereby enhancing their ability to generate personalized recommendations.

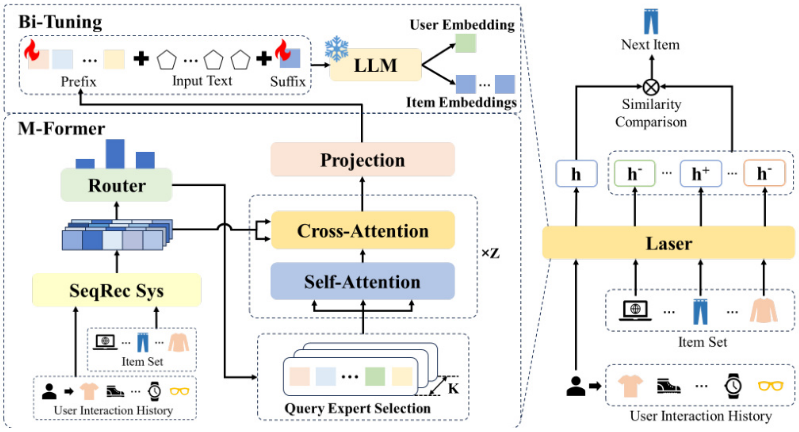


Fig 37: Chart from 'Laser parameter efficient llm bi tuning for sequential recommendation with collaborative information'

The integration of prefix and suffix tokens addresses several limitations inherent in unified frameworks for recommendation tasks [43]. Traditional approaches often struggle to adequately represent personalized feature embeddings due to the lack of explicit mechanisms for isolating collaborative information. By contrast, prefix and suffix tokens provide a structured format for embedding such information directly into the input sequence. For instance, prefix tokens can encapsulate high-level user profiles or behavioral patterns, while suffix tokens may introduce auxiliary details like item categories or descriptions. This dual-token strategy not only improves the granularity of feature representation but also facilitates more effective fine-tuning of PLMs for downstream recommendation tasks.

Despite their advantages, the use of prefix and suffix tokens introduces challenges that warrant further investigation. One notable issue is the potential for increased model complexity, as the inclusion of additional tokens requires careful design to avoid degrading computational efficiency. Moreover, the effectiveness of these tokens is highly dependent on the quality of the embedded information, necessitating robust preprocessing pipelines. Future research should explore optimal strategies for token design and placement, as well as methods to dynamically adapt token usage based on dataset characteristics. Such advancements could unlock the full potential of prefix and suffix tokens in creating more accurate and interpretable recommendation systems.

6.1.2 Variational Autoencoders for Cold Start Solutions

Variational Autoencoders (VAEs) have emerged as a promising solution for addressing cold-start problems in recommendation systems [44]. These models extend traditional autoencoders by incorporating probabilistic elements, enabling them to generate latent representations that capture the underlying distribution of user and item data. In cold-start scenarios, where limited interaction data is available, VAEs excel by leveraging their learned latent space to infer meaningful representations for new users or items. This ability stems from their training process, which involves maximizing a variational lower bound on the data likelihood, thereby encouraging the model to generalize beyond observed interactions. By sampling from the latent distribution, VAEs can produce diverse yet plausible recommendations even when historical data is sparse.

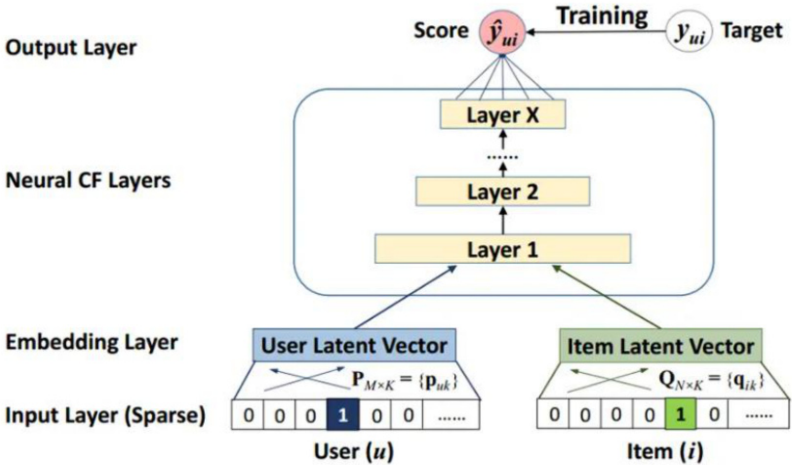


Fig 38: Chart from 'Multi modal clothing recommendation model based on large model and vae enhancement'

The architecture of VAEs allows them to integrate seamlessly with other components of modern recommendation pipelines. For instance, hybrid models that combine VAEs with collaborative filtering techniques have demonstrated superior performance in cold-start settings. These approaches typically encode user-item interactions into a latent space using the VAE framework while simultaneously leveraging side information such as user profiles or item attributes. The probabilistic nature of VAEs ensures that uncertainty in the latent representations is explicitly modeled, leading to more robust predictions. Additionally, recent advancements have explored augmenting VAEs with attention mechanisms or graph-based encoders to further enhance their capacity to handle complex relational data, thereby improving their effectiveness in scenarios with limited interaction histories.

Despite their strengths, VAE-based solutions face challenges that warrant further investigation. One notable limitation is the trade-off between reconstruction accuracy and generalization, which is governed by the choice of prior distributions and regularization techniques. Overly simplistic priors may fail to capture the intricate structure of real-world data, while overly complex priors can lead to computational inefficiencies. Furthermore, the quality of recommendations generated by VAEs heavily depends on the initialization and training stability of the latent space. Addressing these issues through innovative architectural designs or novel training strategies remains an active area of research, offering the potential to unlock even greater capabilities for VAEs in tackling cold-start problems within recommendation systems [44].

6.1.3 Lightweight Querying Transformers for Sequential Tasks

Lightweight querying transformers have emerged as a pivotal advancement for sequential tasks, addressing the computational demands of traditional transformer models while maintaining high performance. These models are designed to optimize the querying process by reducing the complexity associated with attention mechanisms, which are typically resource-intensive. By incorporating techniques such as sparse attention, knowledge distillation, and parameter sharing, lightweight querying transformers achieve significant reductions in both memory usage and inference time. Such optimizations make them particularly suitable for applications requiring real-time processing, such as recommendation systems, where sequential user behavior must be analyzed efficiently. The ability to scale down transformer architectures without compromising their representational power has opened new avenues for deploying these models in resource-constrained environments.

A key aspect of lightweight querying transformers is their adaptability to various sequential tasks through fine-tuning or prompt-based approaches. For instance, models like GPT and BERT have demonstrated versatility in handling tasks ranging from language understanding to personalized recommendations. By leveraging pre-trained weights and adapting them to task-specific objectives, these transformers can effectively capture sequential dependencies in data while minimizing the need for extensive retraining. Furthermore, advancements in few-shot and zero-shot learning have enabled lightweight querying transformers to generalize across domains with limited labeled data. This flexibility is particularly valuable in scenarios where data availability is sparse or where rapid adaptation to new tasks is required.

Despite their advantages, challenges remain in optimizing lightweight querying transformers for specific sequential tasks. Balancing model size and performance often requires careful architectural design and hyperparameter tuning. Additionally, ensuring robustness against biases and noise in sequential data remains an open research question. Recent efforts have focused on integrating external memory components and reinforcement learning strategies to enhance the decision-making capabilities of these models. As the field progresses, further exploration into efficient training paradigms and hybrid architectures will likely yield even more effective solutions for sequential tasks, solidifying the role of lightweight querying transformers as a cornerstone of modern AI systems.

6.2 Retrieval-Augmented and Contextual Models

6.2.1 Embedding and Reranking Strategies for Enhanced Performance

Embedding strategies in recommendation systems have traditionally relied on user-item interaction matrices to generate item embeddings by linearly propagating them across bipartite graphs [36]. While this approach is computationally efficient and straightforward, it often overlooks the rich textual information embedded in auxiliary data such as user reviews, item descriptions, and contextual metadata. This limitation restricts the system's ability to fully capture the nuanced preferences of users and the diverse attributes of items. Recent advancements in Large Language Models (LLMs) present an opportunity to address this gap by leveraging high-quality representations of textual features [45]. These models can extract contextual insights from unstructured data, enabling more informed embedding generation that aligns with real-world user behaviors and item characteristics.

Reranking strategies further enhance recommendation performance by refining the initial outputs of embedding-based models. A common approach involves reformulating recommendation tasks, such as click-through rate prediction or item reranking, into natural language constructs compatible with LLM fine-tuning [43]. However, domain-specific constraints often limit the effectiveness of this method, as LLMs fine-tuned in such a manner may produce outputs that lack contextual relevance. To mitigate these challenges, recent studies propose innovative techniques, such as eliminating user IDs to simplify input spaces and enriching user sequences with item titles or descriptions. These adaptations not only improve model interpretability but also enable the integration of auxiliary information, such as categories or skill-based attributes, which can significantly boost recommendation accuracy depending on the dataset.

Despite their potential, embedding and reranking strategies face practical limitations, particularly in terms of computational costs and scalability. Fine-tuning LLMs for recommendation tasks demands substantial resources, posing challenges for deployment in resource-constrained environments [18]. Moreover, the inherent complexity of neural network-based models can lead to suboptimal performance if not carefully calibrated. To address these issues, hybrid approaches combining high-performance LLMs with cost-effective alternatives have been proposed, achieving a balance between accuracy and efficiency. Such strategies highlight the importance of prompt selection and task decomposition, ensuring that LLMs are effectively guided to leverage their capabilities for diverse and personalized recommendations [18]. These advancements underscore the evolving synergy between embedding techniques and reranking methodologies in driving enhanced recommendation performance.

6.2.2 World Knowledge Integration via LLMs

Large language models (LLMs) have emerged as powerful tools for integrating world knowledge into recommendation systems, leveraging their extensive training on diverse textual data to capture rich contextual and semantic information [46]. Unlike traditional recommender systems that rely heavily on historical user-item interactions, LLM-based approaches can infer preferences even in the absence of explicit interaction data by utilizing their pre-trained knowledge [47]. This capability stems from their proficiency in natural language understanding and reasoning, which allows them to interpret user queries, item descriptions, and other metadata effectively. However, aligning LLMs with the specific requirements of recommendation tasks remains a challenge due to the gap between their general-purpose knowledge and the domain-specific needs of recommendation scenarios [18].

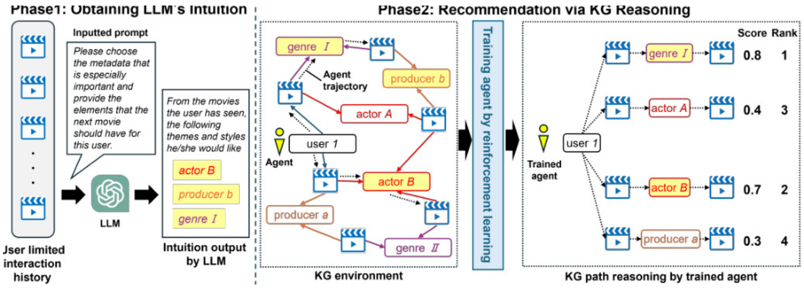

Phase1: Obtaining LLM's Intuition
Phase2: Recommendation via KG Reasoning
Inputted prompt
Please choose the metadata that is especially important and provide the elements that the next movie should have for this user.
LLM
From the movies the user has seen, the following themes and styles he/she would like
actor B
producer b
genre I
Jser limited interaction history
Intuition output by LLM
genre I
Agent trajectory
producer b
actor A
user 1
Agent
actor B
producer a
genre II
KG environment
Training agent by reinforcement learning
Trained agent
user 1
Score
Rank
genre I
0.8
1
actor A
0.4
3
actor B
0.7
2
producer a
0.3
4
KG path reasoning by trained agent

Fig 39: Chart from 'Llm is knowledge graph reasoner llm s intuition aware knowledge graph reasoning for cold start sequential recommendation'

To bridge this gap, recent research has focused on fine-tuning LLMs with domain-specific knowledge and task-oriented prompts [48]. By incorporating recommendation-relevant data during fine-tuning, LLMs can better model complex user-item interactions while retaining their ability to generalize across tasks [32]. For instance, prompts can be designed to include contextual signals such as user preferences, item attributes, and interaction histories, enabling LLMs to generate more accurate and personalized recommendations [18]. Additionally, multi-modal signals and metadata can be seamlessly integrated into the prompting process, enhancing the model's adaptability to various recommendation contexts. This approach not only improves performance but also ensures that the system remains explainable and controllable, addressing key limitations of conventional black-box models.

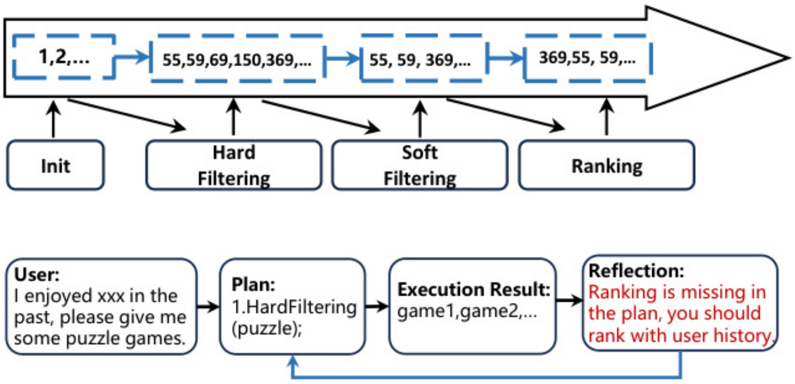

1,2,...
55,59,69,150,369,...
55, 59, 369,...
369,55, 59,...
Init
Hard Filtering
Soft Filtering
Ranking
User:
I enjoyed xxx in the past, please give me some puzzle games.
Plan:
1.HardFiltering (puzzle);
Execution Result:
game1,game2,...
Reflection:
Ranking is missing in the plan, you should rank with user history.

Fig 41: Chart from 'Recommender ai agent integrating large language models for interactive recommendations'

Despite these advancements, challenges remain in fully harnessing the potential of LLMs for world knowledge integration [2]. One critical issue is the need for robust mechanisms to distill high-quality representations from noisy or incomplete data, particularly for users or items with limited interaction histories. Furthermore, ensuring consistency between the semantic understanding of LLMs and the behavioral patterns observed in user-item interactions requires innovative solutions, such as lightweight collaborative adapters that incorporate behavior-aware signals [23]. By addressing these challenges, LLM-based recommender systems can achieve superior performance in zero-shot and few-shot settings, paving the way for more versatile and intelligent recommendation paradigms [17].

6.2.3 Two-Phase Frameworks for Conversational Recommendations

Two-phase frameworks for conversational recommendations have emerged as a pivotal approach to bridge the gap between user intent and system capabilities in recommendation systems. These frameworks typically decompose the recommendation process into two distinct phases: understanding user preferences and generating personalized recommendations. In the first phase, the system leverages large language models (LLMs) to interpret user inputs, often expressed in natural language, transforming them into structured representations that encapsulate user intent and contextual information [37]. This phase is crucial as it sets the foundation for subsequent recommendation generation by ensuring that the system accurately captures nuanced user needs and behavioral patterns.

The second phase focuses on utilizing the structured representations derived from the initial phase to generate tailored recommendations. Here, LLMs are fine-tuned with domain-specific knowledge to enhance their ability to recommend items that align closely with user preferences [4]. A key challenge in this phase is balancing personalization with diversity, ensuring that recommendations are not only relevant but also varied enough to cater to evolving user interests [33]. Recent advancements have introduced task decomposition strategies, breaking down the recommendation task into sub-tasks such as predicting item genres and ranking potential recommendations [33]. This granular approach allows for more precise control over the recommendation process, ultimately improving the quality and relevance of suggested items.

Despite their advantages, two-phase frameworks face several limitations that necessitate further research. One significant issue is the potential loss of contextual information during the transition between phases, which can lead to suboptimal recommendations. Additionally, these frameworks often struggle with scalability and computational efficiency, particularly when handling large datasets or complex user interactions. Future research directions include exploring methods to seamlessly integrate contextual data throughout both phases and developing more efficient algorithms to enhance the scalability of these systems. Addressing these challenges will be instrumental in advancing the field of conversational recommendations and unlocking the full potential of LLMs in personalized recommendation scenarios [16].

6.3 Experimental Validation and Industrial Applications

6.3.1 Offline and Online Evaluations on Industrial Datasets

Offline evaluations on industrial datasets serve as a critical foundation for assessing the performance of recommendation systems before their deployment in real-world scenarios. These evaluations typically involve training models on historical user interaction data and testing their ability to predict future interactions or preferences. A key challenge lies in ensuring that the offline metrics, such as precision, recall, and Mean Reciprocal Rank (MRR), correlate well with actual user satisfaction in live environments. Recent advancements have incorporated transformer-based architectures, such as BERT and GPT, to better capture sequential patterns and contextual nuances in user behavior. However, discrepancies between offline results and online performance often arise due to biases inherent in static datasets, necessitating careful design of evaluation protocols.

Online evaluations complement offline analyses by directly measuring system performance in production environments through A/B testing or multivariate experiments. These evaluations provide insights into user engagement metrics like click-through rates, conversion rates, and session duration, which are more reflective of real-world utility. Industrial datasets used in online evaluations are dynamic and continuously updated, posing challenges related to scalability and adaptability of recommendation algorithms. Techniques like Supervised Fine-Tuning (SFT) have been employed to align pre-trained language models with specific recommendation tasks, yet computational costs and latency constraints remain significant barriers [32]. Addressing these issues requires balancing model complexity with operational efficiency to ensure seamless integration into existing infrastructures.

A hybrid approach that combines offline and online evaluations has emerged as a promising direction for optimizing recommendation systems on industrial datasets. Offline experiments can be augmented with simulated environments that mimic real-world conditions, enabling more robust validation of novel methodologies. Meanwhile, online evaluations benefit from iterative feedback loops where insights gained from user interactions inform subsequent refinements to the system. This dual-phase strategy not only enhances the reliability of performance assessments but also facilitates the identification of edge cases and rare scenarios that might otherwise go unnoticed. By leveraging both evaluation paradigms, researchers and practitioners can develop recommendation systems that are both theoretically sound and practically effective in industrial settings.

6.3.2 Addressing Cold Start and Data Sparsity Issues

Cold-start and data sparsity issues remain critical challenges in recommendation systems, particularly when user-item interactions are limited or absent. Traditional collaborative filtering methods struggle under these conditions due to their reliance on dense interaction matrices. Recent advancements, however, leverage graph-based neural networks (GCNs) and self-supervised learning techniques to mitigate these limitations. GCN-based models like NGCF and LightGCN propagate embeddings across a user-item interaction graph, enabling the system to infer latent relationships even with sparse data. By aggregating neighborhood information, these approaches effectively enrich representations for new users or items, providing a robust solution to cold-start scenarios. Additionally, self-supervised learning strategies generate auxiliary signals from the existing data, creating richer training samples that improve model generalization.

Another promising direction involves integrating pre-trained language models (PLMs) such as BERT and GPT into recommendation pipelines [24]. These models excel in scenarios where explicit feedback is scarce, leveraging their extensive pre-training on large corpora to infer semantic relationships between users and items. For instance, PLMs can utilize item descriptions or user reviews to derive meaningful embeddings, compensating for the lack of interaction data. This capability proves particularly advantageous in cold-start settings, where traditional systems falter due to insufficient behavioral signals. Furthermore, hybrid approaches combining graph-based methods with PLMs have demonstrated superior performance by simultaneously addressing sparsity and enhancing personalization through enriched feature extraction.

Despite these advancements, challenges persist in fully resolving cold-start and sparsity problems. High computational costs associated with training large-scale models like PLMs hinder their widespread adoption, especially for resource-constrained applications. Moreover, the interpretability of recommendations generated by complex models remains limited, posing transparency concerns. To address these gaps, researchers are exploring lightweight architectures and explainable AI techniques. For example, memory-based strategies and fine-tuned attribute encoders have been introduced to enhance efficiency and interpretability. Such innovations pave the way for more practical and scalable solutions, ensuring recommendation systems can deliver accurate and transparent suggestions even in data-scarce environments [2].

6.3.3 Improving Watch Time and Interaction Metrics

Improving watch time and interaction metrics in recommendation systems is pivotal for enhancing user engagement and satisfaction. Recent advancements have leveraged large language models (LLMs) to generate more personalized and context-aware recommendations, thereby increasing the likelihood of user interactions [22]. By converting traditional recommendation tasks into natural language tasks, LLMs can better understand and predict user preferences [39]. This transformation not only facilitates the incorporation of contextual information but also allows for the integration of supplementary tasks like masked language modeling, which refines item correlations based on user past interactions. The adaptability of LLMs to various data sources ensures a comprehensive understanding of user behavior, leading to improved watch time and interaction rates.

Prompt tuning has emerged as a cost-effective strategy for optimizing LLM-based recommenders without necessitating extensive fine-tuning [9]. By storing small task-specific prompts, these models achieve high accuracy across multiple datasets while minimizing computational overhead. Experimental evaluations reveal that incorporating categories or descriptions within prompts can enhance recommendation precision under specific conditions [40]. Furthermore, reinforcement learning (RL) stages with meticulously designed reward signals refine LLMs' alignment with user expectations, ensuring the generation of suboptimal responses is minimized. This dual-stage approach—combining supervised learning (SL) and RL—not only aligns recommendations with user goals but also maintains high performance metrics, crucial for sustaining user interest and interaction over time.

The impact of profile length and dataset size on recommendation efficacy has been thoroughly examined, providing insights into the balance between model complexity and performance [36]. Studies involving human participants and advanced models like GPT-4 underscore the importance of interpretability in maintaining user trust and engagement. Capturing multi-preferences through zero-shot prompt templates enables LLMs to discern intricate user patterns without explicit examples, fostering a more nuanced understanding of individual preferences [35]. As recommender systems evolve, integrating diverse feedback mechanisms and refining output constraints will be essential for continuously improving watch time and interaction metrics, ultimately driving higher user retention and satisfaction.

7 Future Directions

While significant progress has been made in integrating large language models (LLMs) into recommendation systems, several limitations and gaps persist. Current approaches often struggle with scalability, interpretability, and computational efficiency, particularly in real-world deployment scenarios. The auto-regressive nature of LLMs can lead to slower inference times, which poses challenges for applications requiring rapid responses. Additionally, the balance between personalization and fairness remains a critical issue, as biases in training data can propagate into recommendations, undermining user trust. Furthermore, hybrid approaches combining explicit and implicit preferences, while promising, still face challenges in effectively balancing trade-offs between data quality and quantity, especially in cold-start and data-sparse environments.

To address these limitations, future research should focus on several key directions. First, advancements in model compression and optimization techniques could enhance the scalability and efficiency of LLM-based recommendation systems. Techniques such as knowledge distillation, quantization, and sparse attention mechanisms hold promise for reducing computational overhead without compromising performance. Second, there is a need for more robust methods to ensure fairness and mitigate biases in recommendations. This could involve developing adaptive debiasing strategies that dynamically adjust to changes in user behavior and item landscapes, ensuring equitable treatment of all users and items. Third, exploring novel hybrid frameworks that integrate LLMs with other modalities, such as graph neural networks and multimodal data processing, could further improve recommendation accuracy and adaptability. Specifically, aligning structured data with natural language processing through innovative token alignment and cross-modal fusion techniques could bridge existing gaps in representation and understanding.

The potential impact of these proposed directions is substantial. By improving the scalability and efficiency of LLM-based systems, researchers can enable their deployment in resource-constrained environments, making advanced recommendation capabilities accessible to a broader range of applications. Enhanced fairness and interpretability would foster greater user trust and acceptance, paving the way for more transparent and accountable AI-driven systems. Moreover, integrating LLMs with graph-based and multimodal approaches could unlock new levels of personalization and contextual awareness, enabling recommendation systems to deliver more nuanced and relevant suggestions. Collectively, these advancements would not only address current limitations but also lay the groundwork for next-generation recommendation systems capable of meeting the evolving demands of modern users across diverse domains.

8 Conclusion

This survey has provided a comprehensive exploration of the integration of large language models (LLMs) into recommendation systems, highlighting their transformative potential across multiple dimensions. Key findings include the three-tier taxonomy for LLM-based recommenders, which elucidates how these models can represent user preferences, generate tailored recommendations, and address practical deployment challenges. The survey also underscores the significance of hybrid approaches combining explicit and implicit user data, as well as advancements in explainability, fairness, and multimodal data fusion. Notably, frameworks like SUBER and techniques such as reinforcement learning and token alignment have demonstrated innovative pathways to enhance recommendation accuracy, scalability, and user-centricity.

The significance of this survey lies in its structured synthesis of existing methodologies and its identification of critical challenges, such as scalability, interpretability, and computational efficiency. By offering a clear framework for understanding the state of the art, this work bridges theoretical insights with practical applications, empowering researchers and practitioners to navigate the complexities of LLM-enhanced recommendation systems. Furthermore, the survey highlights the role of causal and counterfactual analysis, zero-shot learning, and end-to-end training pipelines in advancing system robustness and adaptability. These contributions collectively illuminate the potential of LLMs to address longstanding issues like cold-start problems and data sparsity while fostering more personalized, transparent, and scalable solutions.

Looking ahead, future research should prioritize addressing the trade-offs between model complexity and operational efficiency, ensuring that LLM-based recommenders remain viable in resource-constrained environments. Additionally, efforts to enhance interpretability and fairness will be crucial for building trust and acceptance among users. As the field continues to evolve, interdisciplinary collaboration and the integration of emerging technologies—such as graph neural networks and generative agents—will play a pivotal role in unlocking the full potential of LLMs in recommendation systems. By advancing both methodological rigor and practical applicability, the research community can pave the way for more intelligent, adaptive, and user-centric recommendation paradigms that meet the demands of modern digital ecosystems.

References

[1] A survey on llm based news recommender systems
[2] A survey on llm powered agents for recommender systems
[3] A review of methods using large language models in news recommendation systems
[4] Chat rec towards interactive and explainable llms augmented recommender system
[5] Enhanced recommendation combining collaborative filtering and large language models
[6] Causality enhanced behavior sequence modeling in llms for personalized recommendation
[7] Towards next generation llm based recommender systems a survey and beyond
[8] Empowering few shot recommender systems with large language models enhanced representations
[9] Recommender systems in the era of large language models llms
[10] A large language model enhanced sequential recommender for joint video and comment recommendation
[11] Lusifer llm based user simulated feedback environment for online recommender systems
[12] Suber an rl environment with simulated human behavior for recommender systems
[13] On generative agents in recommendation
[14] Large language models make sample efficient recommender systems
[15] Llm based cross modality retrieval to improve recommendation performance
[16] Palr personalization aware llms for recommendation
[17] Aligning large language models with recommendation knowledge
[18] Adapting large language models by integrating collaborative semantics for recommendation
[19] Product recommendation system using large language model llama 2
[20] Unveiling user preferences a knowledge graph and llm driven approach for

conversational recommendation
[21] A survey on large language models for recommendation
[22] Rallrec improving retrieval augmented large language model recommendation with representation learning
[23] Xrec large language models for explainable recommendation
[24] A prompting based representation learning method for recommendation with large language models
[25] Pmg personalized multimodal generation with large language models
[26] Training large recommendation models via graph language token alignment
[27] Emerging synergies between large language models and machine learning in ecommerce recommendations
[28] Recexplainer aligning large language models for explaining recommendation models
[29] Recsys arena pair wise recommender system evaluation with large language models
[30] Once boosting content based recommendation with both open and closed source large language models
[31] Reasoningrec bridging personalized recommendations and human interpretable explanations through llm reasoning
[32] Recommendation as instruction following a large language model empowered recommendation approach
[33] Dlcrec a novel approach for managing diversity in llm based recommender systems
[34] A survey on large language models for personalized and explainable recommendations
[35] Lane logic alignment of non tuning large language models and online recommendation systems for explainable reason generation
[36] End to end training for recommendation with language based user profiles
[37] Ragsys item cold start recommender as rag system
[38] E4srec an elegant effective efficient extensible solution of large language models for sequential recommendation
[39] Enhancing user intent for recommendation systems via large language models
[40] Are longer prompts always better prompt selection in large language models for recommendation systems
[41] Genrec large language model for generative recommendation
[42] Peapod personalized prompt distillation for generative recommendation
[43] Laser parameter efficient llm bi tuning for sequential recommendation with collaborative information
[44] Multi modal clothing recommendation model based on large model and vae enhancement
[45] Personalized large language models
[46] Llm is knowledge graph reasoner llm s intuition aware knowledge graph reasoning for cold start sequential recommendation
[47] Sprec self play to debias llm based recommendation
[48] Recommender ai agent integrating large language models for interactive recommendations