02

Large Language Model

-based Generative Recommendation

Action Tokenization

Human-readable Data 👨 👩

Tokenize

Machine-readable Data 🤖

Premium Men’s Short Sleeve Athletic Training T-Shirt Made of Lightweight Breathable Fabric, Ideal for Running, Gym Workouts, and Casual Sportswear in All Seasons; High-Performance Breathable Cotton Crew Socks for Men with Arch Support, Cushioned Heel and Toe, and Moisture Control, Perfect for Sports, Walking, and Everyday Comfort; Men’s Loose-Fit Basketball Shorts with Elastic Drawstring Waistband, Quick-Dry Mesh Fabric, and Printed Number 11 for Professional and Recreational Play; Official Size 7 Composite Leather Basketball Designed for Indoor and Outdoor Use, Deep Channel Design for Enhanced Grip and Ball Control, Ideal for Training and Competitive Matches;

Text description of each action

Cons: *Inefficient;*

Action Tokenization

Human-readable Data 👨 👩

Tokenize

Machine-readable Data 🤖

Premium Men’s Short Sleeve Athletic Training T-Shirt Made of Lightweight Breathable Fabric, Ideal for Running, Gym Workouts, and Casual Sportswear in All Seasons; High-Performance Breathable Cotton Crew Socks for Men with Arch Support, Cushioned Heel and Toe, and Moisture Control, Perfect for Sports, Walking, and Everyday Comfort; Men’s Loose-Fit Basketball Shorts with Elastic Drawstring Waistband, Quick-Dry Mesh Fabric, and Printed Number 11 for Professional and Recreational Play; Official Size 7 Composite Leather Basketball Designed for Indoor and Outdoor Use, Deep Channel Design for Enhanced Grip and Ball Control, Ideal for Training and Competitive Matches;

Text description of each action

Action Tokenization

Cons: *Inefficient;*

Pros: *The underlying distribution aligns with that of LLMs*

Human-readable Data 👨 👩

Tokenize

Machine-readable Data 🤖

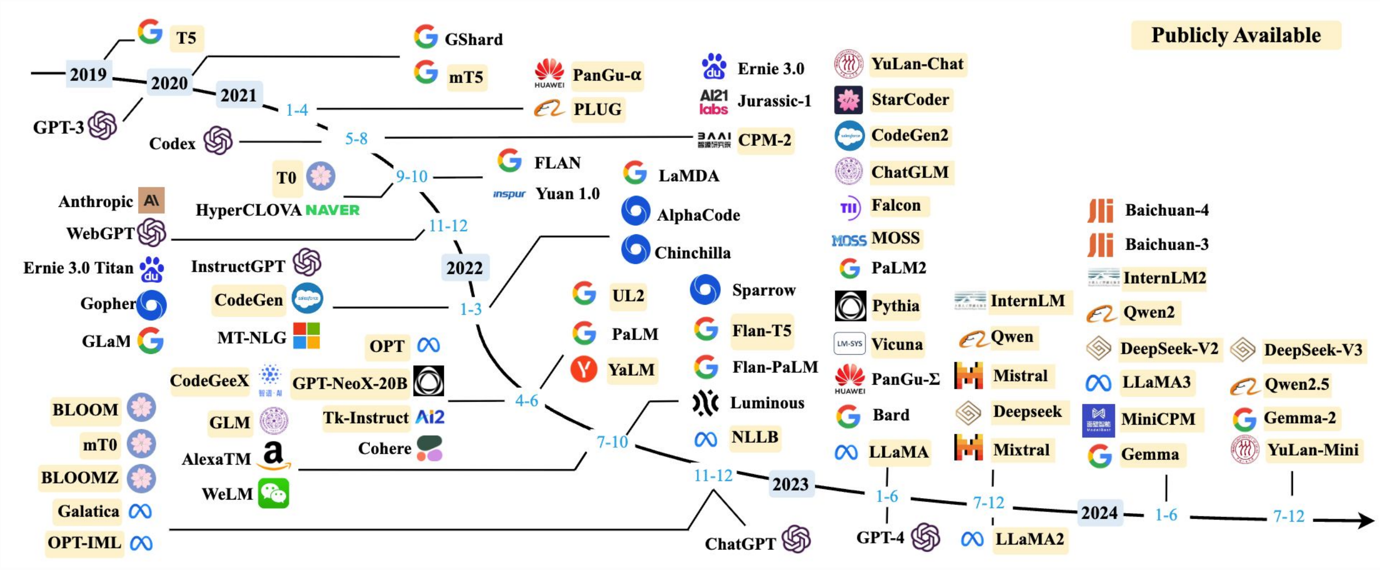
Premium Men’s Short Sleeve Athletic Training T-Shirt Made of Lightweight Breathable Fabric, Ideal for Running, Gym Workouts, and Casual Sportswear in All Seasons; High-Performance Breathable Cotton Crew Socks for Men with Arch Support, Cushioned Heel and Toe, and Moisture Control, Perfect for Sports, Walking, and Everyday Comfort; Men’s Loose-Fit Basketball Shorts with Elastic Drawstring Waistband, Quick-Dry Mesh Fabric, and Printed Number 11 for Professional and Recreational Play; Official Size 7 Composite Leather Basketball Designed for Indoor and Outdoor Use, Deep Channel Design for Enhanced Grip and Ball Control, Ideal for Training and Competitive Matches;

Text description of each action

The Rise of Large Language Models

Transformer

2017



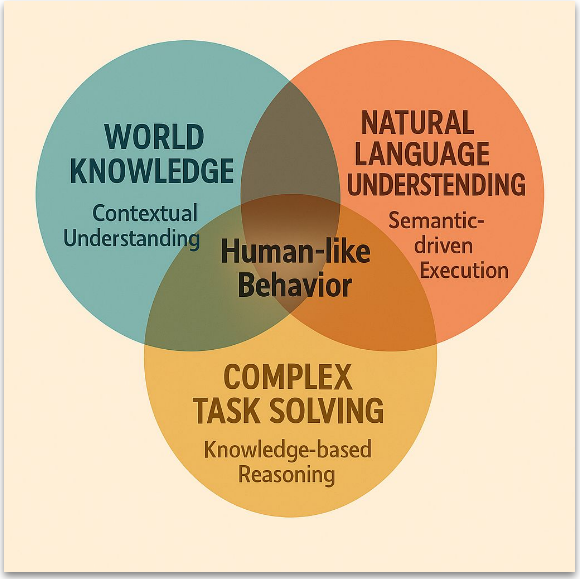
O3, R1...

2025

LLMs are developing so fast recently...

Zhao et al. A Survey of Large Language Models. arXiv:2303.18223

Large Language Models

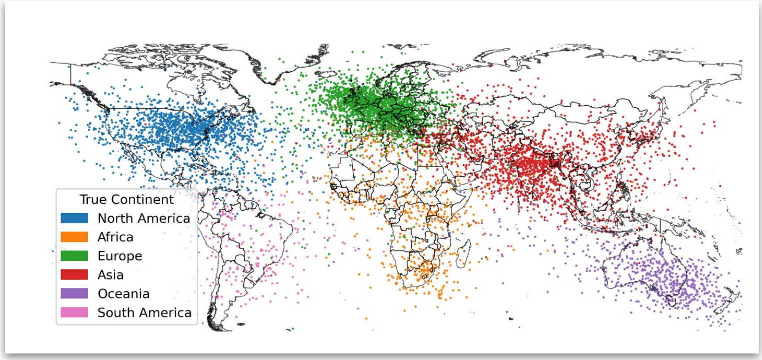


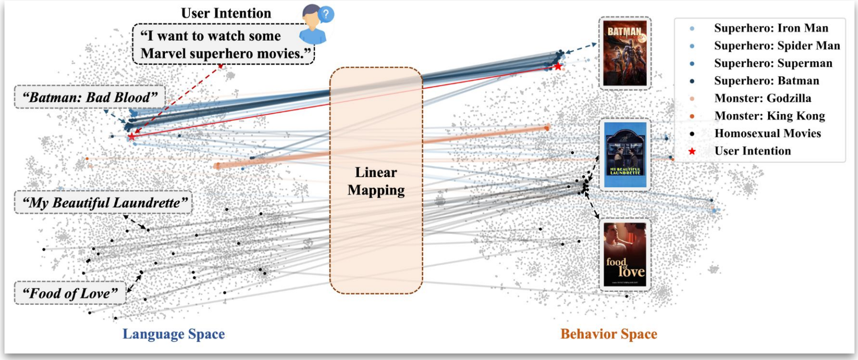
Key features:

- World knowledge.
- Natural language understanding.
- Human-like behavior.

Benefits of LLMs for Recommendation

(1) World knowledge – from pretraining





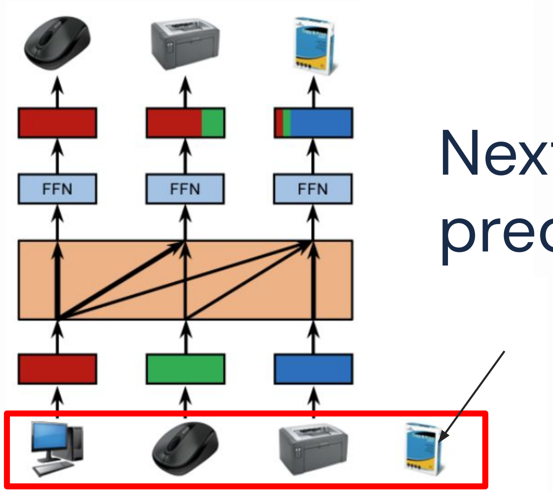
In space

In recommendation

Gurnee et al. Language Models Represent Space and Time. ICLR 2024.
Sheng et al. Language Representations Can be What Recommenders Need: Findings and Potentials. ICLR 2025.

Benefits of LLMs for Recommendation

(1) World knowledge



Next ID prediction

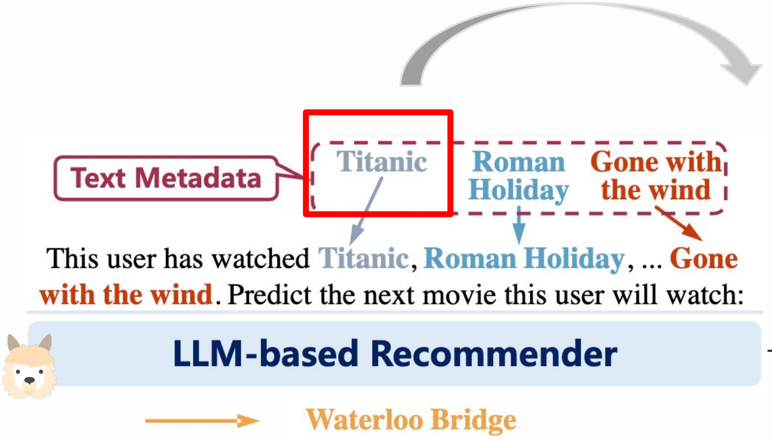
Item IDs

ID-based item modeling lack semantic meanings

Example: SASRec [*ICDM'18*]

Benefits of LLMs for Recommendation

(1) World knowledge

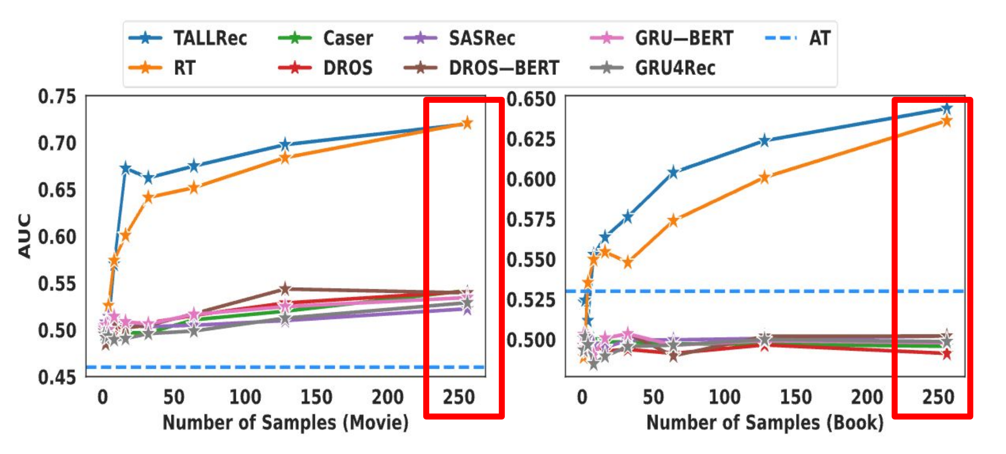


Titanic is a 1997 epic romance and disaster film directed by James Cameron, telling the tragic love story between Jack and Rose aboard the ill-fated RMS Titanic. It blends historical events with fictional drama, becoming one of the most iconic and emotionally powerful films of all time.

Abundant prior knowledge about items

Benefits of LLMs for Recommendation

(1) World knowledge

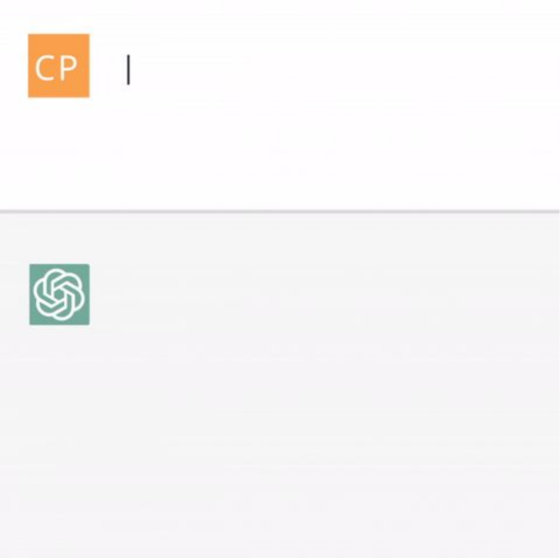


Few data -> a good recommender

Bao et al. TALLRec: An Effective and Efficient Tuning Framework to Align Large Language Model with Recommendation. RecSys 2023.

Benefits of LLMs for Recommendation

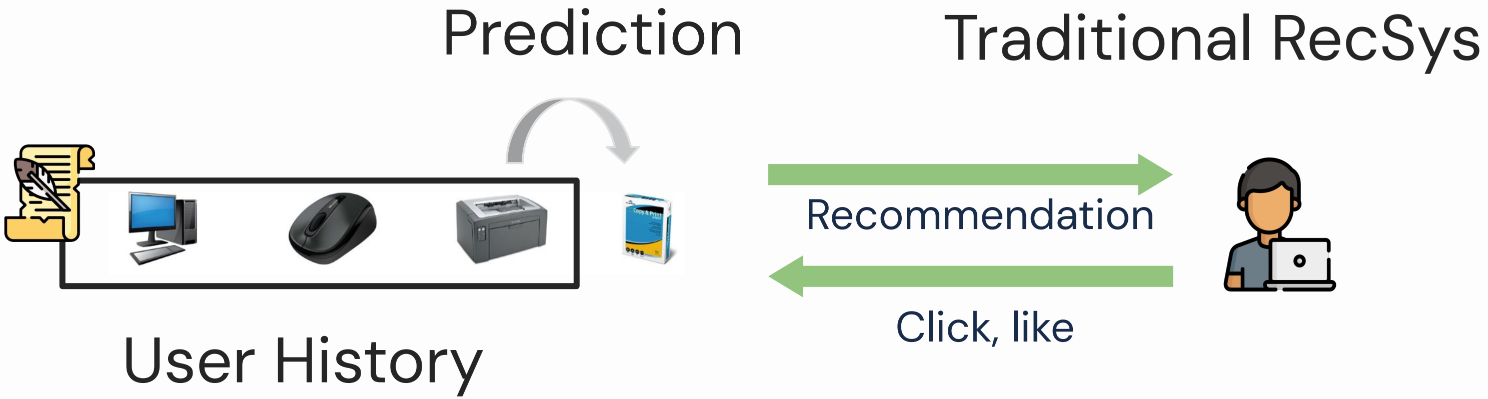
(2) Natural language understanding & generation



LLMs can interact with users fluently

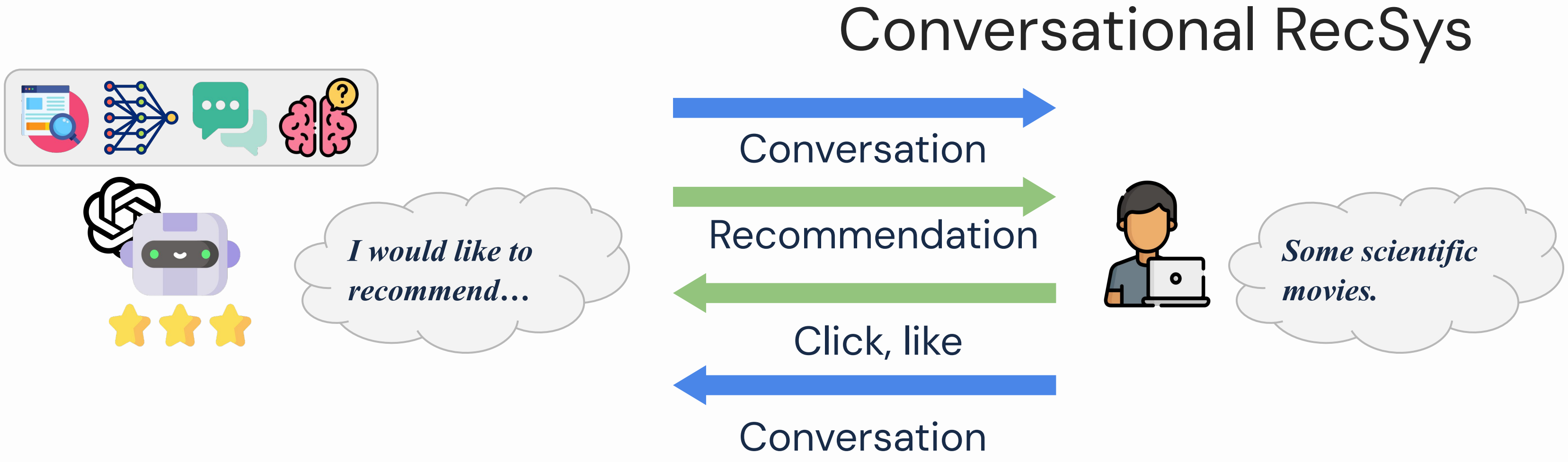
Benefits of LLMs for Recommendation

(2) Natural language understanding & generation



Benefits of LLMs for Recommendation

(2) Natural language understanding & generation



Benefits of LLMs for Recommendation

(3) Human-like behavior

Park et al. Generative Agents: Interactive Simulacra of Human Behavior. UIST 2023

Benefits of LLMs for Recommendation

(3) Human-like behavior

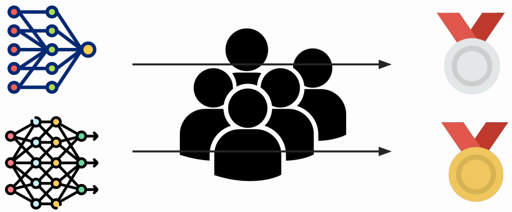
Offline recommender evaluation

Inaccurate, but affordable

Benefits of LLMs for Recommendation

(3) Human-like behavior

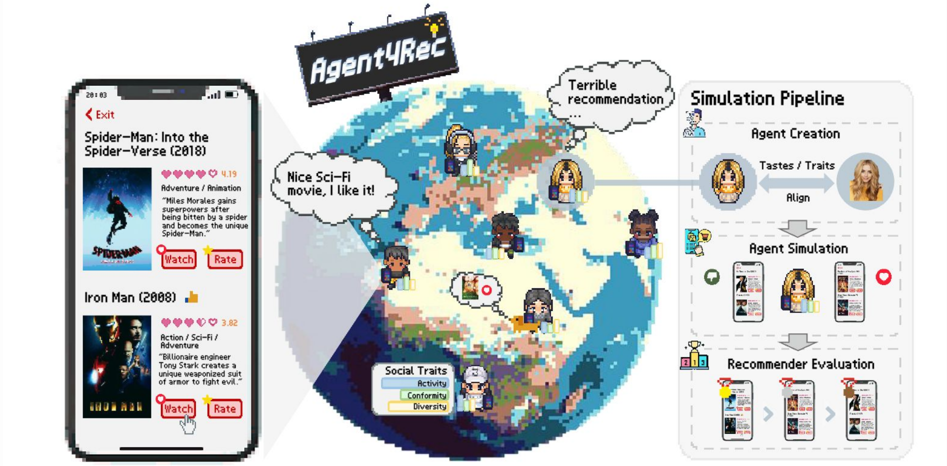
Online recommender evaluation



Accurate, but costly

Benefits of LLMs for Recommendation

(3) Human-like behavior



[LLM as user simulator]()

Faithful
Affordable
Controllable
...

Zhang et al. On Generative Agents in Recommendation. SIGIR 2024

Part 1: LLM as Sequential Recommender

(i) **Early efforts**: Pretrained LLMs for recommendation;

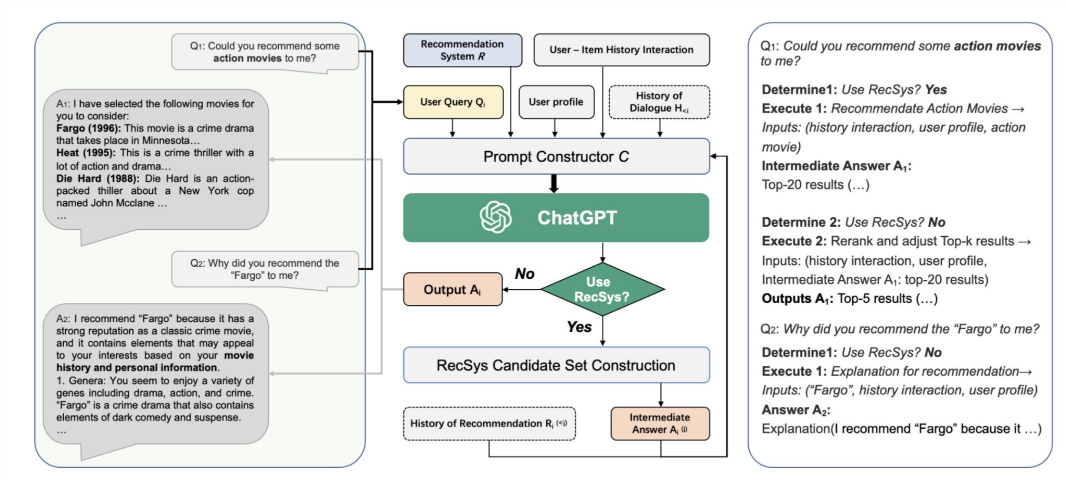
Early efforts

- Directly use freezed LLMs (e.g., GPT 4) for recommendation.

Early efforts

Prompt Engineering + In-Context Learning (ChatRec)

Key idea: LLMs as the recsys controller

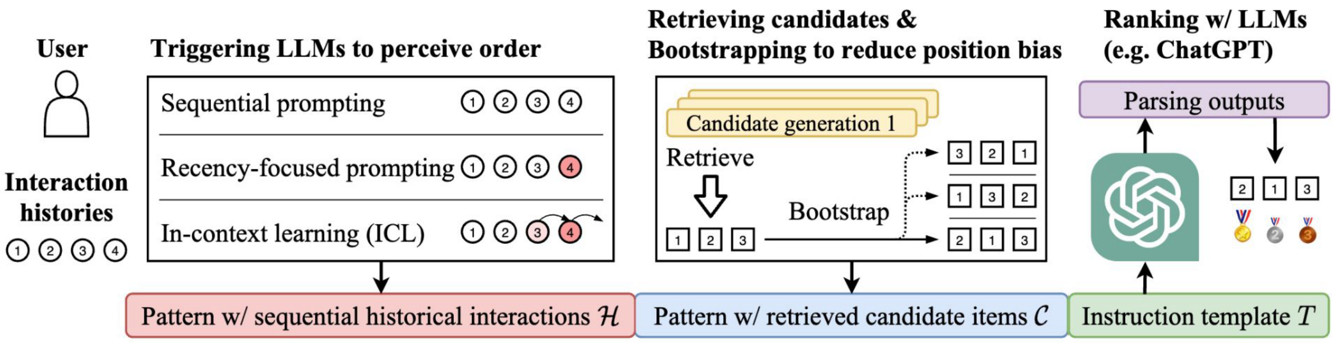


Gao et al. Chat-REC: Towards Interactive and Explainable LLMs-Augmented Recommender System. arXiv 2303.14524.

Early efforts

Prompt Engineering + In-Context Learning (LLMRank)

Key idea: LLMs as the reranker



Hou et al. Large Language Models are Zero-Shot Rankers for Recommender Systems. ECIR 2024.

Early efforts

- Directly use freezed LLMs (e.g., GPT 4) for recommendation.
- A performance gap compared to traditional recommenders exists.

Early efforts

Sub-optimal performance

	Method	ML-1M				Games			
		N@1	N@5	N@10	N@20	N@1	N@5	N@10	N@20
full	Pop	0.08	1.20	4.13	5.79	0.13	1.00	2.27	2.62
full	BPRMF [49]	0.26	1.69	4.41	6.04	0.55	1.98	**2.96**	**3.19**
full	SASRec [33]	**3.76**	**9.79**	**10.45**	**10.56**	**1.33**	**3.55**	**4.02**	**4.11**
zero-shot	BM25 [50]	0.26	0.87	2.32	5.28	0.18	1.07	1.80	2.55
zero-shot	UniSRec [30]	0.88	3.46	5.30	6.92	0.00	1.86	2.03	2.31
zero-shot	VQ-Rec [29]	0.20	1.60	3.29	5.73	0.20	1.21	1.91	2.64
zero-shot	Ours	**1.74**	**5.22**	**6.91**	**7.90**	**0.90**	**2.26**	2.80	3.08

Hou et al. Large Language Models are Zero-Shot Rankers for Recommender Systems. ECIR 2024.

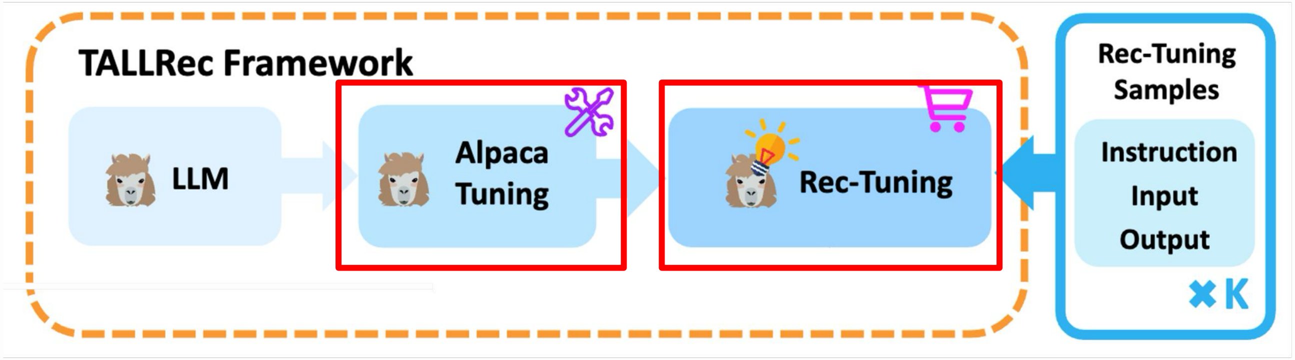
Part 1: LLM as Sequential Recommender

(i) Early efforts: Pretrained LLMs for recommendation;

(ii) **Aligning** LLMs for recommendation;

Aligning LLMs for recommendation

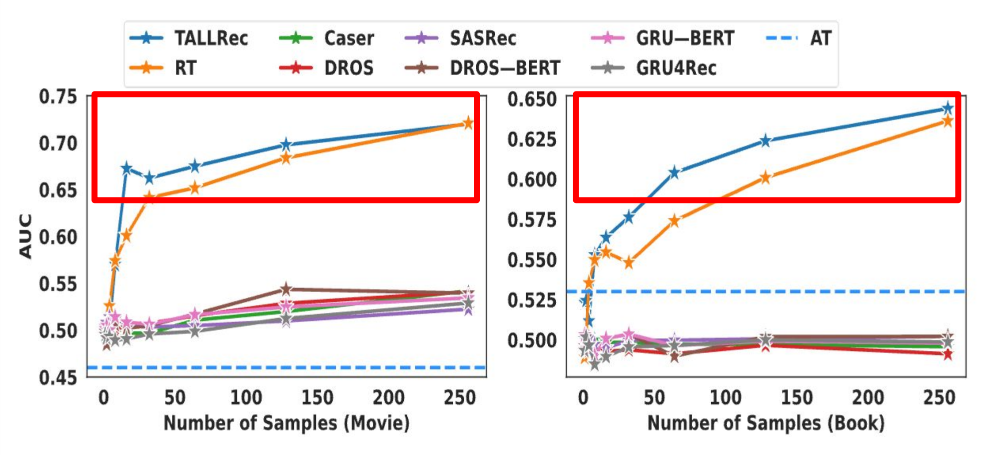
TALLRec



General task alignment -> Recommendation alignment

Aligning LLMs for recommendation

TALLRec



Good recommender with few training instances

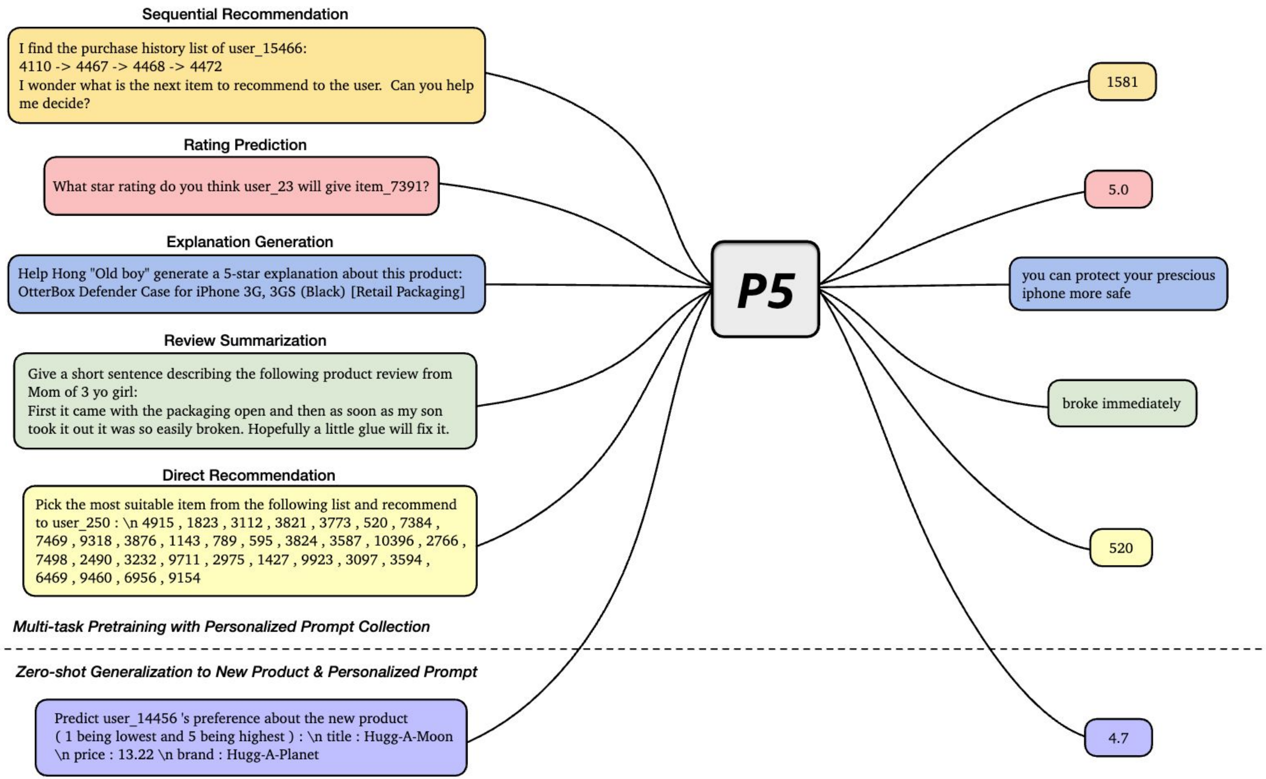
Bao et al. TALLRec: An Effective and Efficient Tuning Framework to Align Large Language Model with Recommendation. RecSys 2023.

Aligning LLMs for recommendation

P5

Multi-task

Cross-task generalization



Geng et al. Recommendation as Language Processing (RLP): A Unified Pretrain, Personalized Prompt & Predict Paradigm (P5). RecSys 2022.

Aligning LLMs for recommendation

InstructRec

Table 1: Example instructions with various types of user preferences, intentions, and task forms. To enhance the readability, we make some modifications to the original instructions that are used in our experiments.

Instantiation	Model Instructions
$\langle P_1, I_0, T_0 \rangle$	The user has purchased these items: <historical interactions>. Based on this information, is it likely that the user will interact with <target item> next?
$\langle P_2, I_0, T_3 \rangle$	You are a search engine and you meet a user's query: <explicit preference>. Please respond to this user by selecting items from the candidates: <candidate items>.
$\langle P_0, I_1, T_2 \rangle$	As a recommender system, your task is to recommend an item that is related to the user's <vague intention>. Please provide your recommendation.
$\langle P_0, I_2, T_2 \rangle$	Suppose you are a search engine, now the user search that <specific Intention>, can you generate the item to respond to user's query?
$\langle P_1, P_2, T_2 \rangle$	Here is the historical interactions of a user: <historical interactions>. His preferences are as follows: <explicit preference>. Please provide recommendations.
$\langle P_1, I_1, T_2 \rangle$	The user has interacted with the following <historical interactions>. Now the user search for <vague intention>, please generate products that match his intent.
$\langle P_1, I_2, T_2 \rangle$	The user has recently purchased the following <historical items>. The user has expressed a desire for <specific intention>. Please provide recommendations.

Unify recommendation & search via instruction tuning

Zhang et al. Recommendation as Instruction Following: A Large Language Model Empowered Recommendation Approach. TOIS.

Part 1: LLM as Sequential Recommender

(i) Early efforts: Pretrained LLMs for recommendation;
(ii) Aligning LLMs for recommendation;
(iii) **Training** objective & **inference**

Training objective

(1) Supervised finetuning (SFT)

I have watched Titanic, Roman Holiday, … Gone with the wind. Predict the next movie I will watch:

Training objective

(1) Supervised finetuning (SFT)

I have watched Titanic, Roman Holiday, … Gone with the wind. Predict the next movie I will watch:

Waterloo Bridge.

Prediction

Training objective

(1) Supervised finetuning (SFT)

I have watched Titanic, Roman Holiday, … Gone with the wind. Predict the next movie I will watch:
Waterloo Bridge.

Prediction

Training objective

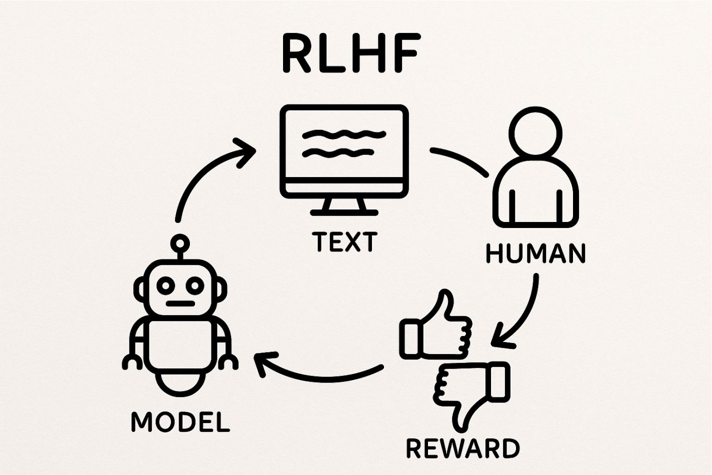
(1) Supervised finetuning (SFT)

$$\mathcal{L}_{\mathrm{SFT}}(\theta) = -\mathbb{E}_{(x,y)\sim\mathcal{D}}\left[\sum_{t=1}^{T}\log \boxed{P_\theta(y_t \mid y_{<t})}\right]$$

Always predict the next token

Training objective

(2) Preference learning



LLMs are trained to align human preferences

Recommendation is about user preferences

Training objective

(2) Preference learning

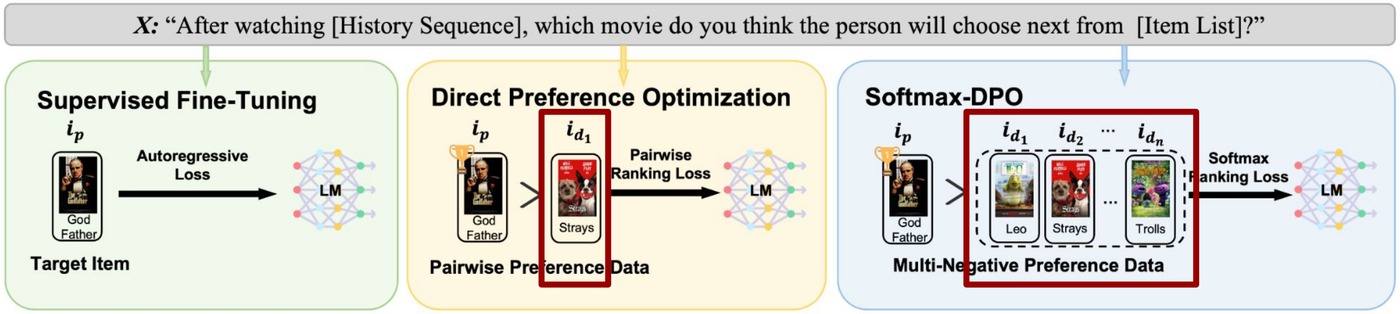
I have watched Titanic, Roman Holiday, … Gone with the wind. Predict the next movie I will watch:

Waterloo Bridge > Harry Potter

Rafailov et al. Direct Preference Optimization: Your Language Model is Secretly a Reward Model. NeurIPS 2023.

Training objective

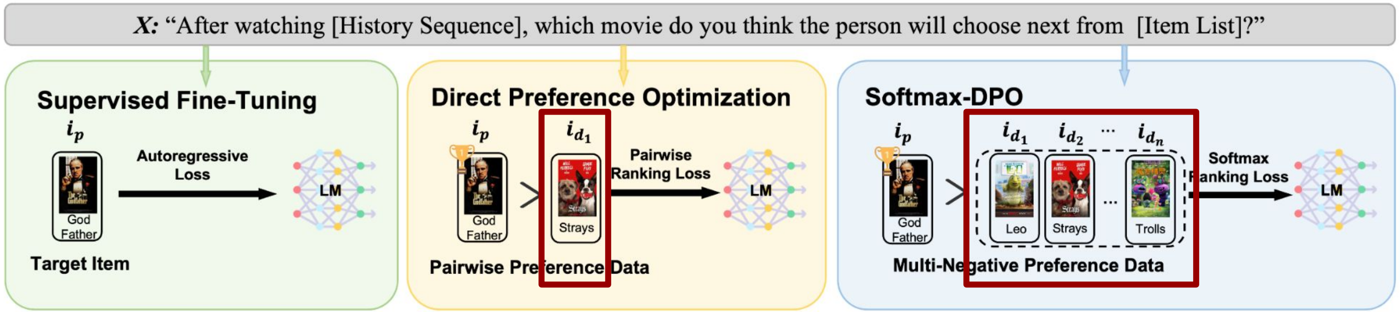
(2) Preference learning (S-DPO [NeurIPS'24])



Single negative → Multiple negatives

Training objective

(2) Preference learning (S-DPO [NeurIPS'24])



$$\mathcal{L}_{\mathrm{S-DPO}}(\pi_\theta;\pi_{\mathrm{ref}}) = -\mathbb{E}_{(x_u,e_p,\mathcal{E}_d)\sim\mathcal{D}}\left[\log\sigma\left(-\log\sum_{e_d\in\mathcal{E}_d}\exp\left(\beta\log\frac{\pi_\theta(e_d|x_u)}{\pi_{\mathrm{ref}}(e_d|x_u)}-\beta\log\frac{\pi_\theta(e_p|x_u)}{\pi_{\mathrm{ref}}(e_p|x_u)}\right)\right)\right].$$

Chen et al. On Softmax Direct Preference Optimization for Recommendation. NeurIPS 2024.

Training objective

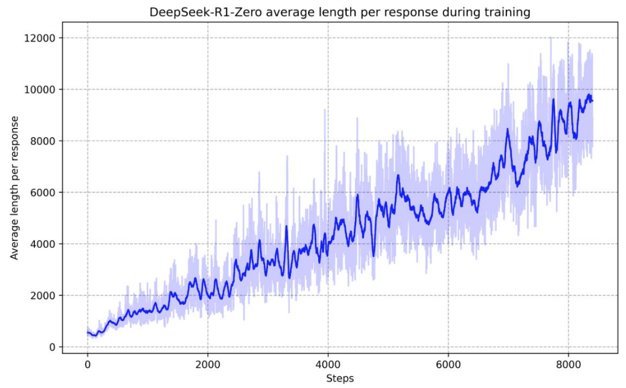
(3) Reinforce learning

Emergent reasoning capabilities through RL

$$\mathcal{J}_{GRPO}(\theta) = \mathbb{E}[q \sim P(Q), \{o_i\}_{i=1}^{G} \sim \pi_{\theta_{old}}(O|q)]$$

$$\frac{1}{G}\sum_{i=1}^{G}\left(\min\left(\frac{\pi_\theta(o_i|q)}{\pi_{\theta_{old}}(o_i|q)}A_i, \text{clip}\left(\frac{\pi_\theta(o_i|q)}{\pi_{\theta_{old}}(o_i|q)}, 1-\varepsilon, 1+\varepsilon\right)A_i\right) - \beta\mathbb{D}_{KL}\left(\pi_\theta||\pi_{ref}\right)\right), \quad (1)$$

$$\mathbb{D}_{KL}\left(\pi_\theta||\pi_{ref}\right) = \frac{\pi_{ref}(o_i|q)}{\pi_\theta(o_i|q)} - \log\frac{\pi_{ref}(o_i|q)}{\pi_\theta(o_i|q)} - 1, \quad (2)$$

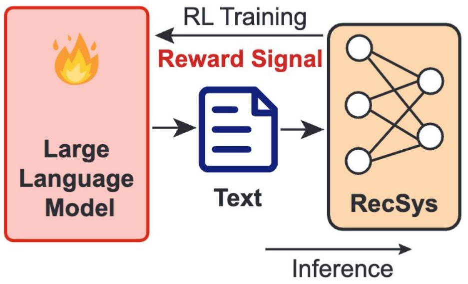


Question: If $a > 1$, then the sum of the real solutions of $\sqrt{a - \sqrt{a+x}} = x$ is equal to

Response: <think>
To solve the equation $\sqrt{a - \sqrt{a+x}} = x$, let's start by squaring both $\cdots$
$\left(\sqrt{a - \sqrt{a+x}}\right)^2 = x^2 \implies a - \sqrt{a+x} = x^2.$
Rearrange to isolate the inner square root term:
$(a - x^2)^2 = a + x \implies a^2 - 2ax^2 + (x^2)^2 = a + x \implies x^4 - 2ax^2 - x + (a^2 - a) = 0$
$\ldots$
Wait, wait. Wait. That's an aha moment I can flag here.
Let's reevaluate this step-by-step to identify if the correct sum can be $\cdots$
We started with the equation:
$\sqrt{a - \sqrt{a+x}} = x$
First, let's square both sides:
$a - \sqrt{a+x} = x^2 \implies \sqrt{a+x} = a - x^2$
Next, I could square both sides again, treating the equation: $\cdots$
$\ldots$

Training objective

(3) Reinforce learning (Rec-R1)



$$\max_{\theta} \mathbb{E}_{s\sim p(s), a\sim\pi_\theta(a|s)} \boxed{[f(a|s)]}.$$

Ranking metrics as rewards

Prompt Template for Rec-R1 + Dense Retriever (Product Search)

```
You are an expert in generating queries for dense retrieval. Given a customer
query, your task is to retain the original query while expanding it with
additional semantically relevant information, retrieve the most relevant
products, ensuring they best meet customer needs. If no useful expansion is
needed, return the original query as is.

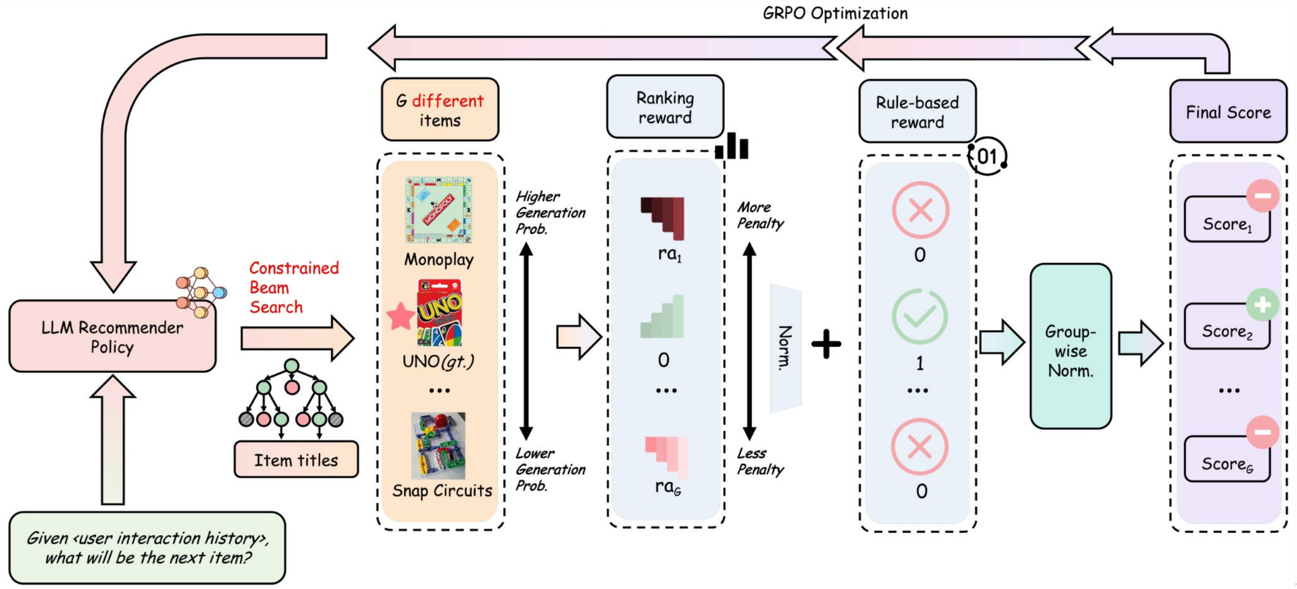
Below is the query:
``` {user_query} ```

<|im_start|>system
You are a helpful AI assistant. You first think about the reasoning process
in the mind and then provide the user with the answer.
<|im_end|>
<|im_start|>user
[PROMPT as above]
Show your work in <think>\think> tags. Your final response must be in JSON
format within <answer>\answer> tags. For example,
<answer>
{ "query": xxx }
</answer>.
<|im_end|>
<|im_start|>assistant
Let me solve this step by step.
<think>
```
```

Training objective

(3) Reinforce learning (ReRe)

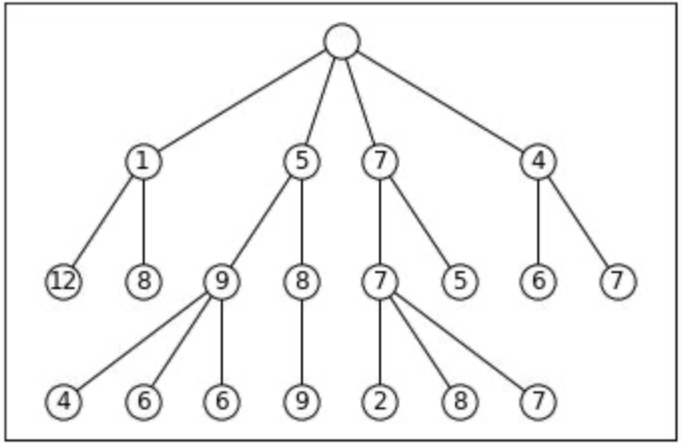
Ranking rewards + rule-based rewards



Tan et al. Reinforced Preference Optimization for Recommendation. arXiv:2510.12211.

Inference

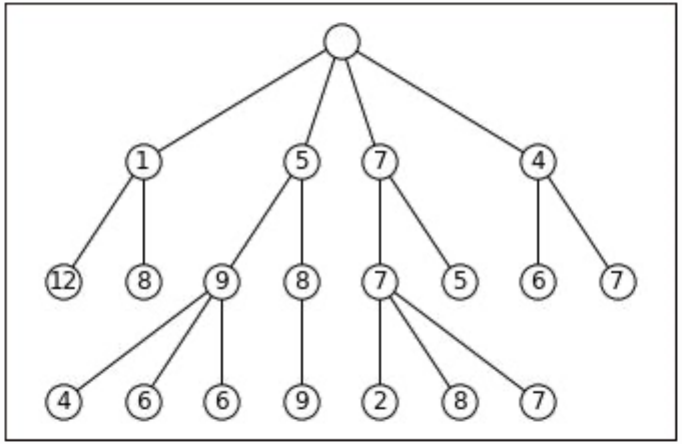
(1) Beam Search



Generating answers with the top-k highest scored beams

Inference

(1) Beam Search



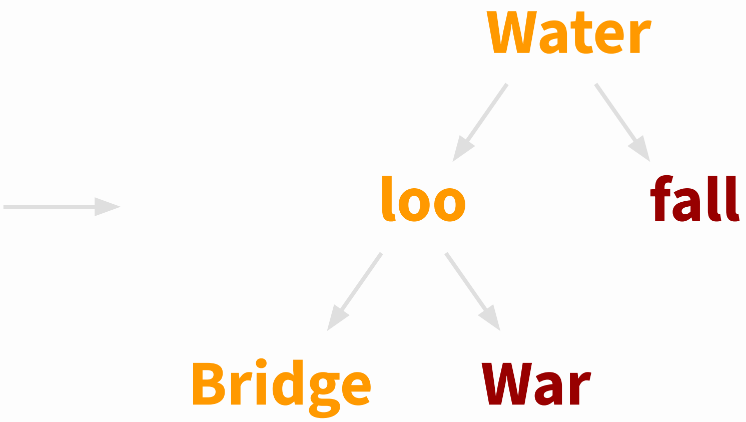
It may generate invalid items

How to ground the LLM outputs to real items?

Inference

(2) Constrained Beam Search

Valid items:
Waterloo Bridge, Waterfall Story, and Waterloo War



Constrained search tree

Inference

(2) Constrained Beam Search

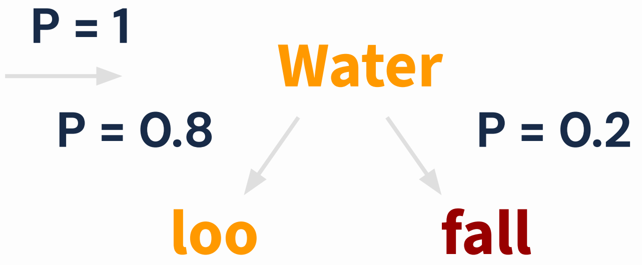
I have watched Titanic, Roman Holiday, …
Gone with the wind. Predict the next movie
I will watch:

P = 1 → **Water**

Inference

(2) Constrained Beam Search

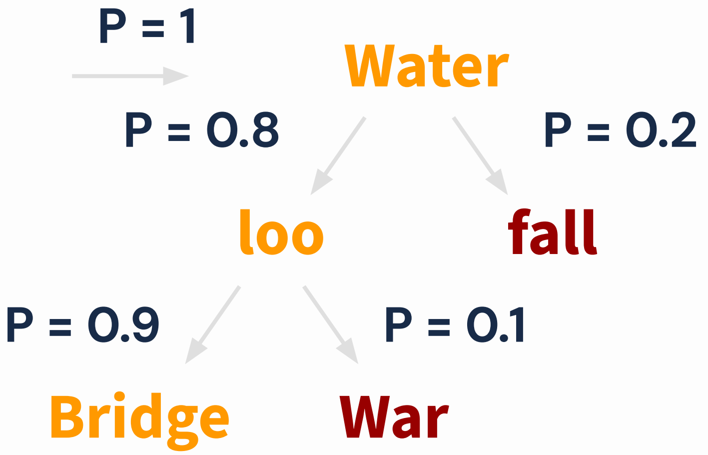
**I have watched Titanic, Roman Holiday, …
Gone with the wind. Predict the next movie
I will watch:**



Inference

(2) Constrained Beam Search

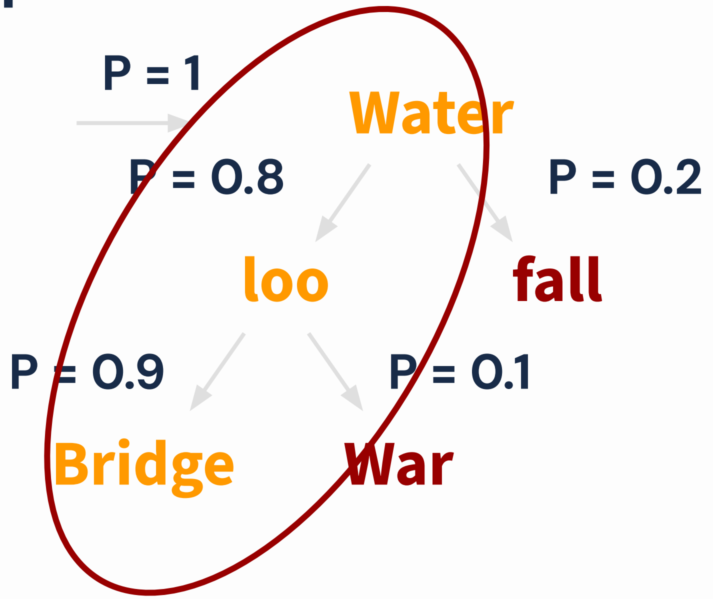
I have watched Titanic, Roman Holiday, … Gone with the wind. Predict the next movie I will watch:



Inference

(2) Constrained Beam Search

I have watched Titanic, Roman Holiday, … Gone with the wind. Predict the next movie I will watch:



Valid Item!

Inference

(3) Improved Constrained Beam Search (D3)

$$\mathcal{S}(h_{\leq t}) = \mathcal{S}(h_{\leq t-1}) + \log(p(h_t|x, h_{\leq t-1})),$$

$$\mathcal{S}(h) = \mathcal{S}(h)\boxed{/h_L^{\alpha},}$$

Length penalty in beam search;
Human does not like over long sentences.

Redundant for recommendation

Bao et al. Decoding Matters: Addressing Amplification Bias and Homogeneity Issue for LLM-based Recommendation. EMNLP 2024.

Inference

(3) Improved Constrained Beam Search (D3)

$$S(h_{\leq t}) = S(h_{\leq t-1}) + \log(p(h_t | x, h_{\leq t-1})),$$

$$S(h) = S(h)/\cancel{L},$$

Remove length penalty

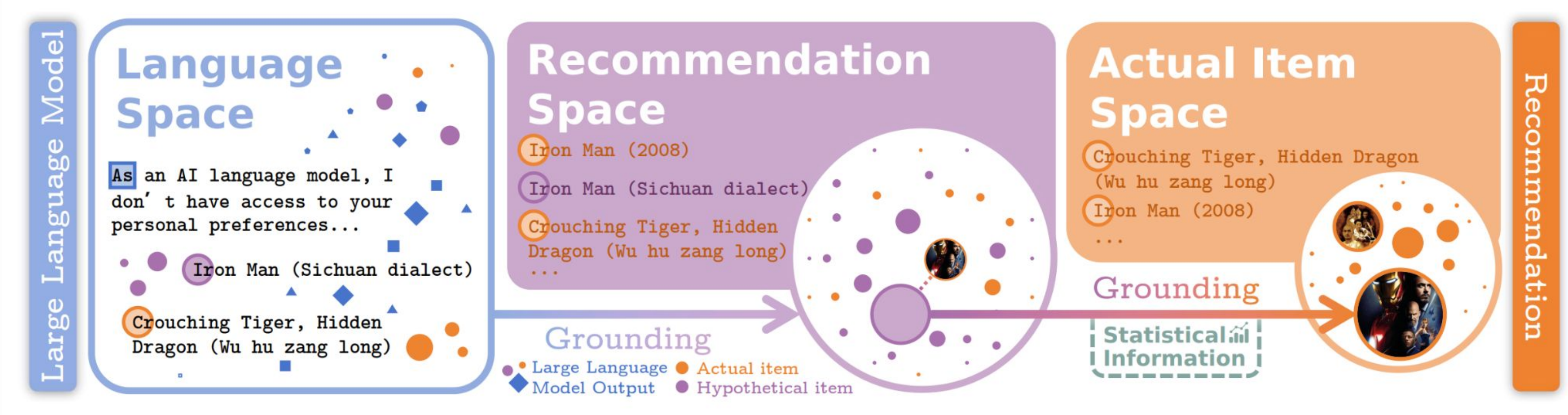
	Instruments	Books	CDs	Sports	Toys	Games
Baseline	0.1062	0.0308	0.0956	0.1171	0.0965	0.0610
D^3	**0.1111**	**0.0354**	**0.1190**	**0.1215**	**0.1025**	**0.0767**
- RLN	0.1093	0.0353	0.1000	0.1200	0.0975	0.0659
- TFA	0.1086	0.0309	0.1115	0.1192	0.1006	0.0732

Imp when removing

Bao et al. Decoding Matters: Addressing Amplification Bias and Homogeneity Issue for LLM-based Recommendation. EMNLP 2024.

Inference

(4) Dense Retrieval Grounding (BIGRec)



Retrieve real items by generated text

Part 1: LLM as Sequential Recommender

(1) Early efforts: using LLMs in a zero-shot setting

(2) Aligning LLMs for recommendation

(Multi-task) instruction tuning

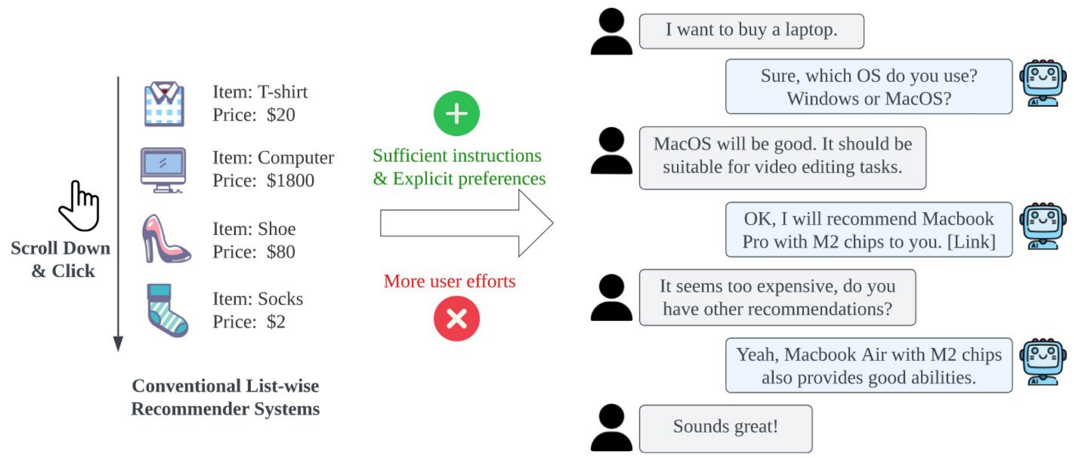
(3) Training objective: SFT, DPO, RL;

Inference: (constrained) beam search, retrieval;

Application 1:
LLM as Conversational Recommender

Conversational Recommender System (CRS)

- Recommendations with multiple turns conversation
- Interactive; engaging users in the loop



Chen et al. All Roads Lead to Rome: Unveiling the Trajectory of Recommender Systems Across the LLM Era. arXiv.2407.10081

Paradigms of CRS before the era of LLM

Features: Task-specific conversational recommenders, trained on limited conversation data.

Paradigms of CRS before the era of LLM

Features: Task-specific conversational recommenders, trained on limited conversation data.

- Lack of world knowledge.
- Requirement of complicated strategies.
- Lack of generalization capabilities.

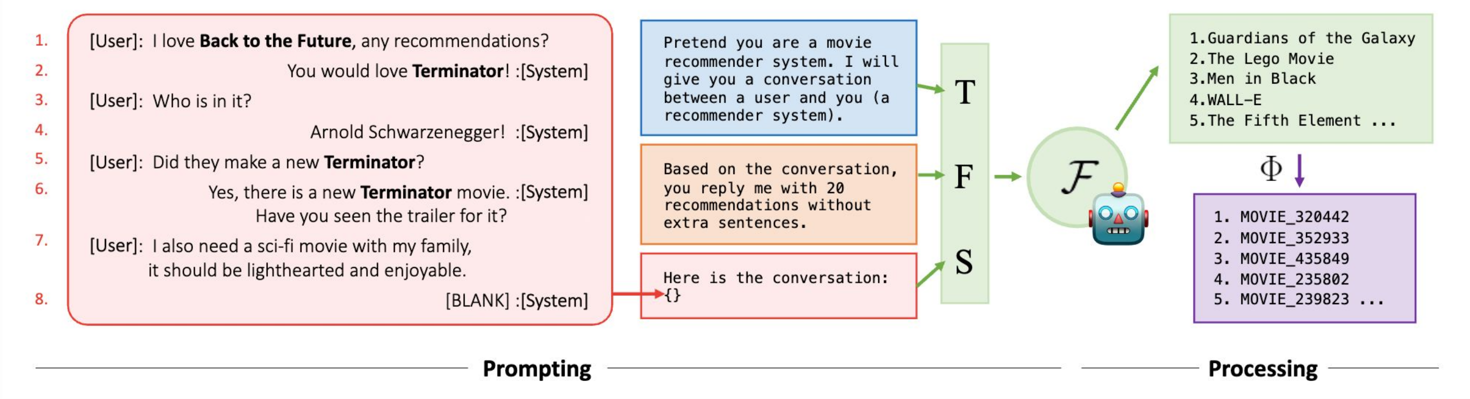
Example

LLM as conversational recommender

https://www.amazon.com/Rufus/

LLM as Conversational Recommender

LLMs as zero-shot CRS

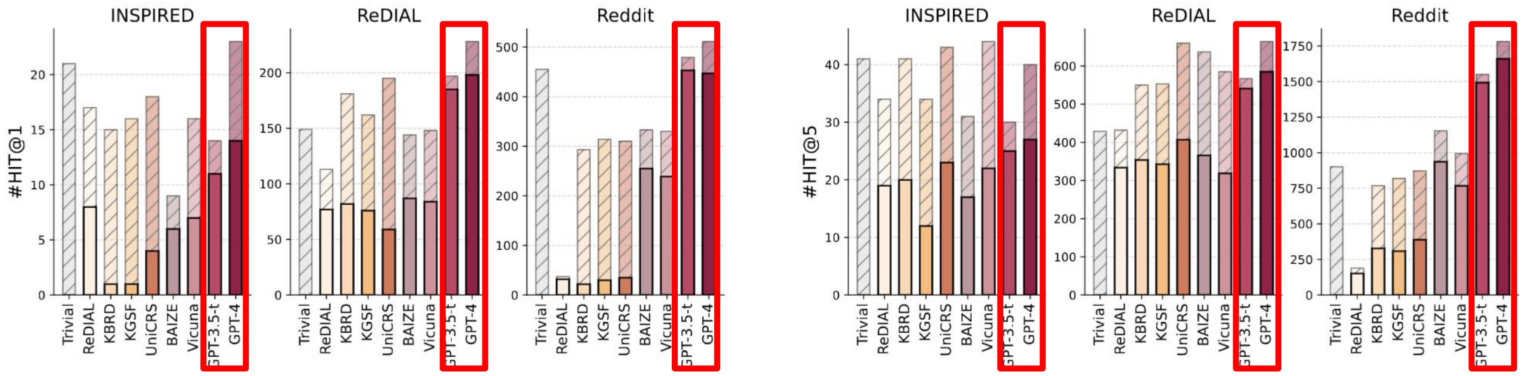


How powerful are LLMs for zero-shot CRS?

He et al. Large Language Models as Zero-Shot Conversational Recommenders. CIKM 2023.

LLM as Conversational Recommender

LLMs as zero-shot CRS

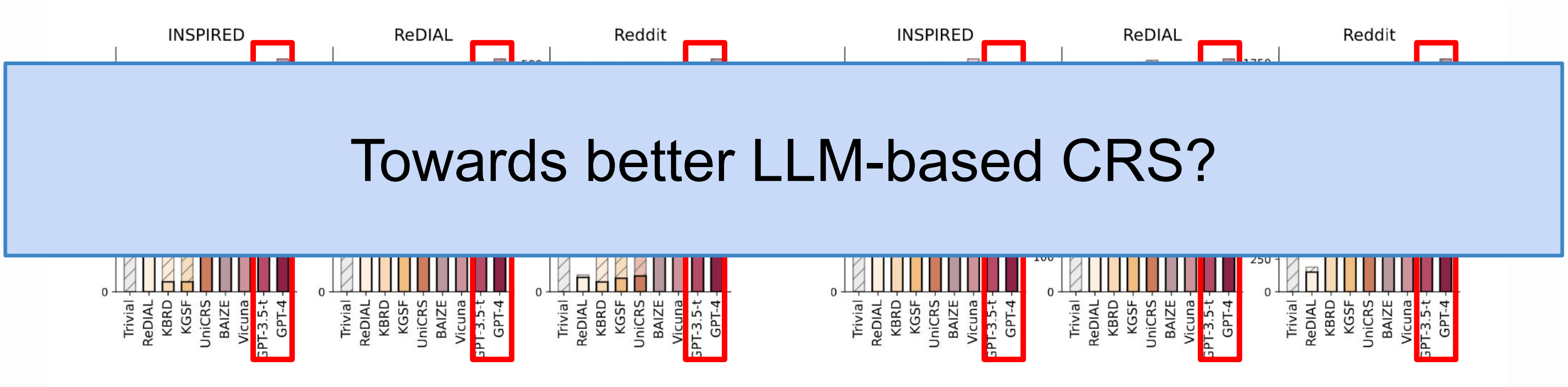


Can surpass traditional CRSs!

He et al. Large Language Models as Zero-Shot Conversational Recommenders. CIKM 2023.

LLM as Conversational Recommender

LLMs as zero-shot CRS

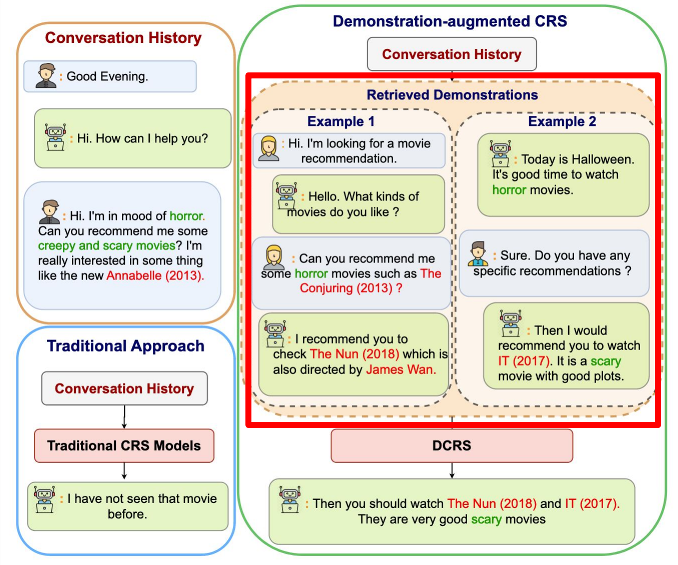


Can surpass traditional CRSs!

He et al. Large Language Models as Zero-Shot Conversational Recommenders. CIKM 2023.

LLM as Conversational Recommender

+ Demonstration



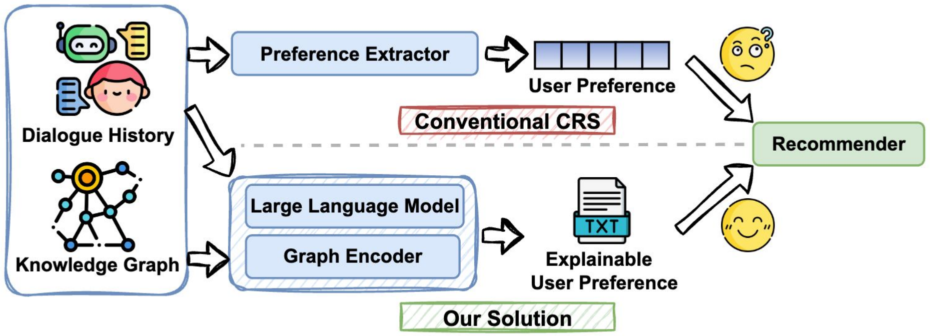
Prompting with previously successful conversation

Relevant conversation history helps!

Dao et al. Broadening the View: Demonstration-augmented Prompt Learning for Conversational Recommendation. SIGIR 2024.

LLM as Conversational Recommender

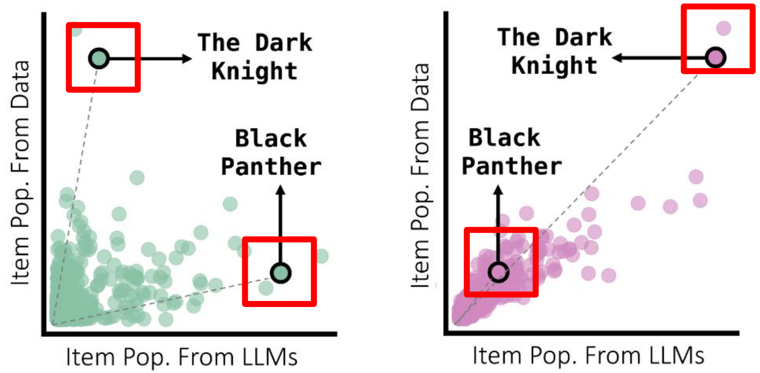
+ **Knowledge graph**



Recommendation-specific knowledge graph helps

LLM as Conversational Recommender

+ Collaborative information



(a) Before *RTA* **(b) After *RTA***

Collaborative information (e.g., popularity) helps LLMs fit the real distribution in CRS

He et al. Reindex-Then-Adapt: Improving Large Language Models for Conversational Recommendation. WSDM 2025.

LLM as Conversational Recommender

Challenges – Datasets

Public datasets for CRS are limited, due to the scarcity of conversational products and real-world CRS datasets

LLM as Conversational Recommender

Challenges – Evaluation

Traditional metrics like NDCG and BLEU are often insufficient to assess user experience

LLM as Conversational Recommender

Challenges – Product

What is the form of LLM-based CRS products?

ChatBot? Search bar? Independent App?

Application 1:
LLM as Conversational Recommender

(1) LLMs are promising backbone models for CRS

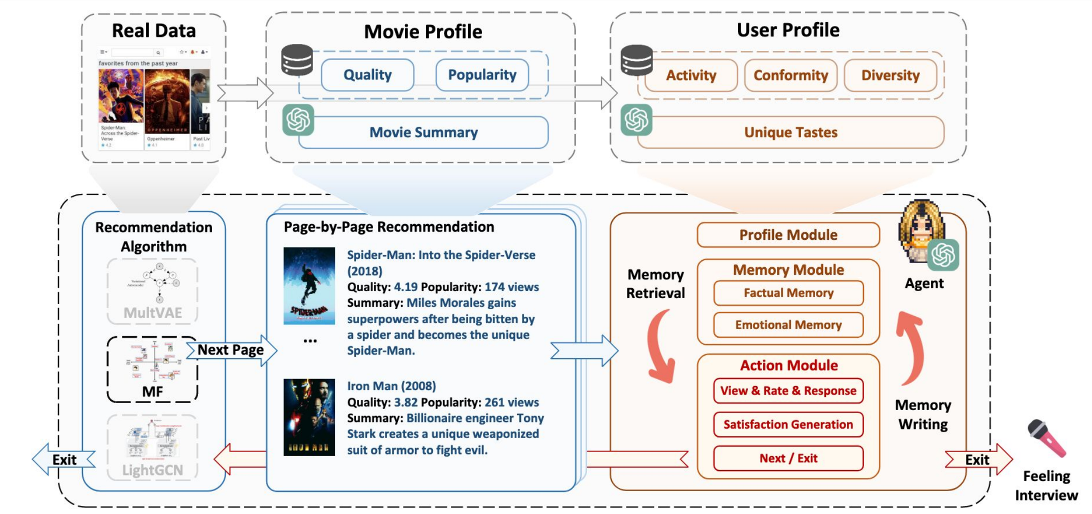
(2) Challenges in LLM-based CRSs:

dataset, evaluation, and product

Application 2:
LLM as User Simulator

LLM as User Simulator

Generative agents for recommendation



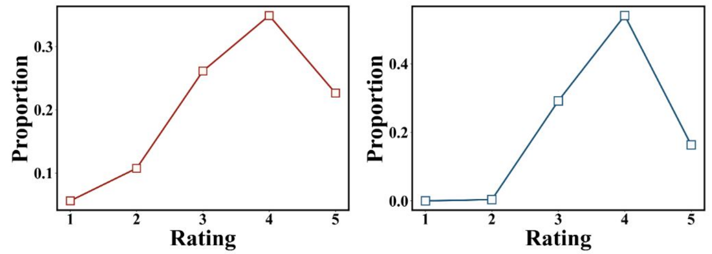
Realworld-like simulation paradigm

- 1000 users
- Page-by-page simulation

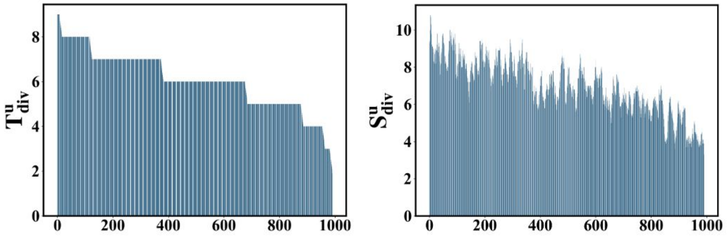
Zhang et al. On generative agents in recommendation. SIGIR 2024.

LLM as User Simulator

Generative agents for recommendation



(a) Distribution on MovieLens (b) Agent-simulated distribution



(a) Ground-truth diversity (b) Simulated diversity

Aligned user preferences
& Recommender evaluation

Table 2: Recommendation strategies evaluation.

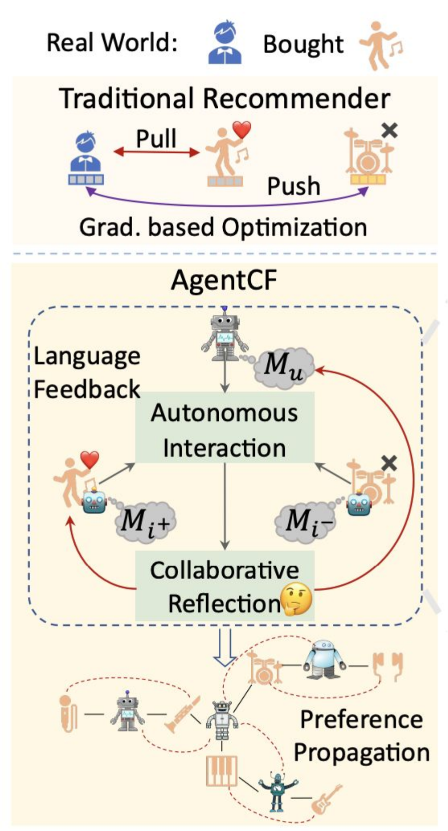
	$\overline{P}_{view}$	$\overline{N}_{like}$	$\overline{P}_{like}$	$\overline{N}_{exit}$	$\overline{S}_{sat}$
Random	0.312	3.3	0.269	2.99	2.93
Pop	0.398	4.45	0.360	3.01	3.42
MF	0.488	**6.07***	0.462	**3.17***	3.80
MultVAE	0.495	5.69	0.452	3.10	3.75
LightGCN	**0.502***	5.73	**0.465***	3.02	**3.85***

Zhang et al. On generative agents in recommendation. SIGIR 2024.

LLM as User Simulator

AgentCF

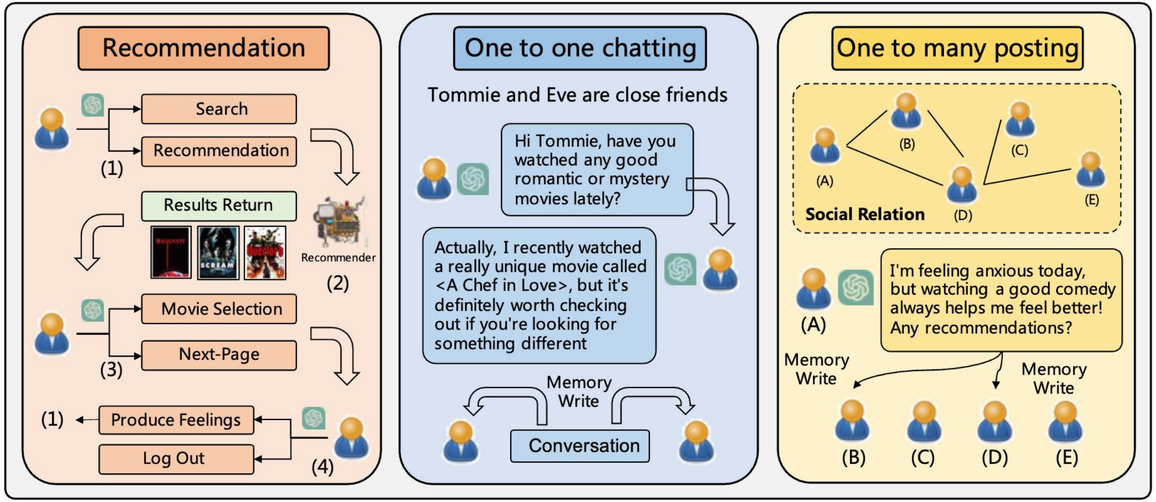
- Agents for both users and items
- Co-optimized by real collaborative filtering signals (user-item interactions)
- Memory updated by reflection



Zhang et al. AgentCF: Collaborative Learning with Autonomous Language Agents for Recommender Systems. WWW 2024.

LLM as User Simulator

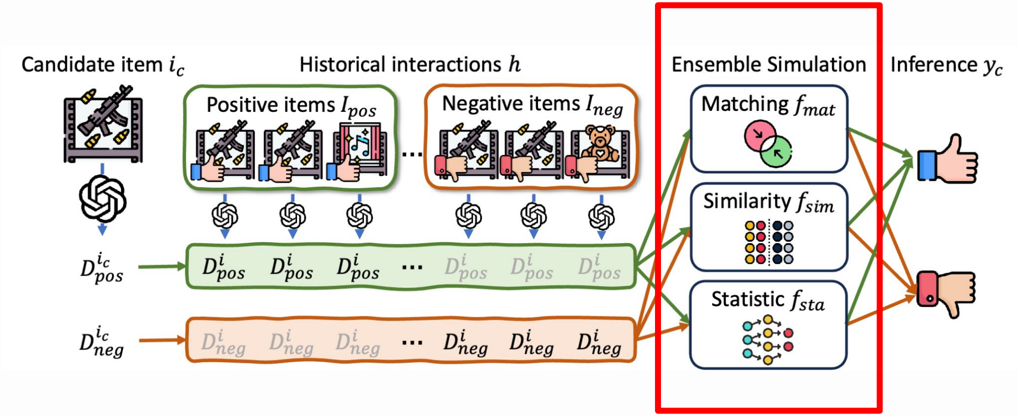
+ ## Social behaviors



Recommendation
Chat
Networking

LLM as User Simulator

+ Multi-facet simulation objective



Category matching
Fine-grained similarity
Statistic information

LLM-based Generative Rec

(1) Tokenize actions by text

Pros: distribution naturally aligned with LLMs

Cons: inefficient

(2) From zero-shot to instruction tuning

Training objectives: SFT, DPO, RL, …

Inference: constrained beam search, retrieval

(3) Applications

Conversational RS, User Simulator