RUTGERS

香港浸會大學
HONG KONG BAPTIST UNIVERSITY

Tutorial on Large Language Models for Recommendation





Wenyue Hua
Rutgers University
wenyue.hua@rutgers.edu

Lei Li
Hong Kong Baptist University
csleili@comp.hkbu.edu.hk

Shuyuan Xu
Rutgers University
shuyuan.xu@rutgers.edu

Li Chen
Hong Kong Baptist University
lichen@comp.hkbu.edu.hk

Yongfeng Zhang
Rutgers University
yongfeng.zhang@rutgers.edu

RUTGERS

Outline

- Background and Introduction
- Large Language Models for Recommendation
- Trustworthy LLMs for Recommendation
- Hands-on Demo of LLM-RecSys Development based on OpenP5
- Summary

Recommender Systems are Everywhere

- Influence our daily life by providing personalized services

E-commerce

Social Networks

News Feeding

Search Engine

Navigation

Travel Planning

Professional Networks

Healthcare

Online Education

Technical Advancement of Recommender Systems

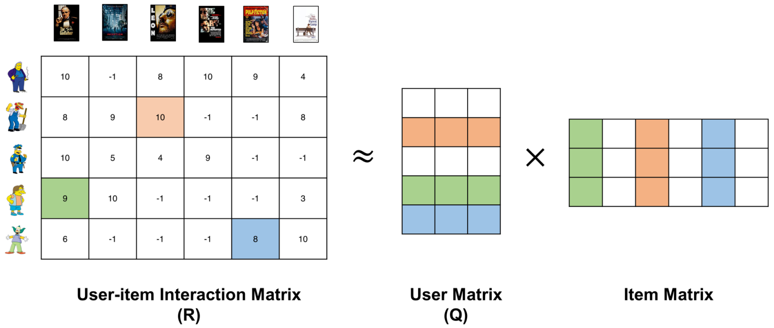
- From Shallow Model, to Deep Model, and to Large Model

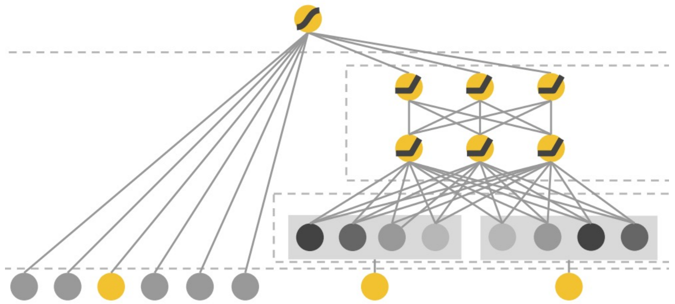
Shallow Models → Deep Models → Large Models

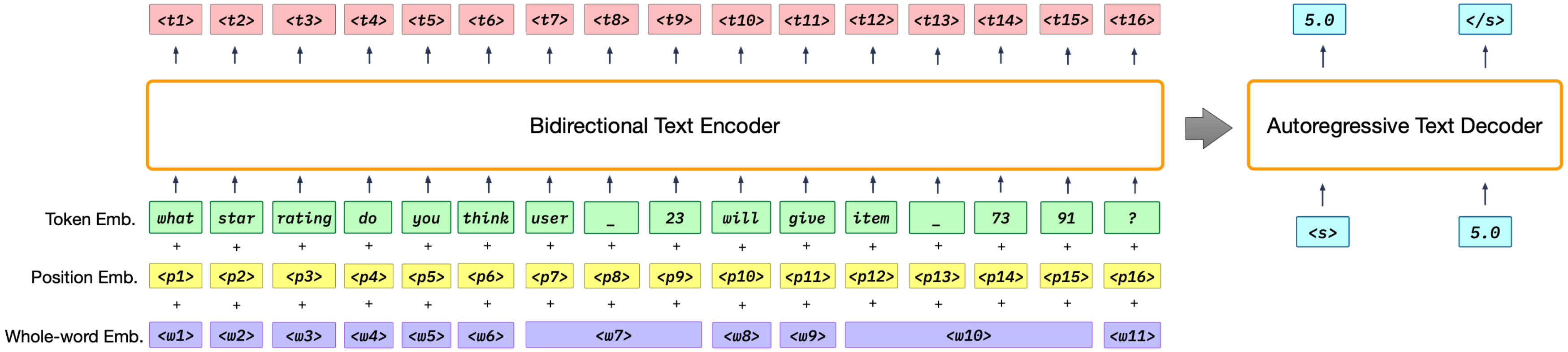
e.g. Matrix Factorization [1]

e.g. Deep & Wide NN [2]

e.g. P5 [3]







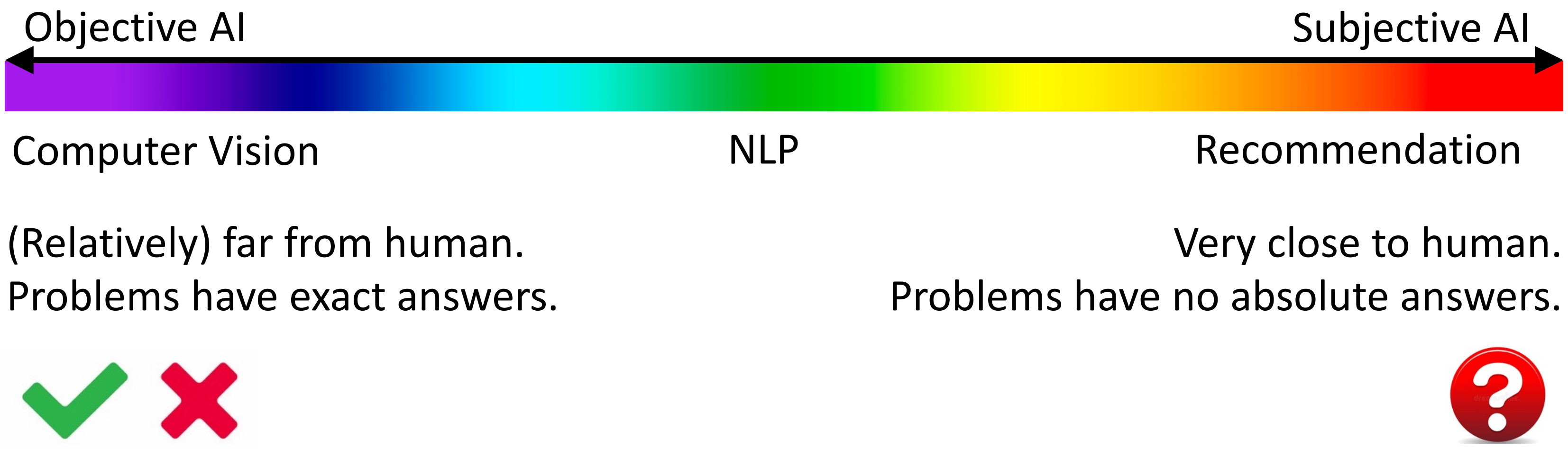
[1] Koren, Yehuda, Robert Bell, and Chris Volinsky. "Matrix factorization techniques for recommender systems." Computer 42, no. 8 (2009): 30-37.

[2] Cheng, Heng-Tze, Levent Koc, Jeremiah Harmsen, Tal Shaked, Tushar Chandra, Hrishi Aradhye, Glen Anderson et al. "Wide & deep learning for recommender systems." DLRS 2016.

[3] Geng, Shijie, Shuchang Liu, Zuohui Fu, Yingqiang Ge, and Yongfeng Zhang. "Recommendation as Language Processing (RLP): A Unified Pretrain, Personalized Prompt & Predict Paradigm (P5)." RecSys 2022.

Objective AI vs. Subjective AI

- Recommendation is unique in the AI family
  - Recommendation is most close to human among all AI tasks
  - Recommendation is a very representative Subjective AI
  - Thus, leads to many unique challenges in recommendation research



RUTGERS

Computer Vision: (mostly) Objective AI Tasks

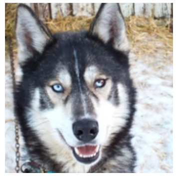
Objective AI ⟷ Subjective AI

Computer Vision | NLP | Recommendation

Image Classification

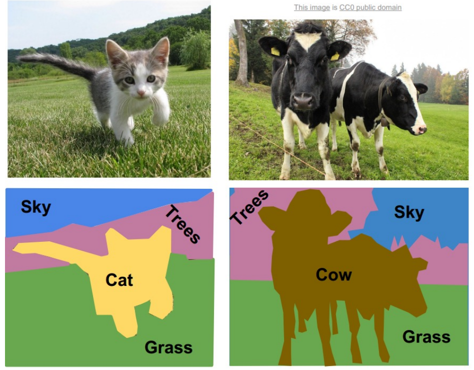
cat

dog

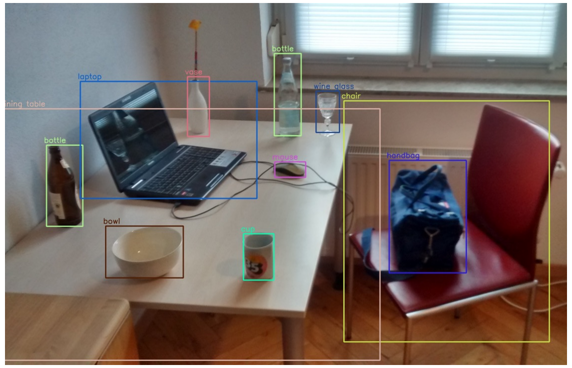


Husky like a wolf

Image Segmentation

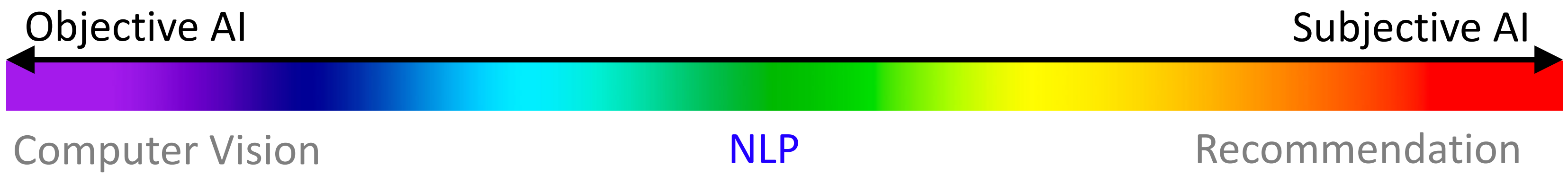


Object Detection

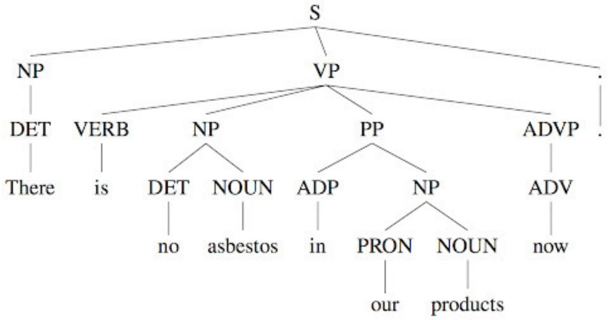


RUTGERS

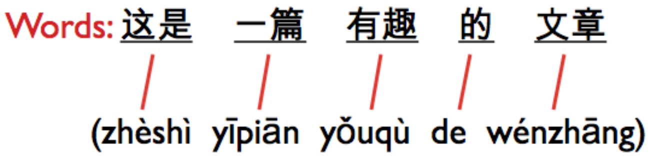
NLP: partly Objective, partly Subjective



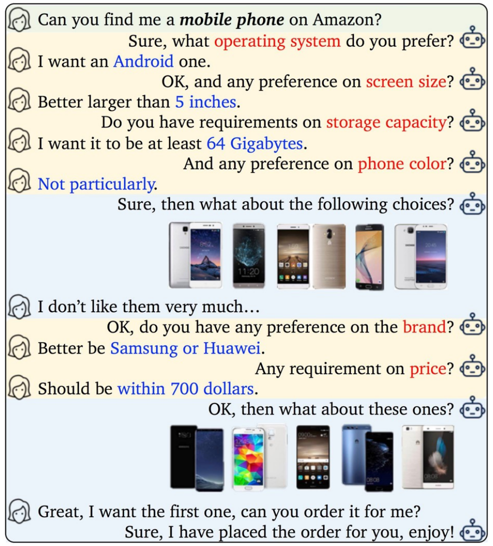
Syntactic Analysis



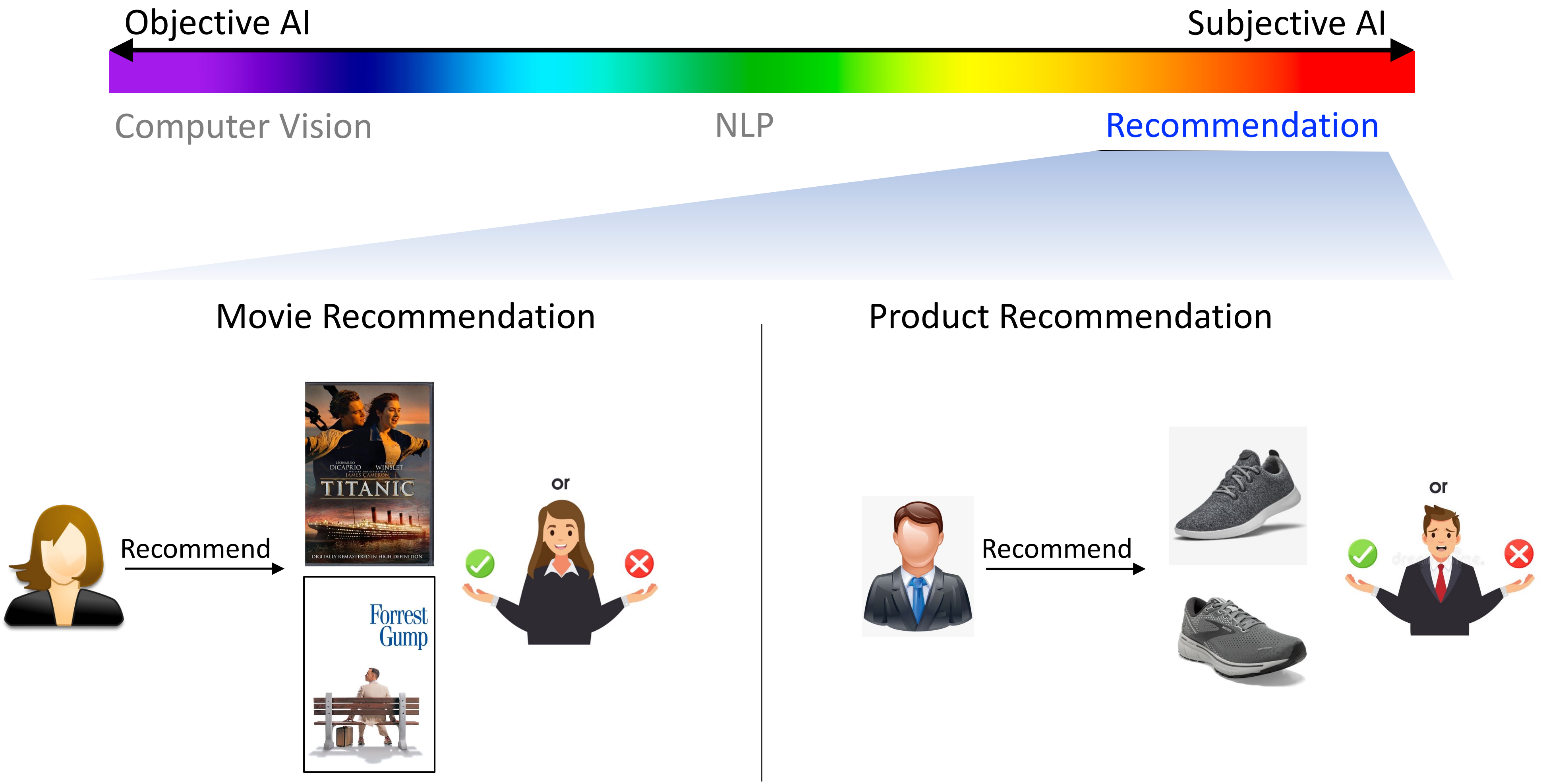
Word Segmentation



Dialog Systems



Recommendation: mostly Subjective AI Tasks



Recommendation is not only about Item Ranking

- A diverse set of recommendation tasks
  - Rating Prediction
  - Item Ranking
  - Sequential Recommendation
  - User Profile Construction
  - Review Summarization
  - Explanation Generation
  - Fairness Consideration
  - etc.

Example: Subjective AI needs Explainability

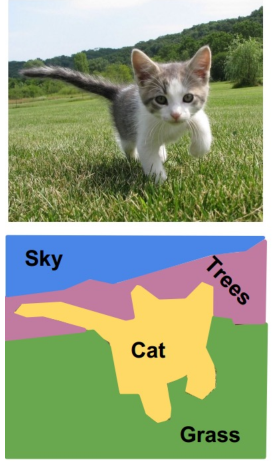
- Objective vs. Subjective AI on Explainability

Objective AI

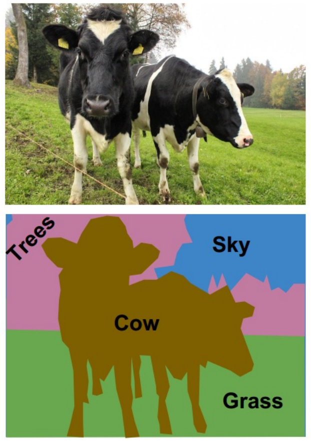
Human can directly identify if the AI-produced result is right or wrong

cat

dog

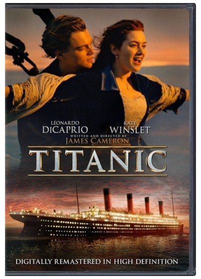


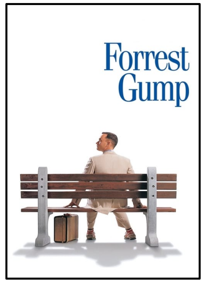
This image is CC0 public domain



Subjective AI

Human can hardly identify if the AI-produced result is right or wrong. Users are very vulnerable, could be manipulated, utilized or even cheated by the system





or

Can you find me a ***mobile phone*** on Amazon?
Sure, what operating system do you prefer?
I want an Android one.
OK, and any preference on screen size?
Better larger than 5 inches.
Do you have requirements on storage capacity?
I want it to be at least 64 Gigabytes.
And any preference on phone color?
Not particularly.
Sure, then what about the following choices?

I don't like them very much…
OK, do you have any preference on the brand?
Better be Samsung or Huawei.
Any requirement on price?
Should be within 700 dollars.
OK, then what about these ones?

Great, I want the first one, can you order it for me?
Sure, I have placed the order for you, enjoy!

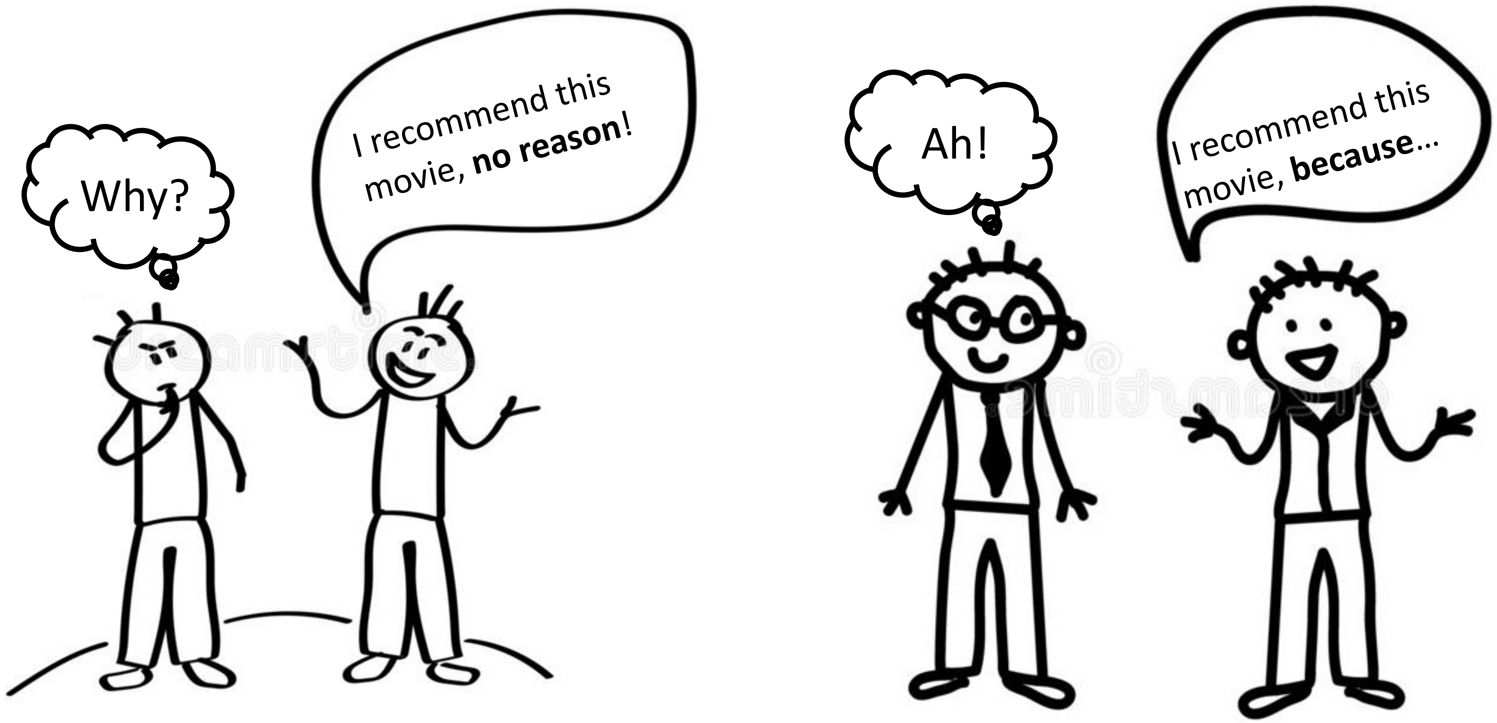
Nothing is definitely right or wrong.

Highly subjective, and usually personalized.

RUTGERS

Example: Subjective AI needs Explainability

- In many cases, it doesn't matter what you recommend, but how you explain your recommendation
- How do humans make recommendation?



Can we Handle all RecSys tasks Together?

- A diverse set of recommendation tasks
  - Rating Prediction
  - Item Ranking
  - Sequential Recommendation
  - User Profile Construction
  - Review Summarization
  - Explanation Generation
  - Fairness Consideration
  - etc.
- Do we really need to design thousands of recommendation models?
  - Difficult to integrate so many models in industry production environment

A Bird's View of Traditional RecSys

- The Multi-Stage Filtering RecSys Pipeline

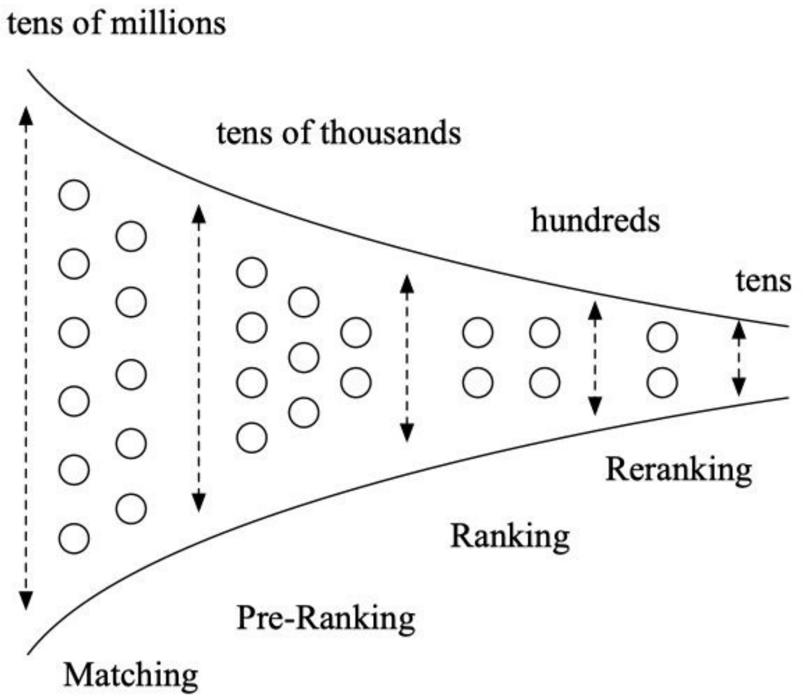
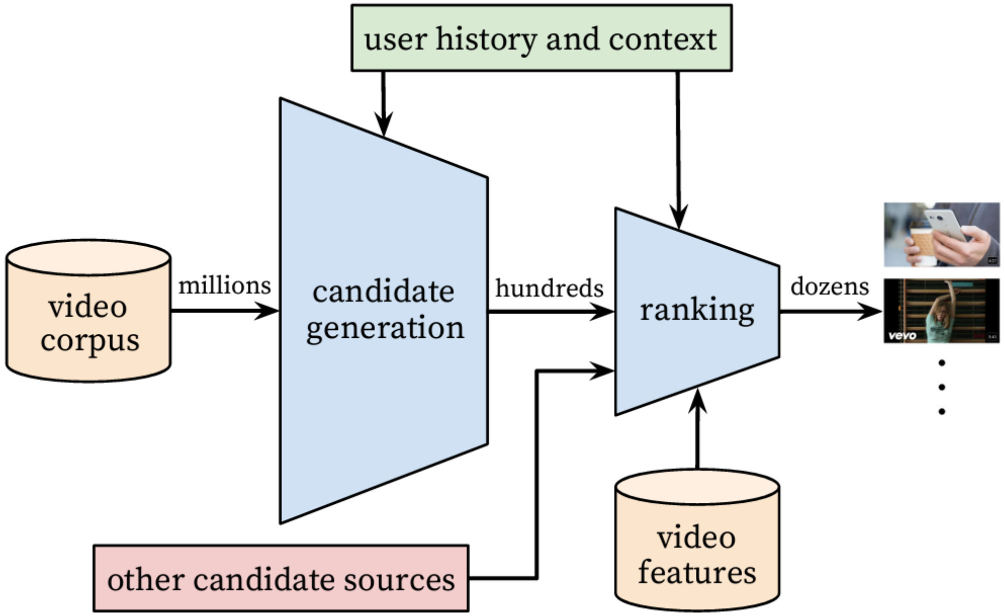


Image credit to [1]



Youtube recommendation engine

Image credit to [2]

[1] Jiang, Biye, Pengye Zhang, Rihan Chen, Xinchen Luo, Yin Yang, Guan Wang, Guorui Zhou, Xiaoqiang Zhu, and Kun Gai. "DCAF: A Dynamic Computation Allocation Framework for Online Serving System." DLP-KDD 2020.
[2] Covington, Paul, Jay Adams, and Emre Sargin. "Deep neural networks for youtube recommendations." In *Proceedings of the 10th ACM conference on recommender systems*, pp. 191-198. 2016.

Discriminative Ranking

- User-item matching based on embeddings

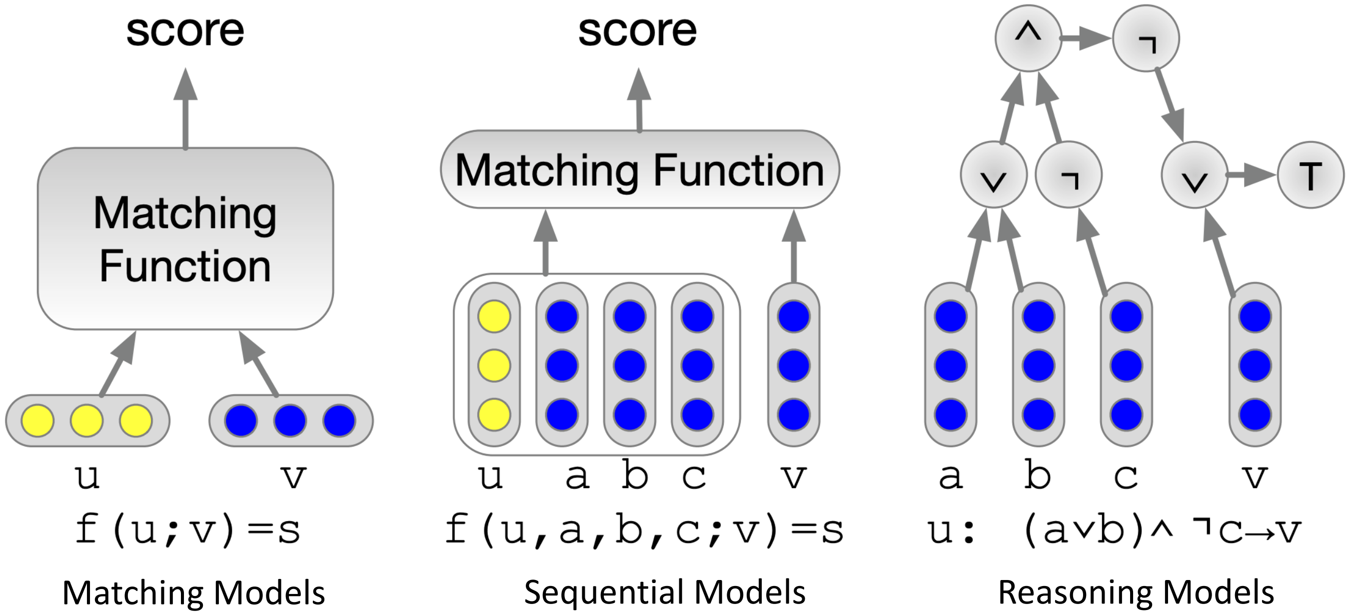


Image credit to [1]

- Discriminative ranking loss function
  - e.g., Bayesian Personalized Ranking (BPR) loss

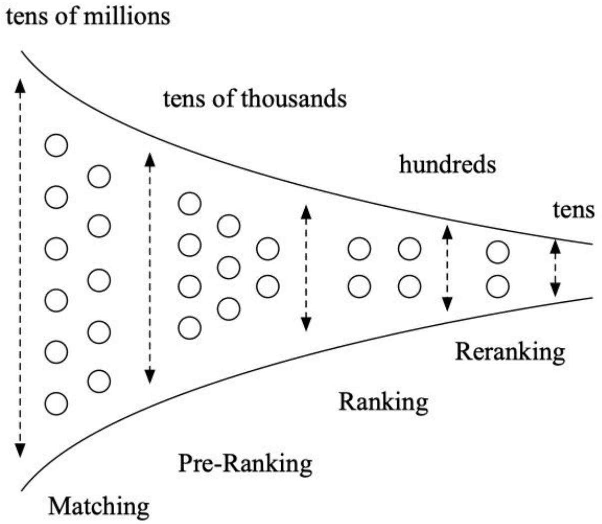
$$maximize \sum_{(u,i,j)\in D_S} \ln \sigma(\hat{x}_{uij}) - \lambda_\Theta ||\Theta||^2 \qquad where: \hat{x}_{uij} = p_u q_i^T - p_u q_j^T$$

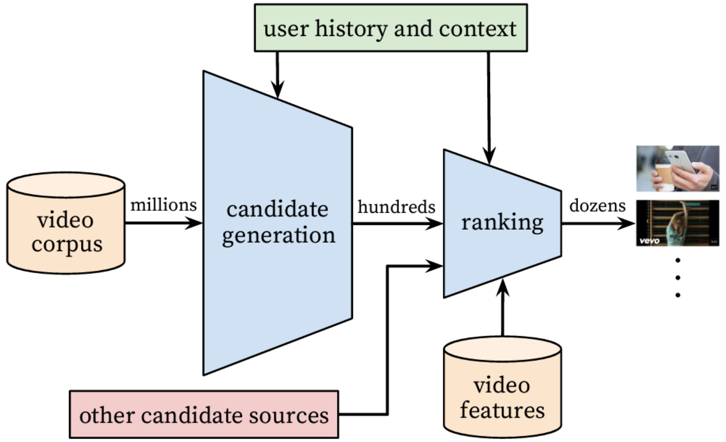
[1] Chen, Hanxiong, Shaoyun Shi, Yunqi Li, and Yongfeng Zhang. "Neural collaborative reasoning." In Proceedings of the Web Conference 2021, pp. 1516-1527. 2021.

RUTGERS

Problem with Discriminative Ranking

- Huge numbers of users and items
  - Amazon: 300 million customers, 350 million products*
  - YouTube: 2.6+ billion monthly active users, 5+ billion videos**
  - We have to use multi-stage filtering: Simple rules are used at early stages, advanced algorithms are only applied to a small number of items at later stages



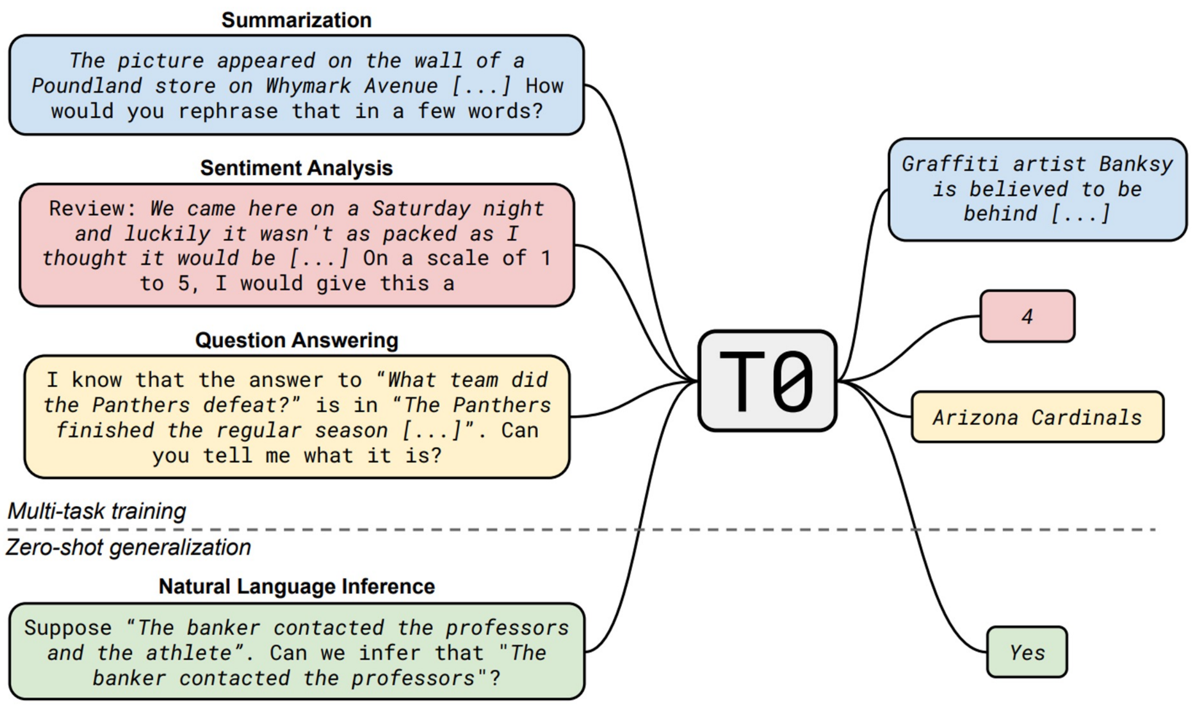


- Too many candidate items, difficult for evaluation
  - Many research papers use sampled evaluation: 1-in-100, 1-in-1000, etc.

*https://sell.amazon.com/blog/amazon-stats, and https://www.bigcommerce.com/blog/amazon-statistics/
**https://www.globalmediainsight.com/blog/youtube-users-statistics/

Large Language Models (LLMs)

- Auto-regressive decoding for generative prediction



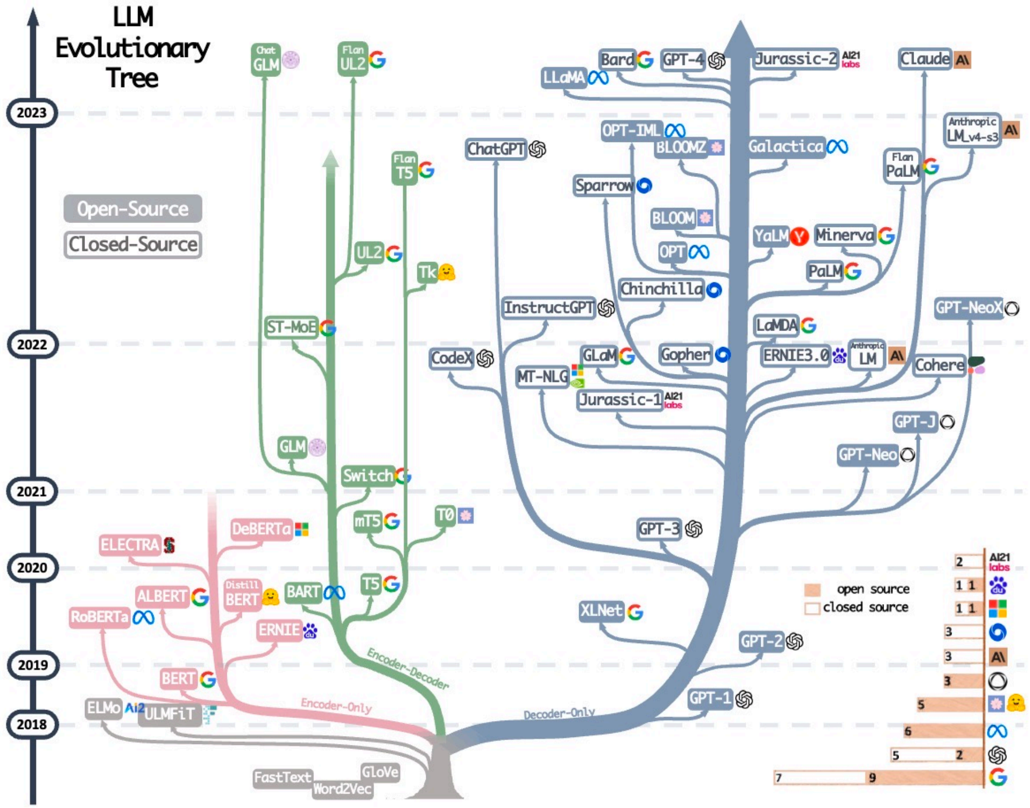


Image credit to [1]

Image credit to [2]

[1] Sanh, Victor, Albert Webson, Colin Raffel, Stephen H. Bach, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin et al. "Multitask prompted training enables zero-shot task generalization." ICLR 2022.

[2] Yang, Jingfeng, Hongye Jin, Ruixiang Tang, Xiaotian Han, Qizhang Feng, Haoming Jiang, Bing Yin, and Xia Hu. "Harnessing the Power of LLMs in Practice: A Survey on ChatGPT and Beyond." arXiv preprint arXiv:2304.13712 (2023).

Generative Pre-training and Prediction

- Generative Pre-training
  - Generative Loss Function
    - Use the previous tokens to predict next token

$$L_1(\mathcal{U}) = \sum_i \log P(u_i | u_{i-k}, \ldots, u_{i-1}; \Theta)$$

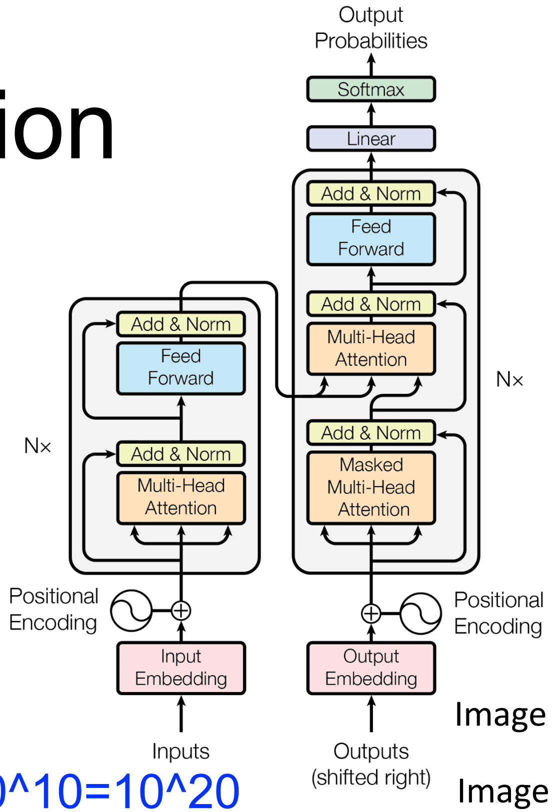
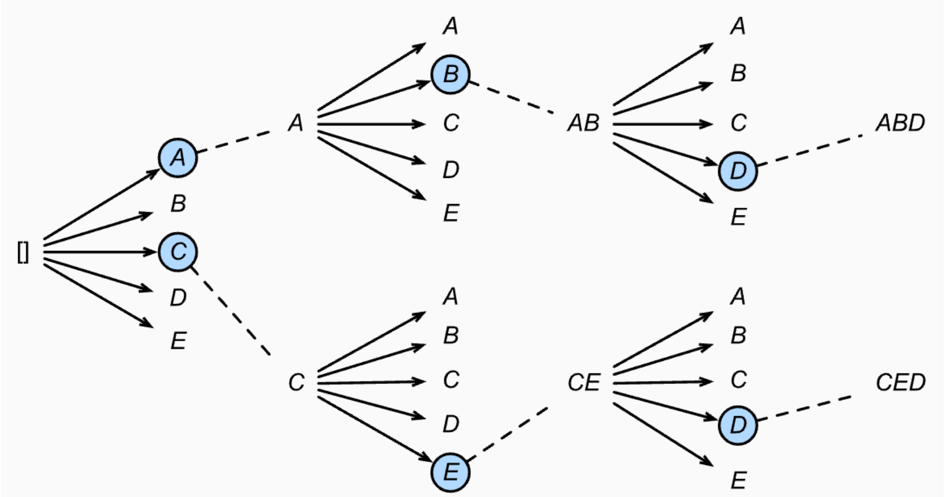


Image credit to [1]

- Generative Prediction
  - Beam Search
    - Using finite tokens to represent (almost) infinite items
      - e.g., 100 vocabulary tokens, ID size 10 => #items = 100^10=10^20

Image credit to [2]

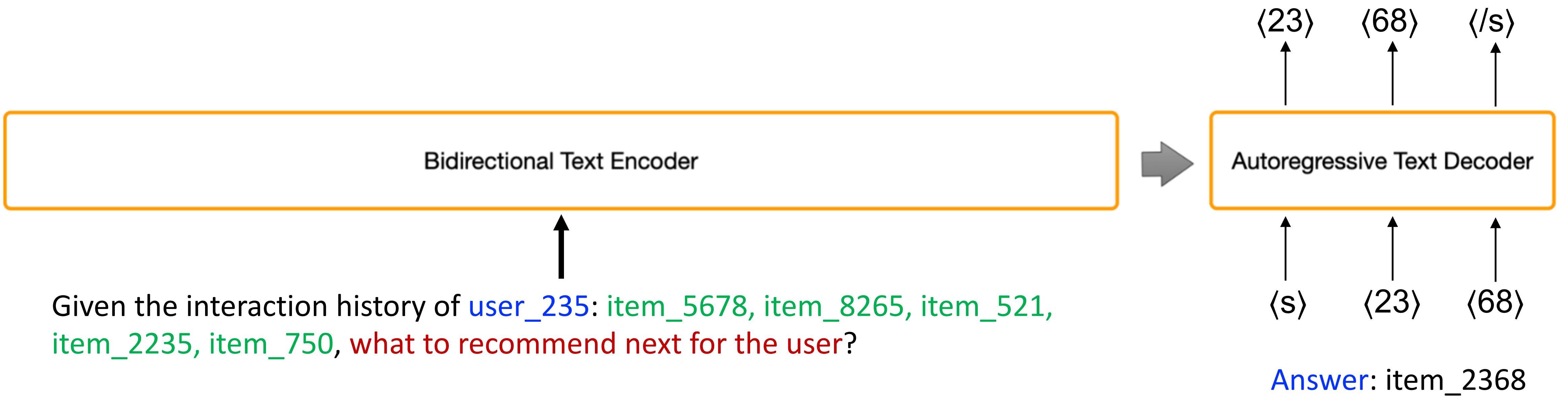


    - # of candidate tokens at each beam is bounded
    - No longer need one-by-one candidate score calculation as in discriminative ranking
    - Directly generate the item ID to recommend

[1] Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. "Attention is all you need." Advances in neural information processing systems 30 (2017).
[2] https://d2l.ai/chapter_recurrent-modern/beam-search.html

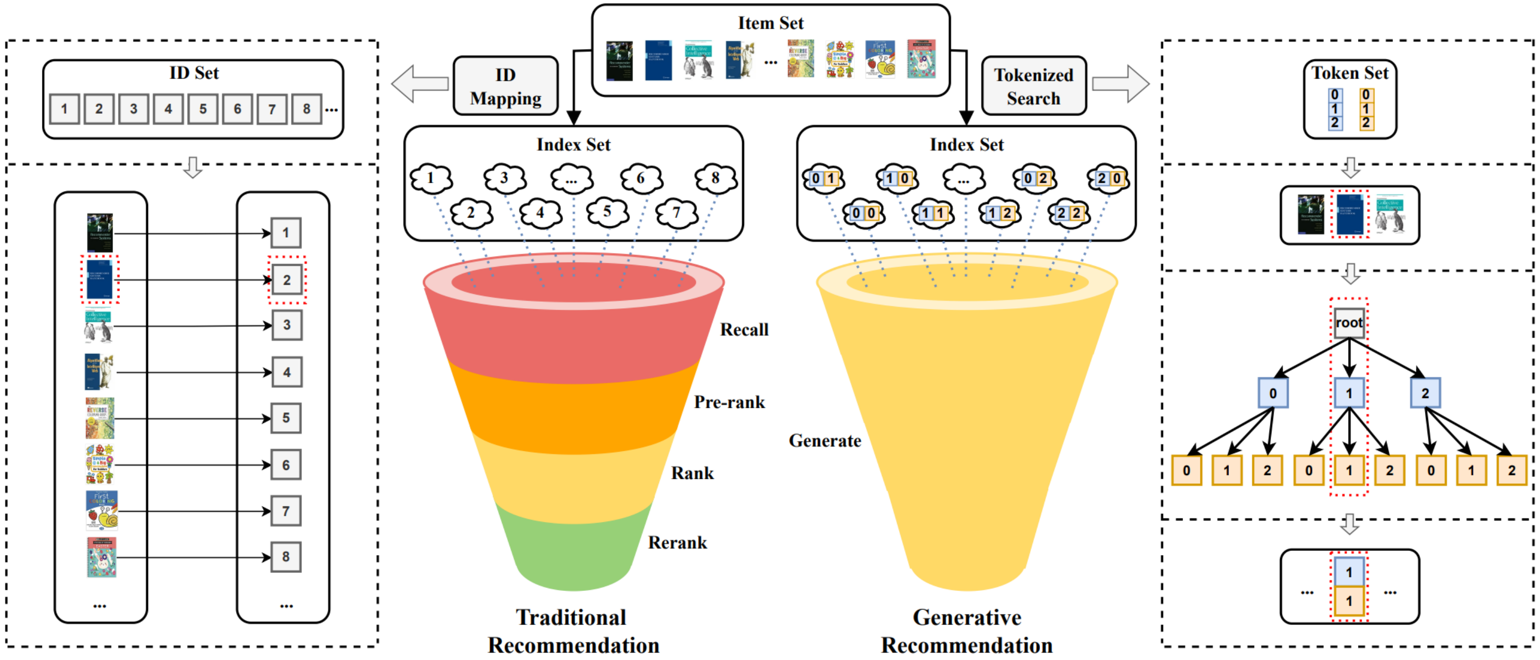
Generative Ranking

- From Multi-stage ranking to Single-stage ranking
  - The model automatically considers all items as the candidate pool
  - Fixed-size item decoding
    - e.g., using 100 tokens ⟨00⟩⟨01⟩…⟨99⟩ for item ID representation



Generative Recommendation with Beam Search

- Since item IDs are tokenized (e.g., [“item”, “_”, “73”, “91”]), beam search is bounded on width
  - E.g., 100 tokens width: ⟨00⟩, ⟨01⟩, ⟨02⟩, …, ⟨98⟩, ⟨99⟩
- Assigning an item a token as in traditional recommendation is infeasible for LLM
  - Consume a lot of memory and computationally expensive



[1] Li, Lei, Yongfeng Zhang, Dugang Liu, and Li Chen. "Large Language Models for Generative Recommendation: A Survey and Visionary Discussions." *arXiv preprint arXiv:2309.01157* (2023).

Large Language Models for Recommendation

RUTGERS

How to Categorize LLM-based Recommendation

- Whether to Fine-tune LLM for Recommendation or Not
  - With Fine-tuning [1]
  - Without Fine-tuning [2]
- The Role of LLM in Recommendation
  - LLM as RecSys [1]
  - LLM in RecSys [3]
    - e.g., LLM as a feature extractor for recommender systems
  - RecSys in LLM [4]
    - e.g., LLM-based Agents, where RecSys is used as on of the tools
- Typical Recommendation Tasks [1]
  - Rating Prediction, Sequential Recommendation, Direct Recommendation, ...

[1] Geng, Shijie, Shuchang Liu, Zuohui Fu, Yingqiang Ge, and Yongfeng Zhang. "Recommendation as Language Processing (RLP): A Unified Pretrain, Personalized Prompt & Predict Paradigm (P5)" RecSys 2022.
[2] Liu, Junling, Chao Liu, Renjie Lv, Kang Zhou, and Yan Zhang. "Is chatgpt a good recommender? a preliminary study." *arXiv preprint arXiv:2304.10149* (2023).
[3] Lin, Jianghao, et al. "How Can Recommender Systems Benefit from Large Language Models: A Survey." *arXiv preprint arXiv:2306.05817* (2023).
[4] Wang, Yancheng, Ziyan Jiang, Zheng Chen, Fan Yang, Yingxue Zhou, Eunah Cho, Xing Fan, Xiaojiang Huang, Yanbin Lu, and Yingzhen Yang. "Recmind: Large language model powered agent for recommendation." *arXiv preprint arXiv:2308.14296* (2023).

Two Broad Categories of Recommendation Tasks

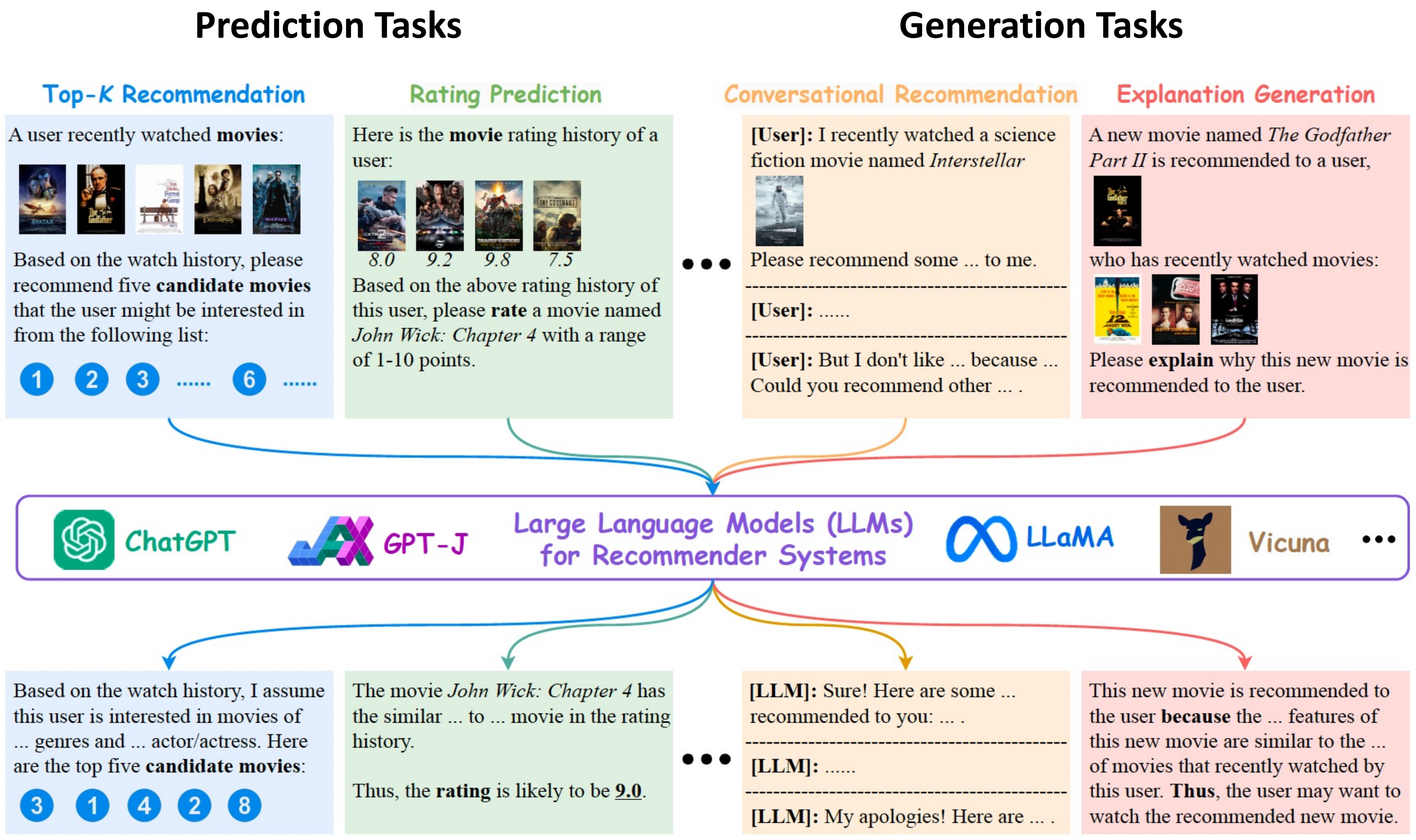


Image credit to [1]

[1] Fan, Wenqi, et al. "Recommender systems in the era of large language models (llms)." *arXiv preprint arXiv:2307.02046* (2023).

Rutgers

Typical Recommendation Tasks

- LLM usually can perform multiple recommendation tasks
  - e.g., P5 [2], POD [3], InstructRec [4]

Rating Prediction	**Top-N Recommendation**	**Sequential Recommendation**	**Explainable Recommendation**	**Review Generation**	**Review Summarization**	**Conversational Recommendation**
P5 [Geng *et al.*, 2022c], BookGPT [Zhiyuli *et al.*, 2023], LLMRec [Liu *et al.*, 2023b], RecMind [Wang *et al.*, 2023b], [Liu *et al.*, 2023a; Dai *et al.*, 2023]	P5 [Geng *et al.*, 2022c], UP5 [Hua *et al.*, 2023a], VIP5 [Geng *et al.*, 2023], OpenP5 [Xu *et al.*, 2023], POD [Li *et al.*, 2023c], GPTRec [Petrov and Macdonald, 2023], LLMRec [Liu *et al.*, 2023b], RecMind [Wang *et al.*, 2023b], [Zhang *et al.*, 2023a; Zhang *et al.*, 2023b; Liu *et al.*, 2023a; Li *et al.*, 2023f; Dai *et al.*, 2023]	P5 [Geng *et al.*, 2022c], UP5 [Hua *et al.*, 2023a], VIP5 [Geng *et al.*, 2023], OpenP5 [Xu *et al.*, 2023], POD [Li *et al.*, 2023c], GenRec [Ji *et al.*, 2023], GPTRec [Petrov and Macdonald, 2023], LMRecSys [Zhang *et al.*, 2021], NIR [Wang and Lim, 2023], PALR [Chen, 2023], LLMRec [Liu *et al.*, 2023b], RecMind [Wang *et al.*, 2023b], BIGRec [Bao *et al.*, 2023a], [Hua *et al.*, 2023b; Liu *et al.*, 2023a; Hou *et al.*, 2023b; Zhang *et al.*, 2023b]	P5 [Geng *et al.*, 2022c], VIP5 [Geng *et al.*, 2023], POD [Li *et al.*, 2023c], PEPLER [Li *et al.*, 2023b], M6-Rec [Cui *et al.*, 2022], LLMRec [Liu *et al.*, 2023b], RecMind [Wang *et al.*, 2023b], KnowRec [Colas *et al.*, 2023], [Liu *et al.*, 2023a]	-	P5 [Geng *et al.*, 2022c], LLMRec [Liu *et al.*, 2023b], RecMind [Wang *et al.*, 2023b], [Liu *et al.*, 2023a]	M6-Rec [Cui *et al.*, 2022], RecLLM [Friedman *et al.*, 2023], Chat-REC [Gao *et al.*, 2023], [Wang *et al.*, 2023a; Lin and Zhang, 2023; He *et al.*, 2023]

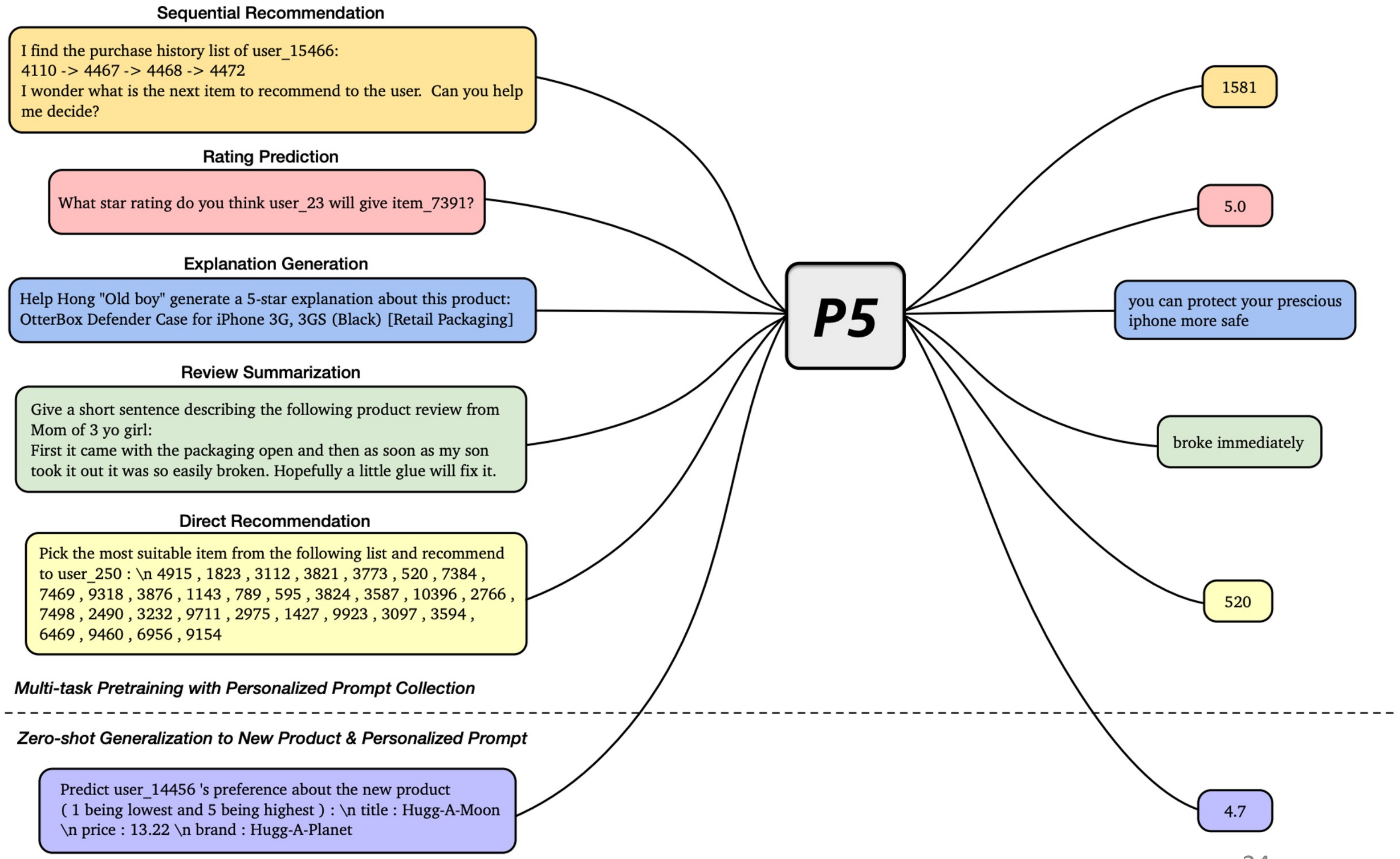
Image credit to [1]

[1] Li, Lei, Yongfeng Zhang, Dugang Liu, and Li Chen. "Large Language Models for Generative Recommendation: A Survey and Visionary Discussions." *arXiv preprint arXiv:2309.01157* (2023).
[2] Geng, Shijie, Shuchang Liu, Zuohui Fu, Yingqiang Ge, and Yongfeng Zhang. "Recommendation as Language Processing (RLP): A Unified Pretrain, Personalized Prompt & Predict Paradigm (P5)." RecSys 2022.
[3] Li, Lei, Yongfeng Zhang, and Li Chen. "Prompt Distillation for Efficient LLM-based Recommendation." CIKM 2023.
[4] Zhang, Junjie, et al. "Recommendation as instruction following: A large language model empowered recommendation approach." *arXiv preprint arXiv:2305.07001* (2023).

The P5 Generative Recommendation Paradigm

- P5: Pretrain, Personalized Prompt & Predict Paradigm [1]

- Learns multiple recommendation tasks together through a unified sequence-to-sequence framework
- Formulates different recommendation problems as prompt-based natural language tasks
- User-item information and corresponding features are integrated with personalized prompts as model inputs



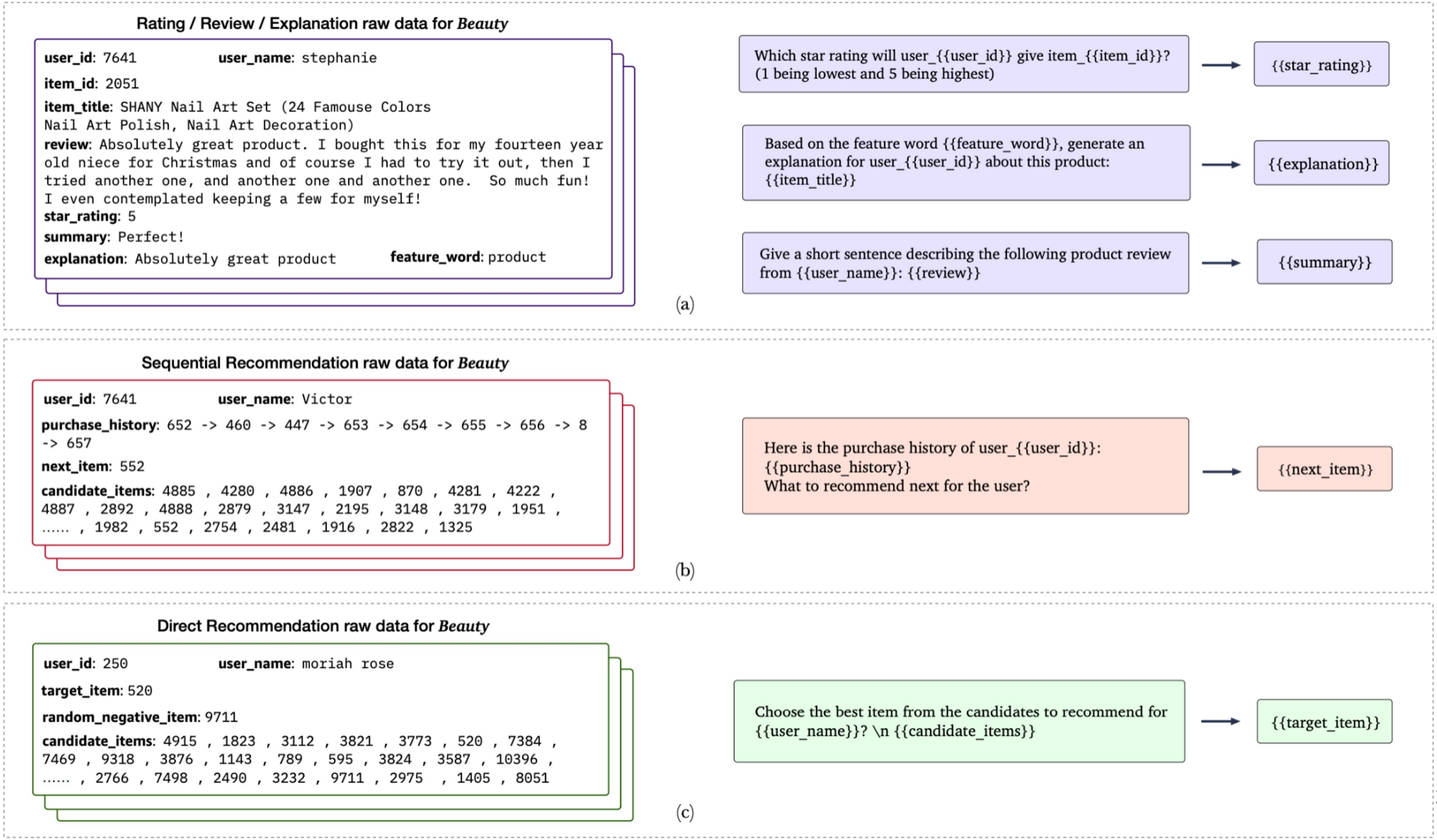
[1] Geng, Shijie, Shuchang Liu, Zuohui Fu, Yingqiang Ge, and Yongfeng Zhang. "Recommendation as Language Processing (RLP): A Unified Pretrain, Personalized Prompt & Predict Paradigm (P5)." RecSys 2022.

Personalization in Prompts

- Definition of **personalized prompts**
  - A prompt that includes personalized fields for different users and items
- User's preference can be indicated through
  - A **user ID** (e.g., "user_23")
  - Content **description of the user** such as location, preferred movie genres, etc.
- Item field can be represented by
  - An **item ID** (e.g., "item_7391")
  - Item **content metadata** that contains **detailed descriptions** of the item, e.g., item category

RUTGERS

Personalized Prompt Design



Design Multiple Prompts for Each Task

- To enhance variation in language style (e.g., sequential recommendation)

Prompt ID: 2-1

Input template: Given the following purchase history of
user_{{user_id}}:
{{purchase_history}}
predict next possible item to be purchased by the user?

Target template: {{next_item}}

Prompt ID: 2-2

Input template: I find the purchase history list of user_{{user_id}}:
{{purchase_history}}
I wonder which is the next item to recommend to the user. Can you
help me decide?

Target template: {{next_item}}

Prompt ID: 2-3

Input template: Here is the purchase history list of
user_{{user_id}}:
{{purchase_history}}
try to recommend next item to the user

Target template: {{next_item}}

Prompt ID: 2-4

Input template: Given the following purchase history of
{{user_desc}}:
{{purchase_history}}
predict next possible item for the user

Target template: {{next_item}}

Prompt ID: 2-5

Input template: Based on the purchase history of {{user_desc}}:
{{purchase_history}}
Can you decide the next item likely to be purchased by the user?

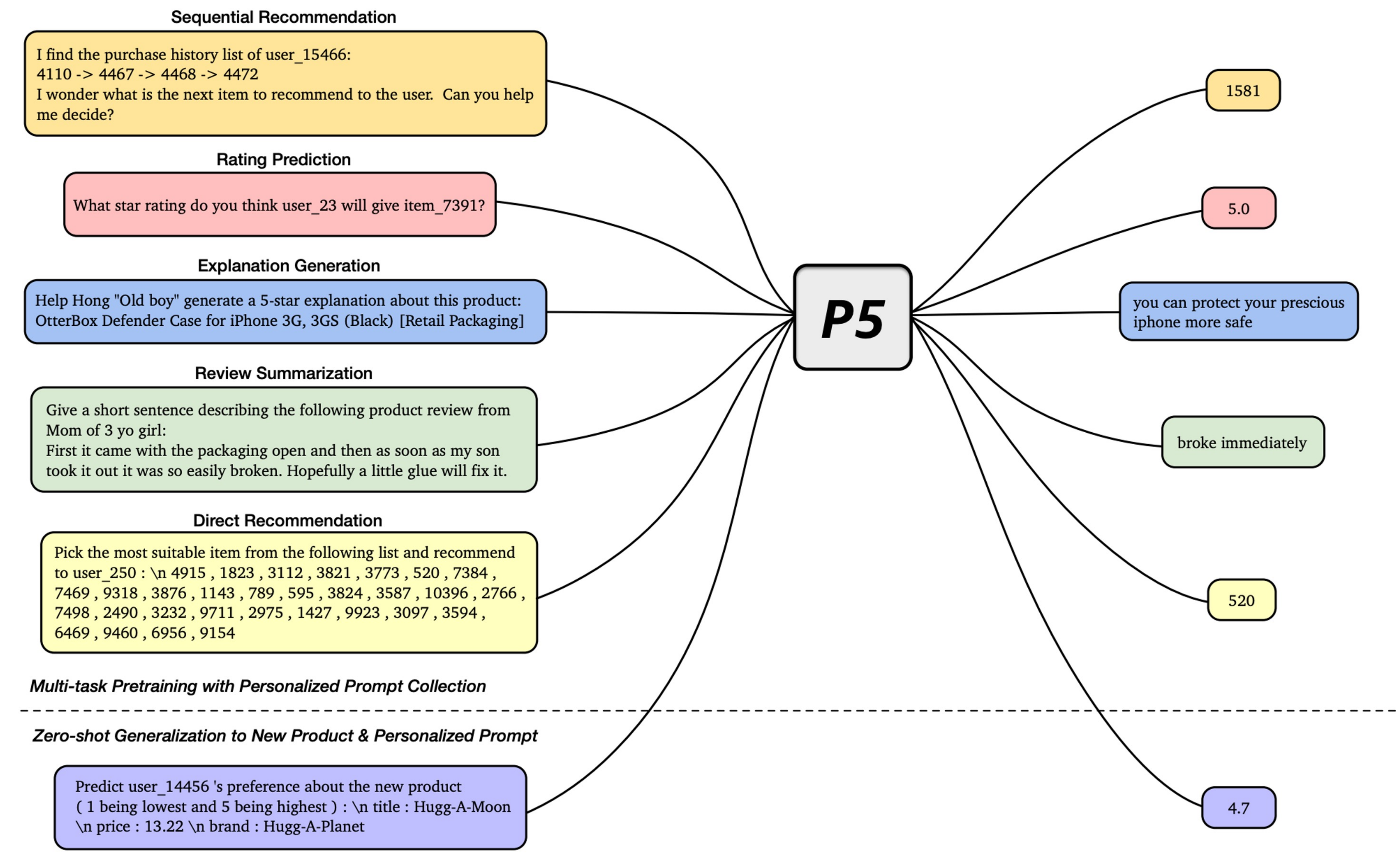
Target template: {{next_item}}

Prompt ID: 2-6

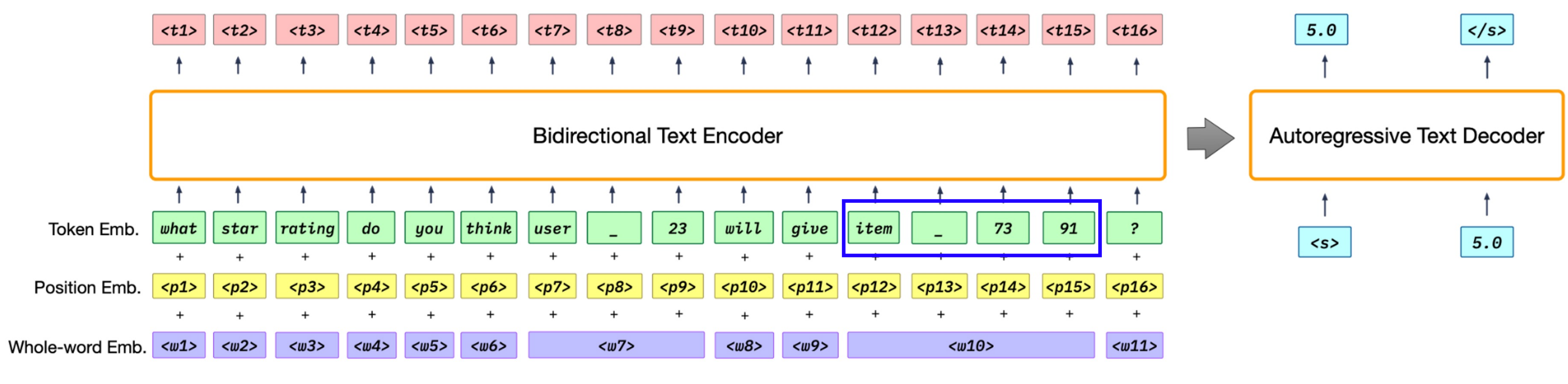
Input template: Here is the purchase history of {{user_desc}}:
{{purchase_history}}
What to recommend next for the user?

Target template: {{next_item}}

Multi-Task Pre-training



Multi-Task Pre-training

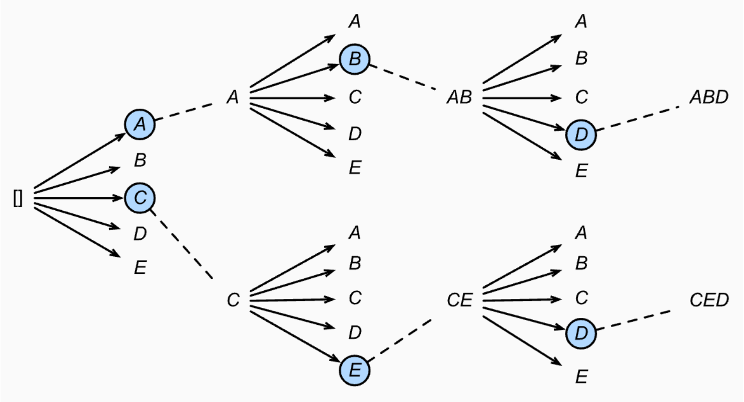


- P5 is pre-trained on top of T5 checkpoints (to enable basic ability for language understanding)
- By default, P5 uses multiple sub-word units to represent personalized fields (e.g., [“item”, “_”, “73”, “91”])

Generative Recommendation with Beam Search

- The encoder takes input sequence
- The decoder autoregressively generates next words:
  - **Autoregressive LM loss** is shared by all tasks: $\mathcal{L}_{\theta}^{\mathrm{P5}} = -\sum_{j=1}^{|\mathbf{y}|} \log P_{\theta}\left(\mathbf{y}_{j} \mid \mathbf{y}_{<j}, \mathbf{x}\right)$

- P5 can unify various recommendation tasks with **one model, one loss, and one data format**



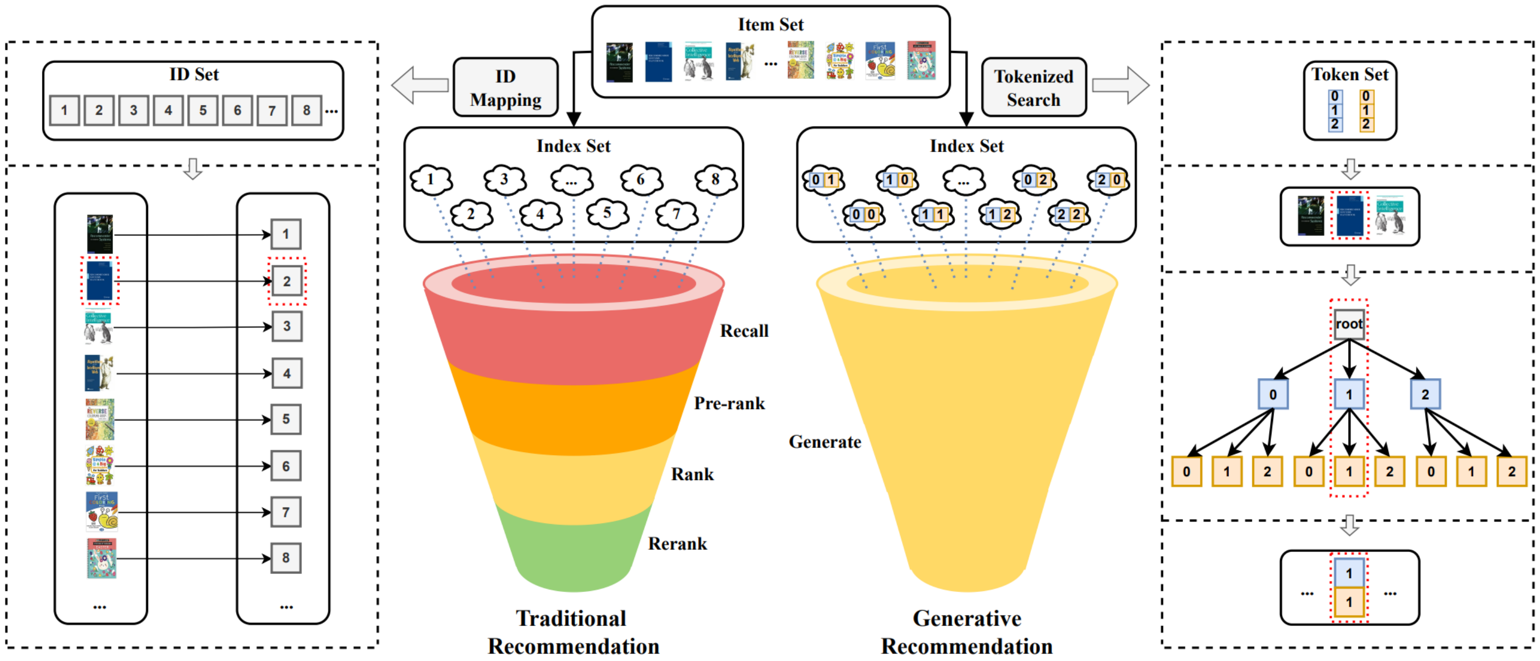
- Inference with pretrained P5
  - Simply apply **beam search** to generate a list of potential next items
    - Beam size set to N (N candidates)

Image credit to [1]

[1] https://d2l.ai/chapter_recurrent-modern/beam-search.html

Generative Recommendation with Beam Search

- Since item IDs are tokenized (e.g., [“item”, “_”, “73”, “91”]), beam search is bounded on width
  - E.g., 100 tokens width: ⟨00⟩, ⟨01⟩, ⟨02⟩, …, ⟨98⟩, ⟨99⟩
- Assigning an item a token as in traditional recommendation is infeasible for LLM
  - Consume a lot of memory and computationally expensive



[1] Li, Lei, Yongfeng Zhang, Dugang Liu, and Li Chen. "Large Language Models for Generative Recommendation: A Survey and Visionary Discussions." *arXiv preprint arXiv:2309.01157* (2023).

Advantages of P5 Generative Recommendation

- Immerses recommendation models into a full language environment
  - With the flexibility and expressiveness of language, there is no need to design feature-specific encoders
- P5 treats all personalized tasks as a conditional text generation problem
  - One data format, one model, one loss for multiple recommendation tasks
  - No need to design data-specific or task-specific recommendation models
- P5 attains sufficient zero-shot performance when generalizing to novel personalized prompts or unseen items in other domains

RUTGERS

Performance of P5 under seen Prompts

Rating Prediction:

Methods	Sports		Beauty		Toys	
	RMSE	MAE	RMSE	MAE	RMSE	MAE
MF	**1.0234**	0.7935	**1.1973**	0.9461	**1.0123**	0.7984
MLP	1.1277	0.7626	1.3078	0.9597	1.1215	0.8097
P5-S (1-6)	1.0594	**0.6639**	1.3128	**0.8428**	1.0746	0.7054
P5-B (1-6)	1.0357	0.6813	1.2843	0.8534	1.0544	0.7177
P5-S (1-10)	1.0522	0.6698	1.2989	0.8473	1.0550	0.7173
P5-B (1-10)	1.0292	0.6864	1.2870	0.8531	1.0245	**0.6931**

Sequential Recommendation:

Methods	Sports				Beauty				Toys			
	HR@5	NDCG@5	HR@10	NDCG@10	HR@5	NDCG@5	HR@10	NDCG@10	HR@5	NDCG@5	HR@10	NDCG@10
Caser	0.0116	0.0072	0.0194	0.0097	0.0205	0.0131	0.0347	0.0176	0.0166	0.0107	0.0270	0.0141
HGN	0.0189	0.0120	0.0313	0.0159	0.0325	0.0206	0.0512	0.0266	0.0321	0.0221	0.0497	0.0277
GRU4Rec	0.0129	0.0086	0.0204	0.0110	0.0164	0.0099	0.0283	0.0137	0.0097	0.0059	0.0176	0.0084
BERT4Rec	0.0115	0.0075	0.0191	0.0099	0.0203	0.0124	0.0347	0.0170	0.0116	0.0071	0.0203	0.0099
FDSA	0.0182	0.0122	0.0288	0.0156	0.0267	0.0163	0.0407	0.0208	0.0228	0.0140	0.0381	0.0189
SASRec	0.0233	0.0154	0.0350	0.0192	0.0387	0.0249	0.0605	0.0318	0.0463	0.0306	0.0675	0.0374
S^3-Rec	0.0251	0.0161	0.0385	0.0204	0.0387	0.0244	0.0647	0.0327	0.0443	0.0294	0.0700	0.0376
P5-S (2-3)	0.0272	0.0169	0.0361	0.0198	0.0503	0.0370	0.0659	0.0421	**0.0648**	**0.0567**	**0.0709**	**0.0587**
P5-B (2-3)	0.0364	0.0296	0.0431	0.0318	**0.0508**	**0.0379**	**0.0664**	**0.0429**	0.0608	0.0507	0.0688	0.0534
P5-S (2-13)	0.0258	0.0159	0.0346	0.0188	0.0490	0.0358	0.0646	0.0409	0.0647	0.0566	0.0705	0.0585
P5-B (2-13)	**0.0387**	**0.0312**	**0.0460**	**0.0336**	0.0493	0.0367	0.0645	0.0416	0.0587	0.0486	0.0675	0.0536

Explanation Generation:

Methods	Sports				Beauty				Toys			
	BLUE4	ROUGE1	ROUGE2	ROUGEL	BLUE4	ROUGE1	ROUGE2	ROUGEL	BLUE4	ROUGE1	ROUGE2	ROUGEL
Attn2Seq	0.5305	12.2800	1.2107	9.1312	0.7889	12.6590	1.6820	9.7481	1.6238	13.2245	2.9942	10.7398
NRT	0.4793	11.0723	1.1304	7.6674	0.8295	12.7815	1.8543	9.9477	1.9084	13.5231	3.6708	11.1867
PETER	0.7112	12.8944	1.3283	9.8635	1.1541	14.8497	2.1413	11.4143	1.9861	14.2716	3.6718	11.7010
P5-S (3-3)	**1.0447**	**14.9048**	**2.1297**	**11.1778**	**1.2237**	**17.6938**	**2.2489**	**12.8606**	2.2892	**15.4505**	3.6974	**12.1718**
P5-B (3-3)	1.0407	14.1589	2.1220	10.6096	0.9742	16.4530	1.8858	11.8765	**2.3185**	15.3474	**3.7209**	12.1312
PETER+	**2.4627**	**24.1181**	5.1937	**18.4105**	**3.2606**	25.5541	5.9668	19.7168	**4.7919**	28.3083	9.4520	**22.7017**
P5-S (3-9)	1.4101	23.5619	**5.4196**	17.6245	1.9788	**25.6253**	**6.3678**	**19.9497**	4.1222	**28.4088**	9.5432	22.6064
P5-B (3-9)	1.4689	23.5476	5.3926	17.5852	1.8765	25.1183	6.0764	19.4488	3.8933	27.9916	9.5896	22.2178
P5-S (3-12)	1.3212	23.2474	5.3461	17.3780	1.9425	25.1474	6.0551	19.5601	4.2764	28.1897	9.1327	22.2514
P5-B (3-12)	1.4303	23.3810	5.3239	17.4913	1.9031	25.1763	6.1980	19.5188	3.5861	28.1369	**9.7562**	22.3056

RUTGERS

Performance of P5 under seen Prompts

Review Summarization:

Methods	Sports				Beauty				Toys			
	BLUE2	ROUGE1	ROUGE2	ROUGEL	BLUE2	ROUGE1	ROUGE2	ROUGEL	BLUE2	ROUGE1	ROUGE2	ROUGEL
T0 (4-1)	2.1581	2.2695	0.5694	1.6221	1.2871	1.2750	0.3904	0.9592	2.2296	2.4671	0.6482	1.8424
GPT-2 (4-1)	0.7779	4.4534	1.0033	1.9236	0.5879	3.3844	0.6756	1.3956	0.6221	3.7149	0.6629	1.4813
P5-S (4-1)	2.4962	11.6701	2.7187	10.4819	**2.1225**	**8.4205**	**1.6676**	**7.5476**	**2.4752**	**9.4200**	**1.5975**	**8.2618**
P5-B (4-1)	**2.6910**	**12.0314**	**3.2921**	**10.7274**	1.9325	8.2909	1.4321	7.4000	1.7833	8.7222	1.3210	7.6134

Direct Recommendation:

Methods	Sports					Beauty					Toys				
	HR@1	HR@5	NDCG@5	HR@10	NDCG@10	HR@1	HR@5	NDCG@5	HR@10	NDCG@10	HR@1	HR@5	NDCG@5	HR@10	NDCG@10
BPR-MF	0.0314	0.1404	0.0848	0.2563	0.1220	0.0311	0.1426	0.0857	0.2573	0.1224	0.0233	0.1066	0.0641	0.2003	0.0940
BPR-MLP	0.0351	0.1520	0.0927	0.2671	0.1296	0.0317	0.1392	0.0848	0.2542	0.1215	0.0252	0.1142	0.0688	0.2077	0.0988
SimpleX	0.0331	**0.2362**	**0.1505**	0.3290	0.1800	0.0325	0.2247	0.1441	0.3090	0.1711	0.0268	**0.1958**	**0.1244**	**0.2662**	**0.1469**
P5-S (5-1)	0.0638	0.2096	0.1375	0.3143	0.1711	0.0600	0.2021	0.1316	0.3121	0.1670	0.0405	0.1538	0.0969	0.2405	0.1248
P5-B (5-1)	0.0245	0.0816	0.0529	0.1384	0.0711	0.0224	0.0904	0.0559	0.1593	0.0780	0.0187	0.0827	0.0500	0.1543	0.0729
P5-S (5-4)	0.0701	0.2241	0.1483	**0.3313**	**0.1827**	**0.0862**	**0.2448**	**0.1673**	**0.3441**	**0.1993**	0.0413	0.1411	0.0916	0.2227	0.1178
P5-B (5-4)	0.0299	0.1026	0.0665	0.1708	0.0883	0.0506	0.1557	0.1033	0.2350	0.1287	0.0435	0.1316	0.0882	0.2000	0.1102
P5-S (5-5)	0.0574	0.1503	0.1050	0.2207	0.1276	0.0601	0.1611	0.1117	0.2370	0.1360	0.0440	0.1282	0.0865	0.2011	0.1098
P5-B (5-5)	0.0641	0.1794	0.1229	0.2598	0.1488	0.0588	0.1573	0.1089	0.2325	0.1330	0.0386	0.1122	0.0756	0.1807	0.0975
P5-S (5-8)	0.0567	0.1514	0.1049	0.2196	0.1269	0.0571	0.1566	0.1078	0.2317	0.1318	**0.0451**	0.1322	0.0889	0.2023	0.1114
P5-B (5-8)	**0.0726**	0.1955	0.1355	0.2802	0.1627	0.0608	0.1564	0.1096	0.2300	0.1332	0.0389	0.1147	0.0767	0.1863	0.0997

Observation: P5 **achieves promising performances** on the five task families when taking **seen prompt** templates as model inputs

RUTGERS

Performance of P5 under unseen Prompts

Observation: Multitask prompted pretraining empowers P5 good robustness to understand **unseen prompts** with wording variations

Sequential Recommendation:

Methods	Sports				Beauty				Toys			
	HR@5	NDCG@5	HR@10	NDCG@10	HR@5	NDCG@5	HR@10	NDCG@10	HR@5	NDCG@5	HR@10	NDCG@10
Caser	0.0116	0.0072	0.0194	0.0097	0.0205	0.0131	0.0347	0.0176	0.0166	0.0107	0.0270	0.0141
HGN	0.0189	0.0120	0.0313	0.0159	0.0325	0.0206	0.0512	0.0266	0.0321	0.0221	0.0497	0.0277
GRU4Rec	0.0129	0.0086	0.0204	0.0110	0.0164	0.0099	0.0283	0.0137	0.0097	0.0059	0.0176	0.0084
BERT4Rec	0.0115	0.0075	0.0191	0.0099	0.0203	0.0124	0.0347	0.0170	0.0116	0.0071	0.0203	0.0099
FDSA	0.0182	0.0122	0.0288	0.0156	0.0267	0.0163	0.0407	0.0208	0.0228	0.0140	0.0381	0.0189
SASRec	0.0233	0.0154	0.0350	0.0192	0.0387	0.0249	0.0605	0.0318	0.0463	0.0306	0.0675	0.0374
S^3-Rec	0.0251	0.0161	0.0385	0.0204	0.0387	0.0244	0.0647	0.0327	0.0443	0.0294	0.0700	0.0376
P5-S (2-3)	0.0272	0.0169	0.0361	0.0198	0.0503	0.0370	0.0659	0.0421	**0.0648**	**0.0567**	**0.0709**	**0.0587**
P5-B (2-3)	0.0364	0.0296	0.0431	0.0318	**0.0508**	**0.0379**	**0.0664**	**0.0429**	0.0608	0.0507	0.0688	0.0534
P5-S (2-13)	0.0258	0.0159	0.0346	0.0188	0.0490	0.0358	0.0646	0.0409	0.0647	0.0566	0.0705	0.0585
P5-B (2-13)	**0.0387**	**0.0312**	**0.0460**	**0.0336**	0.0493	0.0367	0.0645	0.0416	0.0587	0.0486	0.0675	0.0536

Explanation Generation:

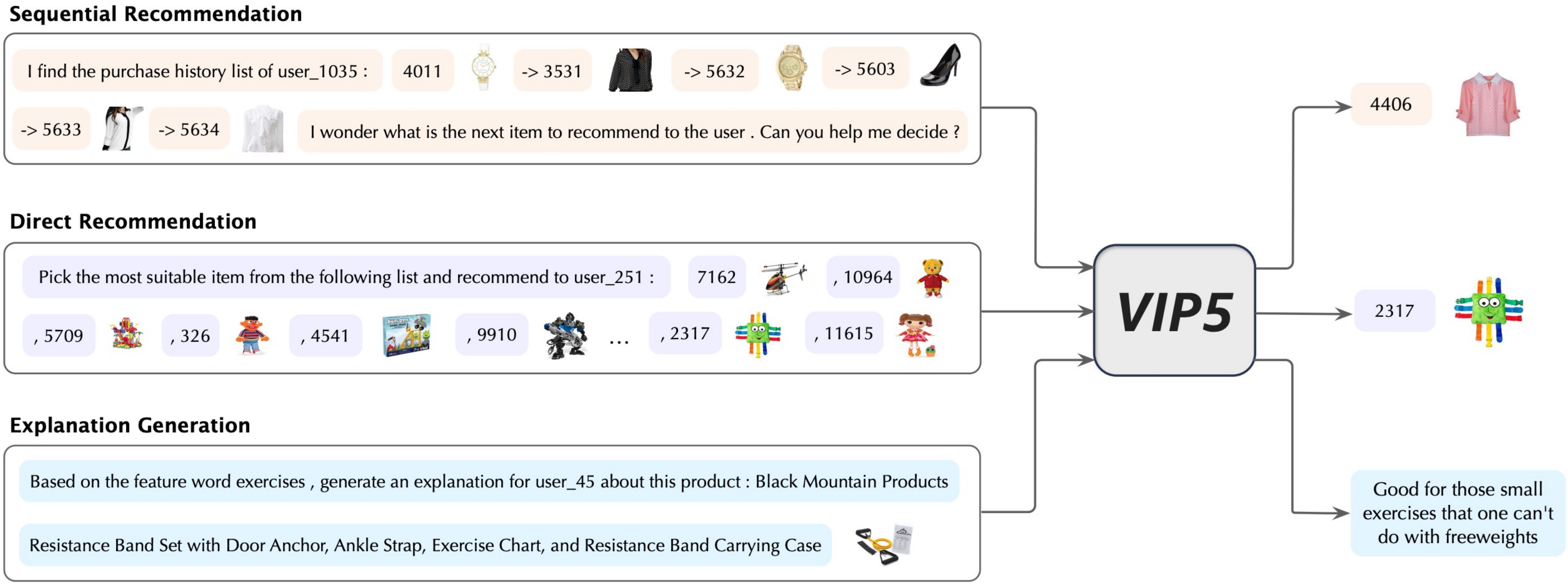
Methods	Sports				Beauty				Toys			
	BLUE4	ROUGE1	ROUGE2	ROUGEL	BLUE4	ROUGE1	ROUGE2	ROUGEL	BLUE4	ROUGE1	ROUGE2	ROUGEL
Attn2Seq	0.5305	12.2800	1.2107	9.1312	0.7889	12.6590	1.6820	9.7481	1.6238	13.2245	2.9942	10.7398
NRT	0.4793	11.0723	1.1304	7.6674	0.8295	12.7815	1.8543	9.9477	1.9084	13.5231	3.6708	11.1867
PETER	0.7112	12.8944	1.3283	9.8635	1.1541	14.8497	2.1413	11.4143	1.9861	14.2716	3.6718	11.7010
P5-S (3-3)	**1.0447**	**14.9048**	**2.1297**	**11.1778**	**1.2237**	**17.6938**	**2.2489**	**12.8606**	2.2892	**15.4505**	3.6974	**12.1718**
P5-B (3-3)	1.0407	14.1589	2.1220	10.6096	0.9742	16.4530	1.8858	11.8765	**2.3185**	15.3474	**3.7209**	12.1312
PETER+	**2.4627**	**24.1181**	5.1937	**18.4105**	**3.2606**	25.5541	5.9668	19.7168	**4.7919**	28.3083	9.4520	**22.7017**
P5-S (3-9)	1.4101	23.5619	**5.4196**	17.6245	1.9788	**25.6253**	**6.3678**	**19.9497**	4.1222	**28.4088**	9.5432	22.6064
P5-B (3-9)	1.4689	23.5476	5.3926	17.5852	1.8765	25.1183	6.0764	19.4488	3.8933	27.9916	9.5896	22.2178
P5-S (3-12)	1.3212	23.2474	5.3461	17.3780	1.9425	25.1474	6.0551	19.5601	4.2764	28.1897	9.1327	22.2514
P5-B (3-12)	1.4303	23.3810	5.3239	17.4913	1.9031	25.1763	6.1980	19.5188	3.5861	28.1369	**9.7562**	22.3056

Direct Recommendation:

Methods	Sports					Beauty					Toys				
	HR@1	HR@5	NDCG@5	HR@10	NDCG@10	HR@1	HR@5	NDCG@5	HR@10	NDCG@10	HR@1	HR@5	NDCG@5	HR@10	NDCG@10
BPR-MF	0.0314	0.1404	0.0848	0.2563	0.1220	0.0311	0.1426	0.0857	0.2573	0.1224	0.0233	0.1066	0.0641	0.2003	0.0940
BPR-MLP	0.0351	0.1520	0.0927	0.2671	0.1296	0.0317	0.1392	0.0848	0.2542	0.1215	0.0252	0.1142	0.0688	0.2077	0.0988
SimpleX	0.0331	**0.2362**	**0.1505**	0.3290	0.1800	0.0325	0.2247	0.1441	0.3090	0.1711	0.0268	**0.1958**	**0.1244**	**0.2662**	**0.1469**
P5-S (5-1)	0.0638	0.2096	0.1375	0.3143	0.1711	0.0600	0.2021	0.1316	0.3121	0.1670	0.0405	0.1538	0.0969	0.2405	0.1248
P5-B (5-1)	0.0245	0.0816	0.0529	0.1384	0.0711	0.0224	0.0904	0.0559	0.1593	0.0780	0.0187	0.0827	0.0500	0.1543	0.0729
P5-S (5-4)	0.0701	0.2241	0.1483	**0.3313**	**0.1827**	**0.0862**	**0.2448**	**0.1673**	**0.3441**	**0.1993**	0.0413	0.1411	0.0916	0.2227	0.1178
P5-B (5-4)	0.0299	0.1026	0.0665	0.1708	0.0883	0.0506	0.1557	0.1033	0.2350	0.1287	0.0435	0.1316	0.0882	0.2000	0.1102
P5-S (5-5)	0.0574	0.1503	0.1050	0.2207	0.1276	0.0601	0.1611	0.1117	0.2370	0.1360	0.0440	0.1282	0.0865	0.2011	0.1098
P5-B (5-5)	0.0641	0.1794	0.1229	0.2598	0.1488	0.0588	0.1573	0.1089	0.2325	0.1330	0.0386	0.1122	0.0756	0.1807	0.0975
P5-S (5-8)	0.0567	0.1514	0.1049	0.2196	0.1269	0.0571	0.1566	0.1078	0.2317	0.1318	**0.0451**	0.1322	0.0889	0.2023	0.1114
P5-B (5-8)	**0.0726**	0.1955	0.1355	0.2802	0.1627	0.0608	0.1564	0.1096	0.2300	0.1332	0.0389	0.1147	0.0767	0.1863	0.0997

Easy Handling of Multi-modality Information

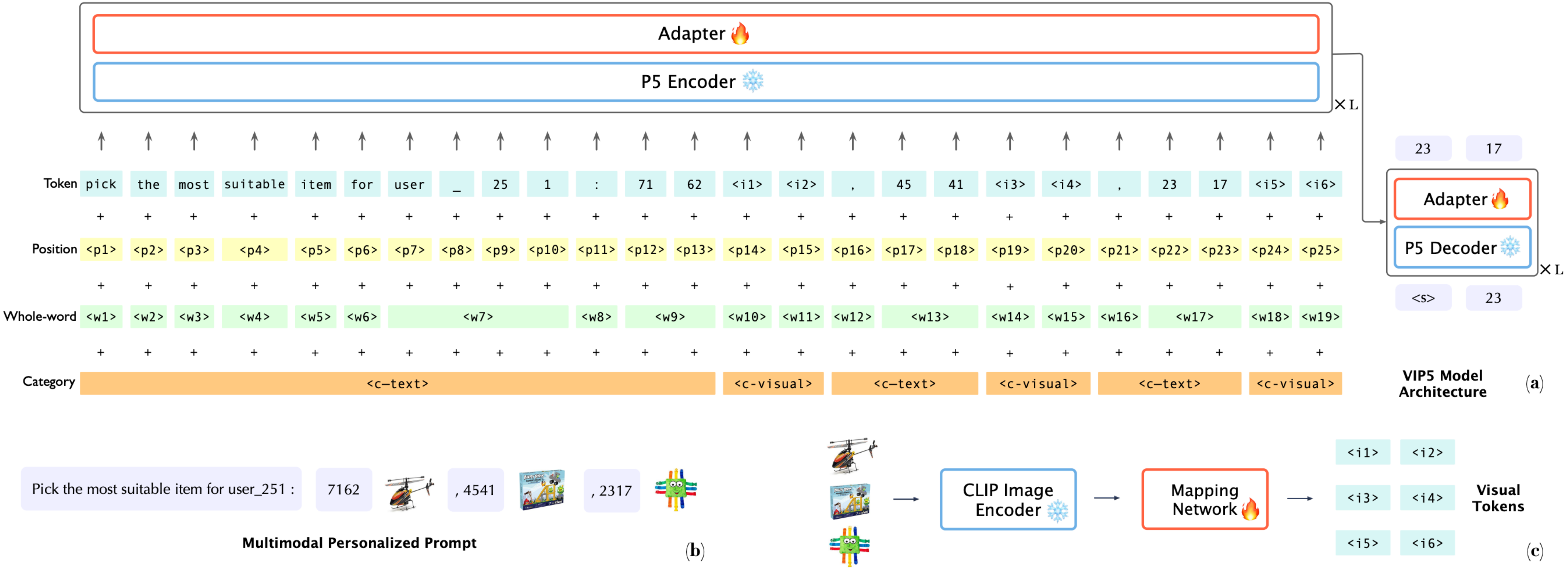
- Item images can be directly inserted into personalized prompts [1]



[1] Geng, Shijie, Juntao Tan, Shuchang Liu, Zuohui Fu, and Yongfeng Zhang. "VIP5: Towards Multimodal Foundation Models for Recommendation." EMNLP 2023.

Easy Handling of Multi-modality Information

- Item images can be converted into visual tokens



[1] Geng, Shijie, Juntao Tan, Shuchang Liu, Zuohui Fu, and Yongfeng Zhang. "VIP5: Towards Multimodal Foundation Models for Recommendation." EMNLP 2023.

RUTGERS

Easy Handling of Multi-modality Information

- Item images can be directly inserted into prompts
  - Multi-modality information further improves performance

Methods	Sports HR@5	Sports NDCG@5	Sports HR@10	Sports NDCG@10	Beauty HR@5	Beauty NDCG@5	Beauty HR@10	Beauty NDCG@10
HGN	0.0189	0.0120	0.0313	0.0159	0.0325	0.0206	0.0512	0.0266
SASRec	0.0233	0.0154	0.0350	0.0192	0.0387	0.0249	0.0605	0.0318
S^3-Rec	0.0251	0.0161	0.0385	0.0204	0.0387	0.0244	0.0647	0.0327
P5 *(A-3)*	0.0272	0.0169	0.0361	0.0198	0.0503	0.0370	0.0659	0.0421
VIP5 *(A-3)*	**0.0412**	**0.0345**	**0.0475**	**0.0365**	**0.0556**	**0.0427**	**0.0677**	**0.0467**
P5 *(A-9)*	0.0258	0.0159	0.0346	0.0188	0.0490	0.0358	0.0646	0.0409
VIP5 *(A-9)*	0.0392	0.0327	0.0456	0.0347	0.0529	0.0413	0.0655	0.0454

Methods	Clothing HR@5	Clothing NDCG@5	Clothing HR@10	Clothing NDCG@10	Toys HR@5	Toys NDCG@5	Toys HR@10	Toys NDCG@10
HGN	0.0107	0.0071	0.0175	0.0092	0.0321	0.0221	0.0497	0.0277
SASRec	0.0107	0.0066	0.0194	0.0095	0.0463	0.0306	0.0675	0.0374
S^3-Rec	0.0076	0.0045	0.0135	0.0063	0.0443	0.0294	0.0700	0.0376
P5 *(A-3)*	0.0478	0.0376	0.0554	0.0401	0.0655	0.0570	0.0726	0.0593
VIP5 *(A-3)*	**0.0603**	**0.0564**	**0.0632**	**0.0573**	**0.0662**	**0.0577**	**0.0749**	**0.0604**
P5 *(A-9)*	0.0455	0.0359	0.0534	0.0385	0.0631	0.0547	0.0701	0.0569
VIP5 *(A-9)*	0.0569	0.0531	0.0597	0.0540	0.0641	0.0556	0.0716	0.0580

Sequential Recommendation Performance

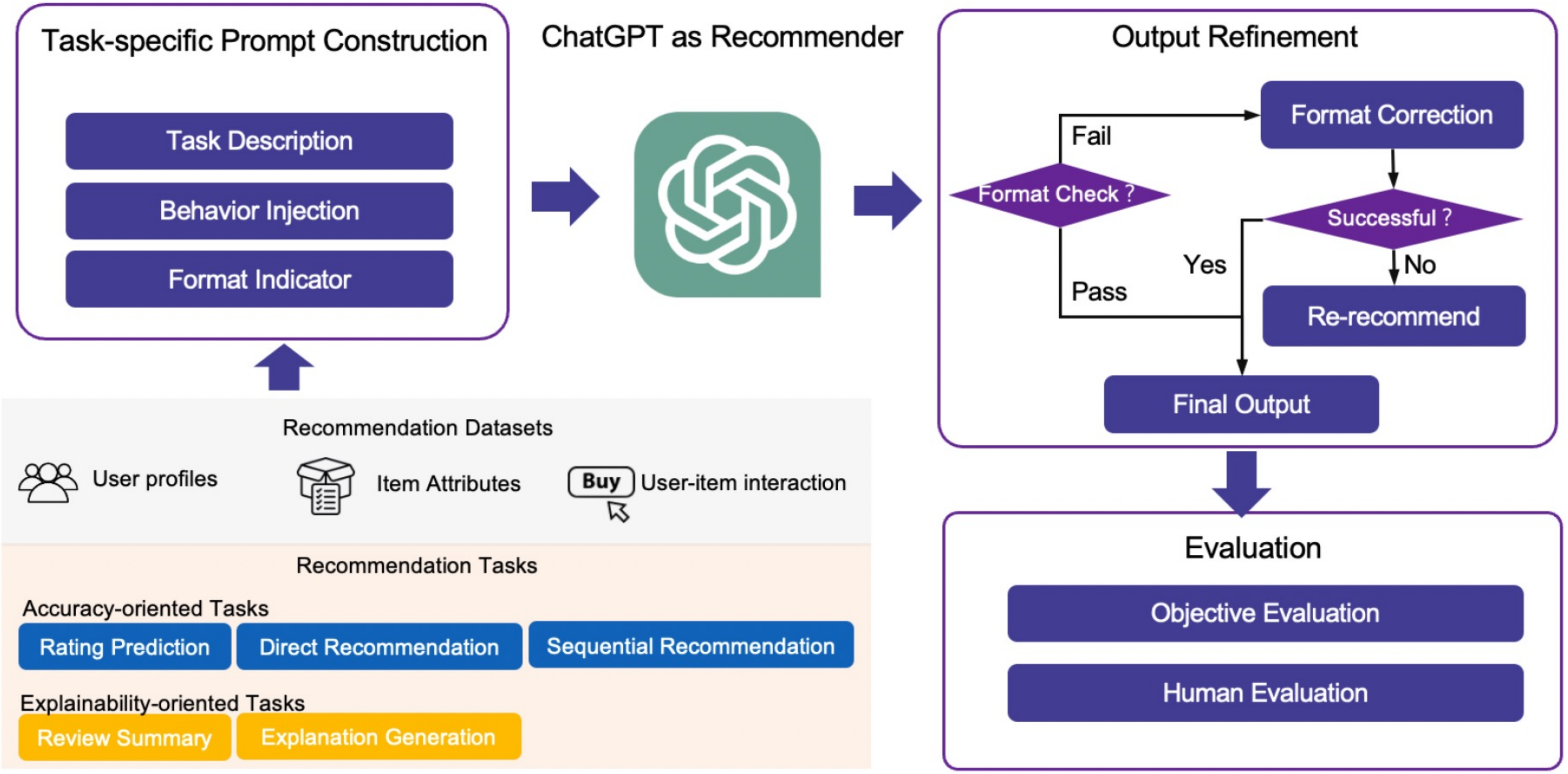
Methods	Sports HR@1	Sports HR@5	Sports NDCG@5	Sports HR@10	Sports NDCG@10	Beauty HR@1	Beauty HR@5	Beauty NDCG@5	Beauty HR@10	Beauty NDCG@10
BPR-MF	0.0314	0.1404	0.0848	0.2563	0.1220	0.0311	0.1426	0.0857	**0.2573**	0.1224
BPR-MLP	0.0351	0.1520	0.0927	0.2671	0.1296	0.0317	0.1392	0.0848	0.2542	0.1215
VBPR	0.0262	0.1138	0.0691	0.2060	0.0986	0.0380	0.1472	0.0925	0.2468	0.1245
P5 *(B-5)*	0.0574	0.1503	0.1050	0.2207	0.1276	0.0601	0.1611	0.1117	0.2370	0.1360
VIP5 *(B-5)*	0.0606	0.1743	0.1185	0.2539	0.1441	0.0580	0.1598	0.1099	0.2306	0.1327
P5 *(B-8)*	0.0567	0.1514	0.1049	0.2196	0.1269	0.0571	0.1566	0.1078	0.2317	0.1318
VIP5 *(B-8)*	**0.0699**	**0.1882**	**0.1304**	**0.2717**	**0.1572**	**0.0615**	**0.1655**	**0.1147**	0.2407	**0.1388**

Methods	Clothing HR@1	Clothing HR@5	Clothing NDCG@5	Clothing HR@10	Clothing NDCG@10	Toys HR@1	Toys HR@5	Toys NDCG@5	Toys HR@10	Toys NDCG@10
BPR-MF	0.0296	0.1280	0.0779	0.2319	0.1112	0.0233	0.1066	0.0641	0.2003	0.0940
BPR-MLP	0.0342	0.1384	0.0858	0.2327	0.1161	0.0252	0.1142	0.0688	0.2077	0.0988
VBPR	0.0352	0.1410	0.0877	**0.2420**	0.1201	0.0337	0.1294	0.0808	**0.2199**	0.1098
P5 *(B-5)*	0.0320	0.0986	0.0652	0.1659	0.0867	0.0418	0.1219	0.0824	0.1942	0.1056
VIP5 *(B-5)*	0.0481	0.1287	0.0890	0.1992	0.1116	0.0428	0.1225	0.0833	0.1906	0.1051
P5 *(B-8)*	0.0355	0.1019	0.0688	0.1722	0.0912	0.0422	0.1286	0.0858	0.2041	0.1099
VIP5 *(B-8)*	**0.0552**	**0.1544**	**0.1058**	0.2291	**0.1297**	**0.0433**	**0.1301**	**0.0875**	0.2037	**0.1110**

Direct Recommendation Performance

ChatGPT as Recommender

- Instruct ChatGPT to perform different tasks w/o fine-tuning
  - Few-shot or zero-shot settings (w/ or w/o demonstration examples)



[1] Liu, Junling, et al. "Is chatgpt a good recommender? a preliminary study." *arXiv preprint arXiv:2304.10149* (2023).

ChatGPT on Recommendation Tasks

- Recommendation performance is relatively weak

Sequential Recommendation

Methods	Beauty			
	HR@5	NDCG@5	HR@10	NDCG@10
Caser	0.0205	0.0131	0.0347	0.0176
HGN	0.0325	0.0206	0.0512	0.0266
GRU4Rec	0.0164	0.0099	0.0283	0.0137
BERT4Rec	0.0203	0.0124	0.0347	0.0170
FDSA	0.0267	0.0163	0.0407	0.0208
SASRec	0.0387	0.0249	0.0605	0.0318
S^3-Rec	0.0387	0.0244	0.0647	0.0327
P5-B	**0.0493**	**0.0367**	**0.0645**	**0.0416**
ChatGPT(zero-shot)	0.0000	0.0000	0.0000	0.0000
ChatGPT(few-shot)	0.0135	0.0135	0.0135	0.0135

Direct Recommendation

Methods	Beauty			
	HR@5	NDCG@5	HR@10	NDCG@10
BPR-MF	0.1426	0.0857	0.2573	0.1224
BPR-MLP	0.1392	0.0848	0.2542	0.1215
SimpleX	**0.2247**	**0.1441**	**0.3090**	**0.1711**
P5-B	0.1564	0.1096	0.2300	0.1332
ChatGPT(zero-shot)	0.0217	0.0111	0.0652	0.0252
ChatGPT(few-shot)	0.0349	0.0216	0.0930	0.0398

Rating Prediction

Methods	Beauty	
	RMSE	MAE
MF	1.1973	0.9461
MLP	1.3078	0.9597
ChatGPT(zero-shot)	1.4059	1.1861
ChatGPT(few-shot)	**1.0751**	**0.6977**

[1] Liu, Junling, et al. "Is chatgpt a good recommender? a preliminary study." *arXiv preprint arXiv:2304.10149* (2023).

RUTGERS

ChatGPT on Generation Tasks

- Performance with automatic metrics is bad
- Rated highly by human evaluators
  - Existing metrics (BLEU and ROUGE) overly stress the matching between generation and ground-truth [2]

Explanation Generation

Methods	Beauty			
	BLUE4	ROUGE1	ROUGE2	ROUGEL
Attn2Seq	0.7889	12.6590	1.6820	9.7481
NRT	0.8295	12.7815	1.8543	9.9477
PETER	1.1541	14.8497	2.1413	11.4143
P5-B	0.9742	16.4530	1.8858	11.8765
PETER+	**3.2606**	**25.5541**	**5.9668**	**19.7168**
ChatGPT(zero-shot)	0.0000	8.5992	0.6995	4.7564
ChatGPT(few-shot)	1.1967	11.4103	2.5675	5.9119

Methods	Evaluators				avg_top1_ration	avg_position
	Eva_1	Eva_2	Eva_3	Eva_4		
Ground truth	25.0%	45.0%	45.0%	50.0%	38.0%	1.83
P5	0.0%	0.0%	0.0%	0.0%	0.0%	2.71
ChatGPT(zero-shot)	75.0%	55.0%	55.0%	50.0%	**62.0%**	**1.46**

Review Summarization

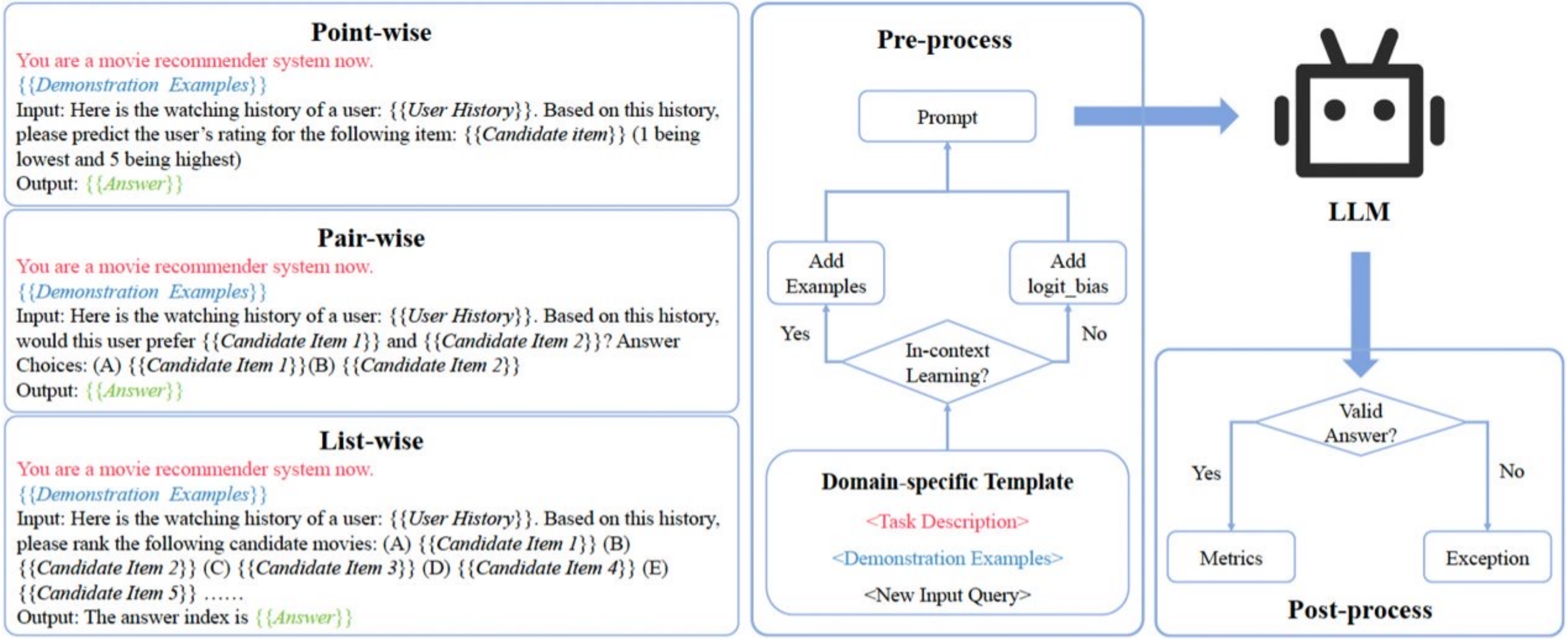
Methods	Beauty			
	BLUE4	ROUGE1	ROUGE2	ROUGEL
T0	1.2871	1.2750	0.3904	0.9592
GPT-2	0.5879	3.3844	0.6756	1.3956
P5-B	**2.1225**	**8.4205**	**1.6676**	**7.5476**
ChatGPT(zero-shot)	0.0000	3.8246	0.2857	3.1344
ChatGPT(few-shot)	0.0000	2.7822	0.0000	2.4328

Methods	Evaluators					avg_top1_ration	avg_position
	Eva_1	Eva_2	Eva_3	Eva_4	Eva_5		
Ground truth	12.5%	10.6%	8.7%	17.3%	22.1%	14.2%	2.91
P5	5.8%	0.0%	5.7%	11.5%	19.2%	8.5%	3.16
ChatGPT(zero-shot)	46.2%	37.5%	36.5%	45.2%	23.1%	37.7%	**1.90**
ChatGPT(few-shot)	35.6%	51.9%	49.0%	26.0%	35.6%	**39.6%**	2.01

[1] Liu, Junling, et al. "Is chatgpt a good recommender? a preliminary study." *arXiv preprint arXiv:2304.10149* (2023).
[2] Wang, Xiaolei, et al. "Rethinking the Evaluation for Conversational Recommendation in the Era of Large Language Models." *arXiv preprint arXiv:2305.13112* (2023).

ChatGPT as Recommender

- ChatGPT on three types of recommendation w/o fine-tuning
  - Point-wise (rate), pair-wise (compare), list-wise (rank)



[1] Dai, Sunhao, et al. "Uncovering ChatGPT's Capabilities in Recommender Systems." RecSys 2023.

RUTGERS

Recommendation Performance of ChatGPT

- Outperform weak baselines on the three recommendation tasks
  - Random, pop

Domain	Metric	random	pop	text-davinci-002			text-davinci-003			gpt-3.5-turbo (ChatGPT)		
				point-wise	pair-wise	list-wise	point-wise	pair-wise	list-wise	point-wise	pair-wise	list-wise
Movie	Compliance Rate	-	-	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	99.98%	100.00%
	NDCG@1	0.2000	0.2240	0.3110	0.3203	0.2600	0.2259	0.2843	0.3260	**0.3342**	0.3230	0.3320
	NDCG@3	0.4262	0.4761	0.5416	0.5728	0.4990	0.4618	0.5441	0.5564	**0.5912**	0.5827	0.5785
	MRR@3	0.3667	0.4103	0.4824	0.5071	0.4363	0.3998	0.4763	0.4950	**0.5260**	0.5162	0.5167
Book	Compliance Rate	-	-	99.96%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	99.98%	99.80%
	NDCG@1	0.2000	0.2440	0.2420	0.2847	0.2000	0.2325	0.2887	0.2440	0.2823	0.3061	**0.3126**
	NDCG@3	0.4262	0.4999	0.4889	0.5298	0.4290	0.4585	0.5293	0.4597	0.5075	0.5350	**0.5395**
	MRR@3	0.3667	0.4340	0.4247	0.4646	0.3690	0.3993	0.4665	0.4040	0.4495	0.4774	**0.4800**
Music	Compliance Rate	-	-	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	99.96%	99.80%
	NDCG@1	0.2000	0.1780	0.2354	0.2397	0.2300	0.2377	0.2690	0.2540	0.2892	0.3077	**0.3086**
	NDCG@3	0.4262	0.4094	0.4623	0.4681	0.4277	0.4732	0.5072	0.4506	0.5201	0.5439	**0.5567**
	MRR@3	0.3667	0.3470	0.4030	0.4082	0.3750	0.4113	0.4448	0.4000	0.4605	0.4830	**0.4950**
News	Compliance Rate	-	-	99.80%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%	99.60%
	NDCG@1	0.2000	**0.3080**	0.2183	0.2200	0.2920	0.2532	0.2630	0.2540	0.2591	0.2491	0.2892
	NDCG@3	0.4262	**0.5444**	0.4483	0.4550	0.5059	0.4880	0.4892	0.4742	0.4826	0.4991	0.5094
	MRR@3	0.3667	**0.4840**	0.3879	0.3936	0.4497	0.4271	0.4294	0.4173	0.4246	0.4354	0.4515

[1] Dai, Sunhao, et al. "Uncovering ChatGPT's Capabilities in Recommender Systems." RecSys 2023.

With Fine-tuning or Without Fine-tuning

- Without fine-tuning, LLM cannot easily solve RS problems
  - RS is a highly specialized area that requires collaborative knowledge, which LLM did not learn during the pre-training stage [1]
  - Collaborative knowledge such as user behavior data is highy dynamic
- RS practitioners do not have an existential crisis as NLP community
  - Many NLP problems can be easily addressed by LLM
- RS is still an open problem and will evolve with LLM

[1] Lin, Jianghao, et al. "How Can Recommender Systems Benefit from Large Language Models: A Survey." *arXiv preprint arXiv:2306.05817* (2023).

Role of LLM in Recommendation

- LLM as RS
  - E.g., P5 and ChatGPT-based recommenders
- LLM in RS as a component

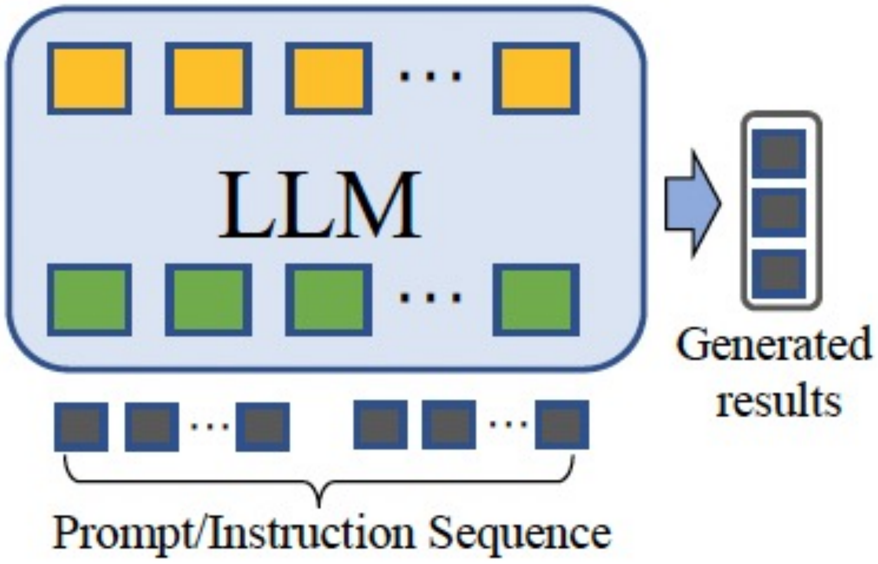


Image credit to [1]

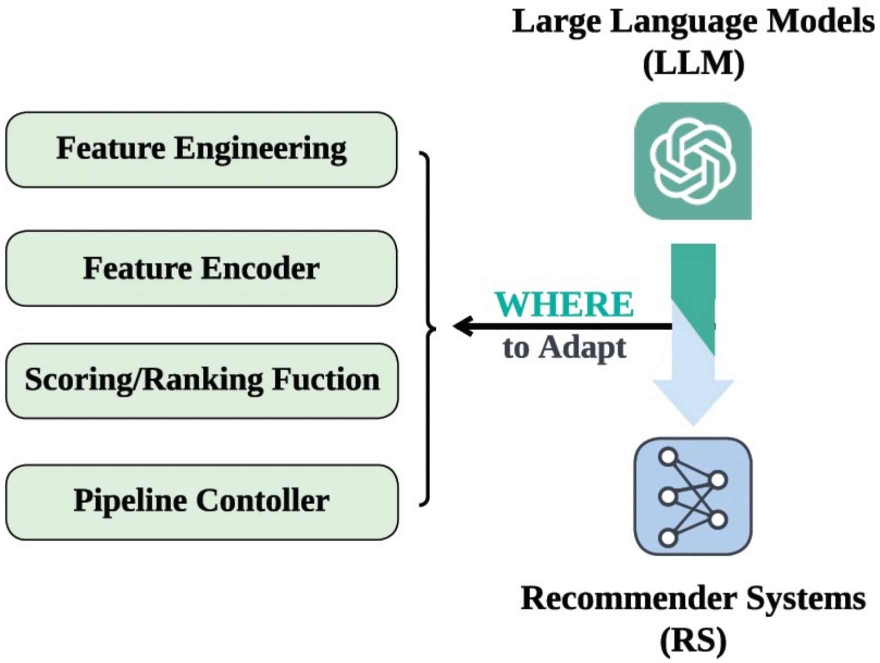


Image credit to [2]

[1] Wu, Likang, et al. "A Survey on Large Language Models for Recommendation." *arXiv preprint arXiv:2305.19860* (2023).
[2] Lin, Jianghao, et al. "How Can Recommender Systems Benefit from Large Language Models: A Survey." *arXiv preprint arXiv:2306.05817* (2023).

LLM as Feature Encoder

- LLM is grounded to recommendation space by generating tokens for items
- Then these tokens are grounded to actual items in the actual item space

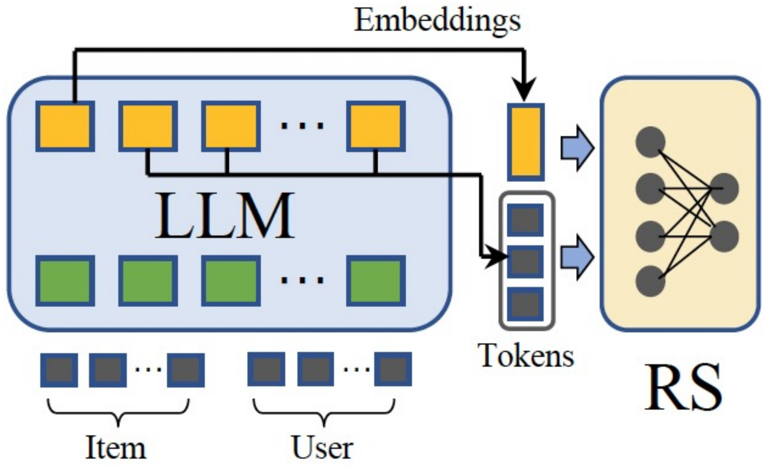


Image credit to [2]

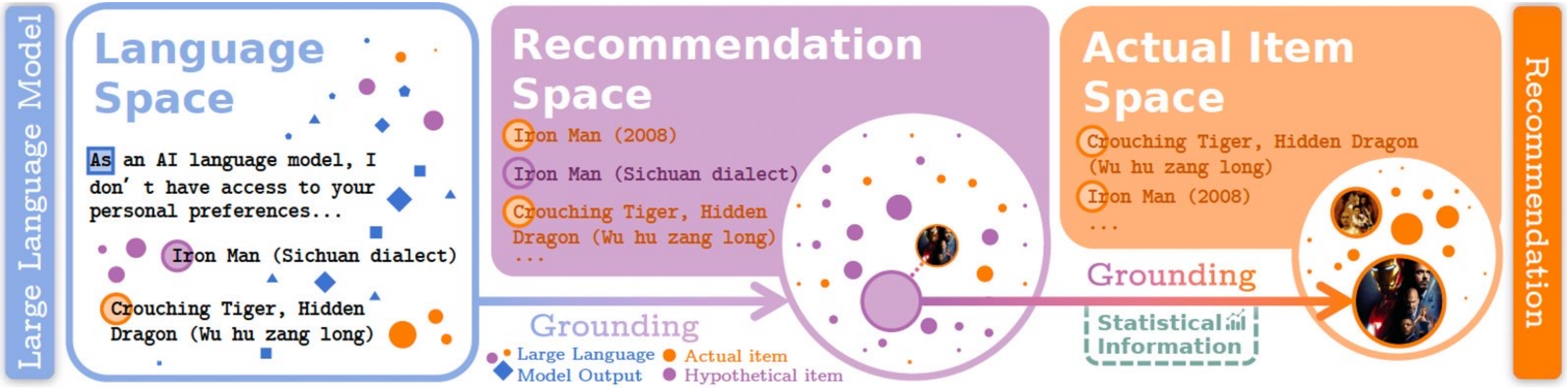
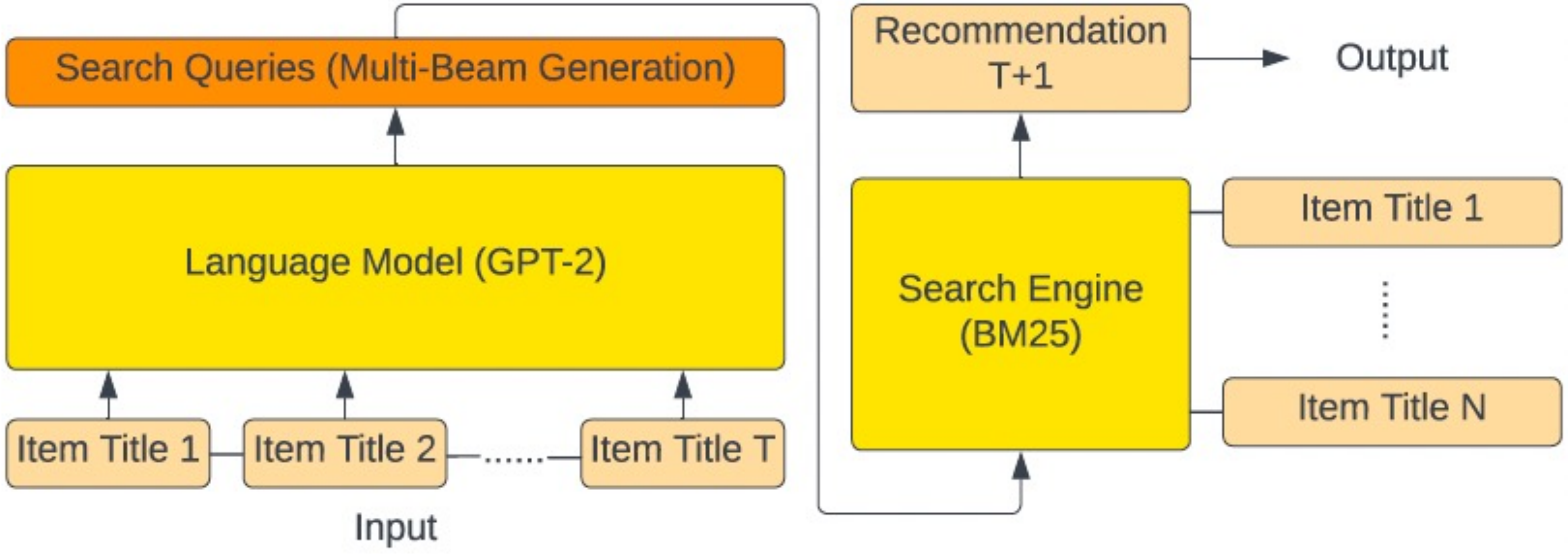


Image credit to [1]

[1] Bao, Keqin, et al. "A bi-step grounding paradigm for large language models in recommendation systems." *arXiv preprint arXiv:2308.08434* (2023).
[2] Wu, Likang, et al. "A Survey on Large Language Models for Recommendation." *arXiv preprint arXiv:2305.19860* (2023).

LLM as Feature Encoder

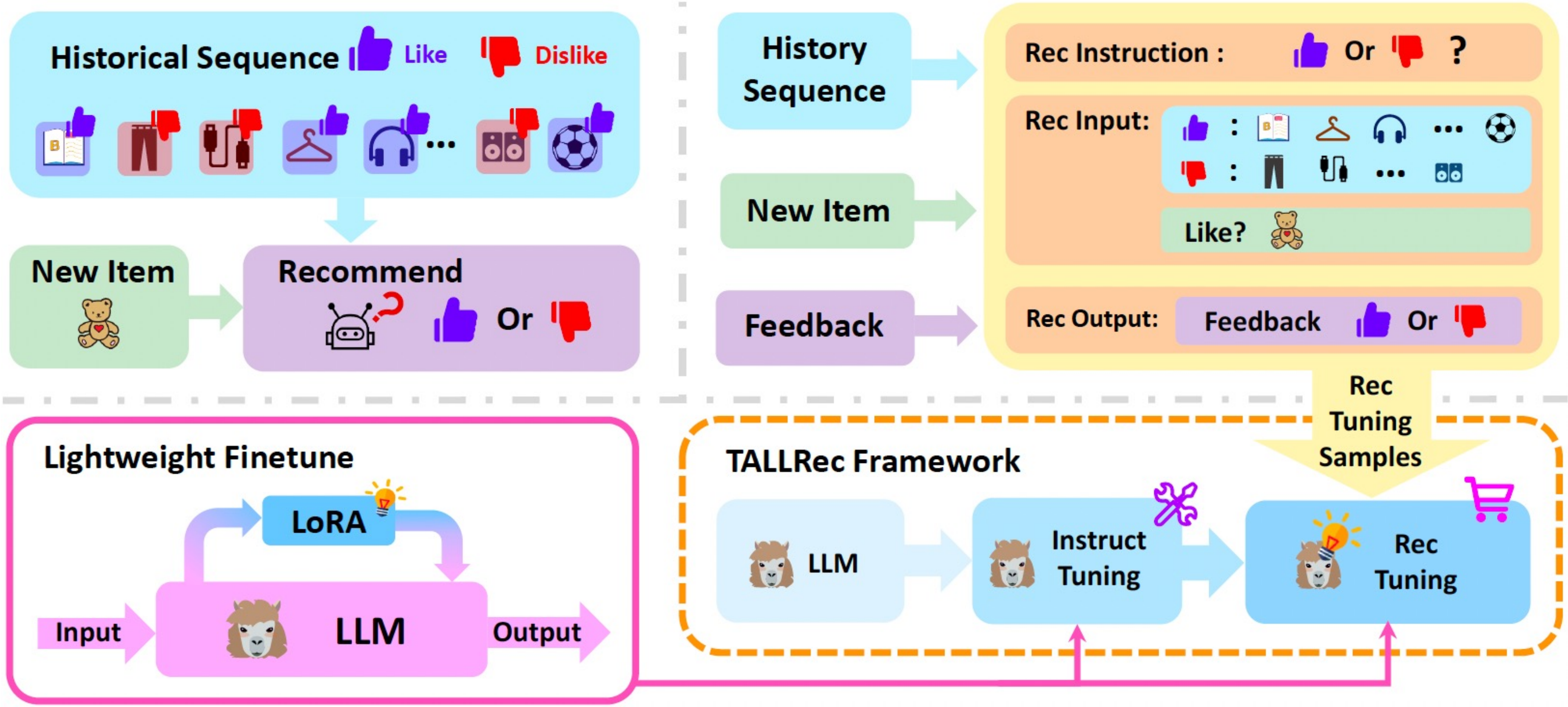
- Instruct LLM to generate search queries
- Then a searching algorithm is applied to retrieve items based on the queries



[1] Li, Jinming, et al. "GPT4Rec: A generative framework for personalized recommendation and user interests interpretation." *arXiv preprint arXiv:2304.03879* (2023).

LLM as Scoring Function

- Instruct LLM to generate a binary score (like or dislike) for each item
  - Discriminative as traditional recommenders



[1] Bao, Keqin, et al. "Tallrec: An effective and efficient tuning framework to align large language model with recommendation." *arXiv preprint arXiv:2305.00447* (2023).

RUTGERS

LLM as Ranking Function

- Provide LLM with candidates from another RS for re-ranking

Chain of thought

1. Preference inference
2. Preferred item selection
3. Recommendation

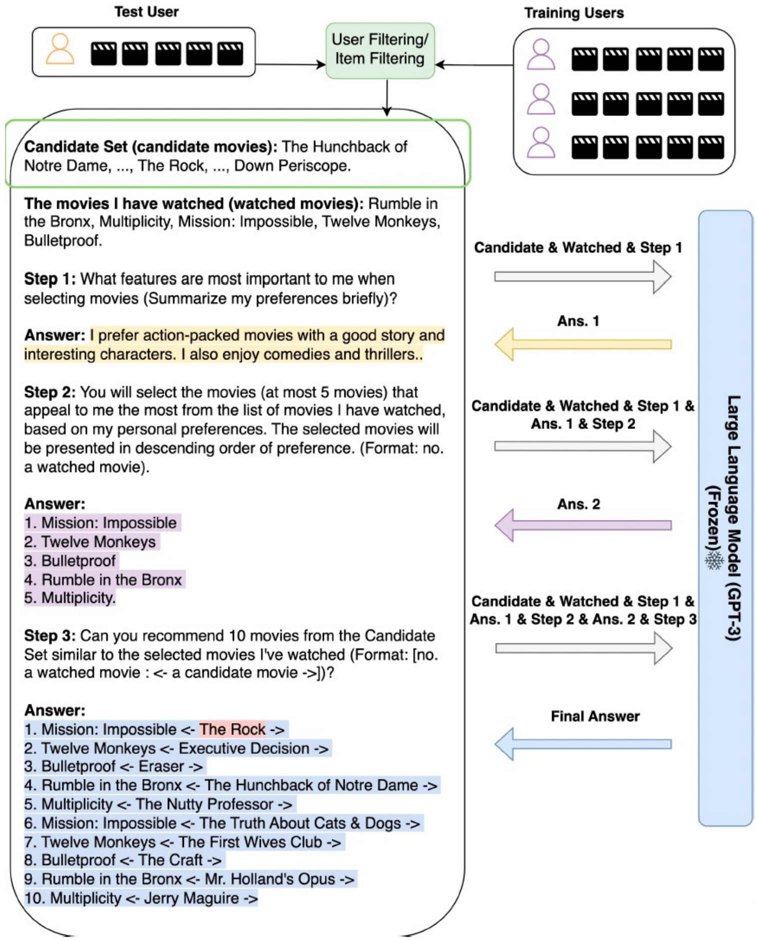


Image credit to NIR [1]

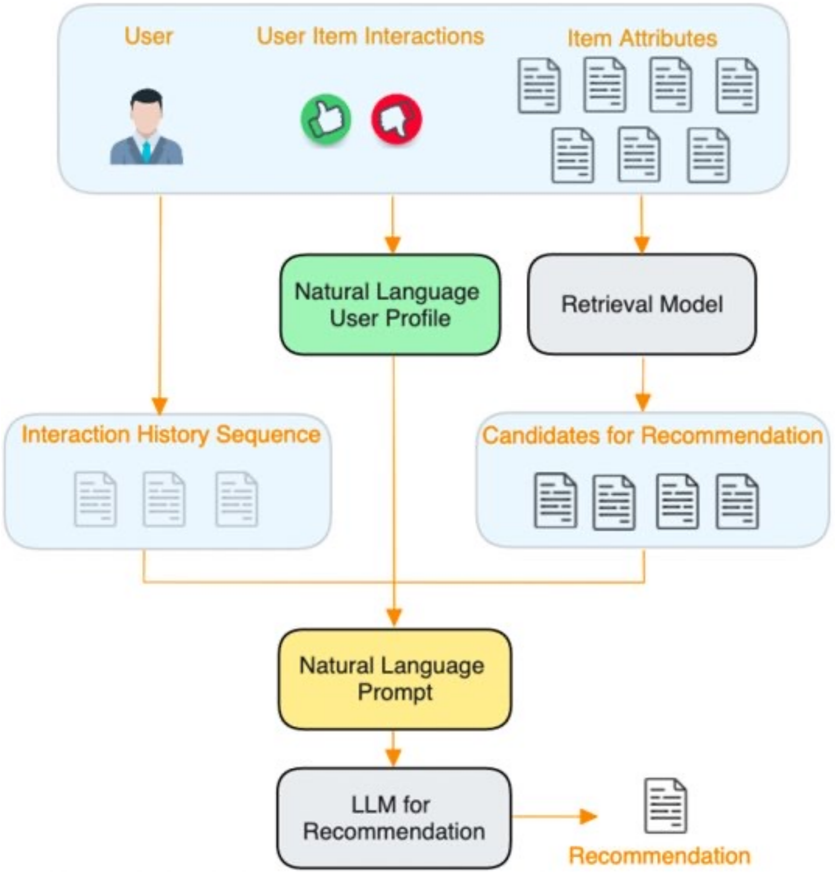
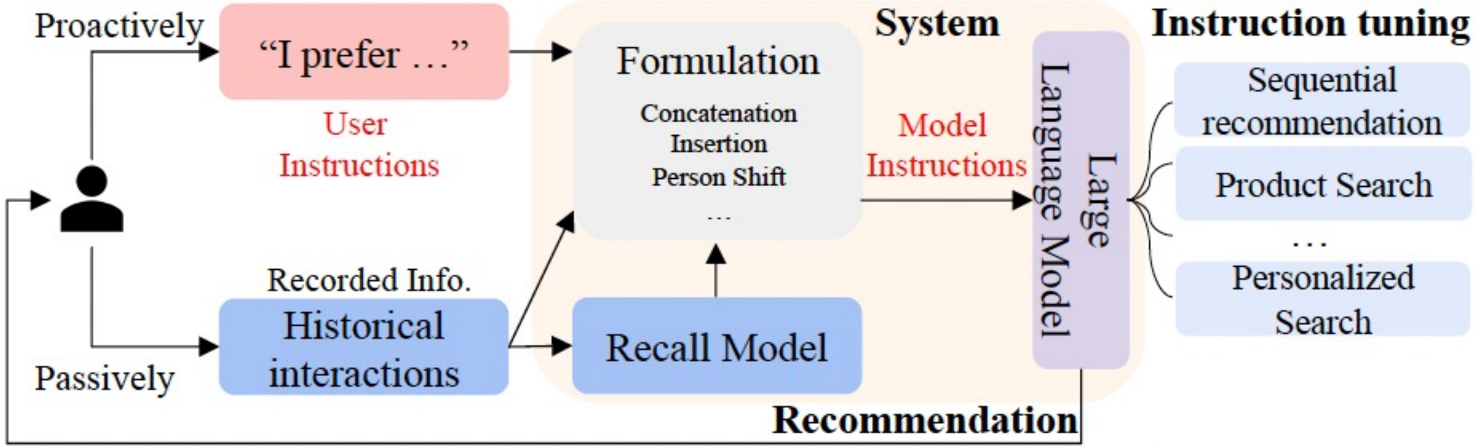


Image credit to PALR [2]

[1] Wang, Lei, and Ee-Peng Lim. "Zero-Shot Next-Item Recommendation using Large Pretrained Language Models." *arXiv preprint arXiv:2304.03153* (2023).
[2] Chen, Zheng. "PALR: Personalization Aware LLMs for Recommendation." Gen-IR@SIGIR 2023: The First Workshop on Generative Information Retrieval (2023).

LLM as Ranking Function

- LLM takes candidates from a Recall model for re-ranking
  - Design prompts for different recommendation settings

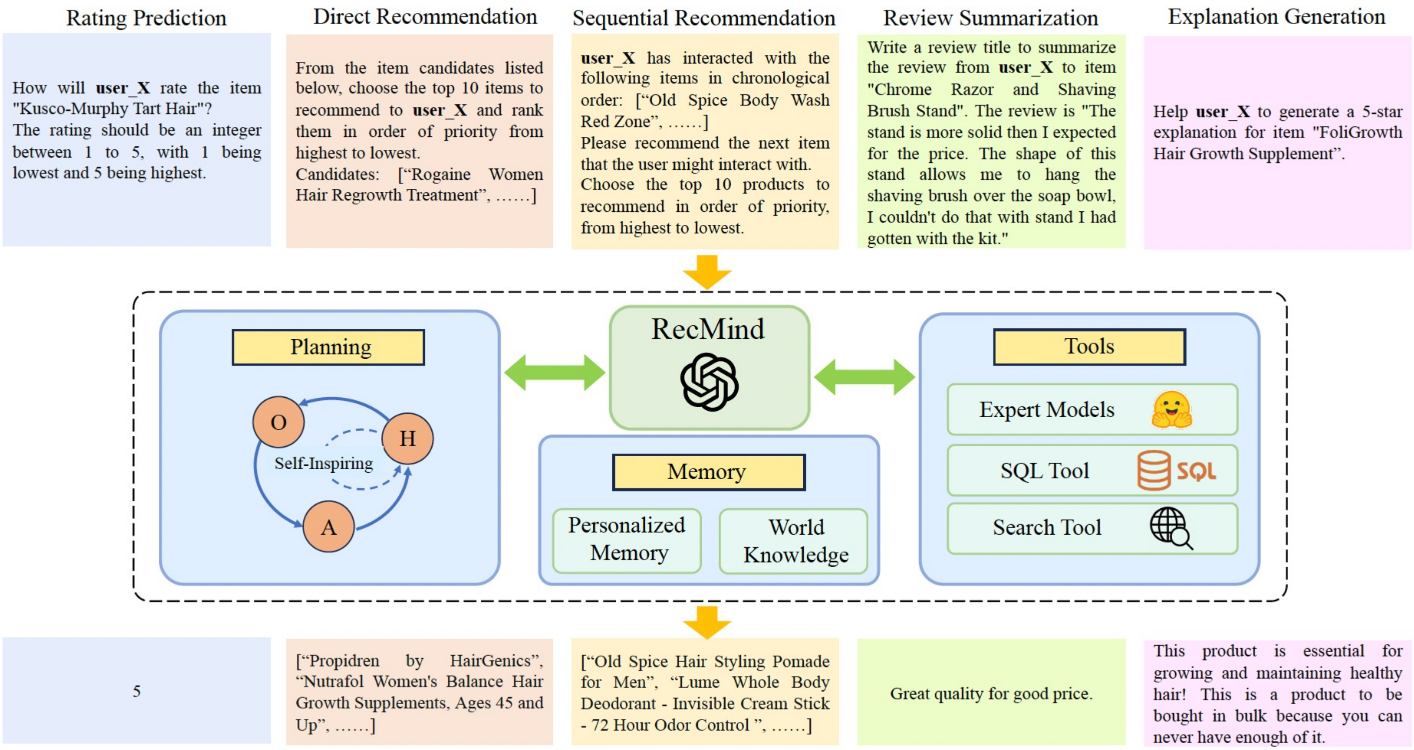


Instantiation	Model Instructions
$\langle P_1, I_0, T_0 \rangle$	The user has purchased these items: <historical interactions>. Based on this information, is it likely that the user will interact with <target item> next?
$\langle P_2, I_0, T_3 \rangle$	You are a search engine and you meet a user's query: <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.

[1] Zhang, Junjie, et al. "Recommendation as instruction following: A large language model empowered recommendation approach." *arXiv preprint arXiv:2305.07001* (2023).

LLM as Pipeline Controller

- Break each task into several planning steps
  - Thought, action and observation
- Control personalized memory and world knowledge
- Perform specific tasks with tools, e.g., task-specific models



[1] Wang, Yancheng, et al. "RecMind: Large Language Model Powered Agent For Recommendation." *arXiv preprint arXiv:2308.14296* (2023).

Recommendation Tasks

- Rating Prediction
- Sequential Recommendation
- Top-N Recommendation
- Explanation Generation
- Review Summarization
- Review Generation
- **Conversational Recommendation**

Conversational Recommendation

- LLM as the whole conversational recommender
  - T: Task description
  - F: Format requirement
  - S: Conversational context

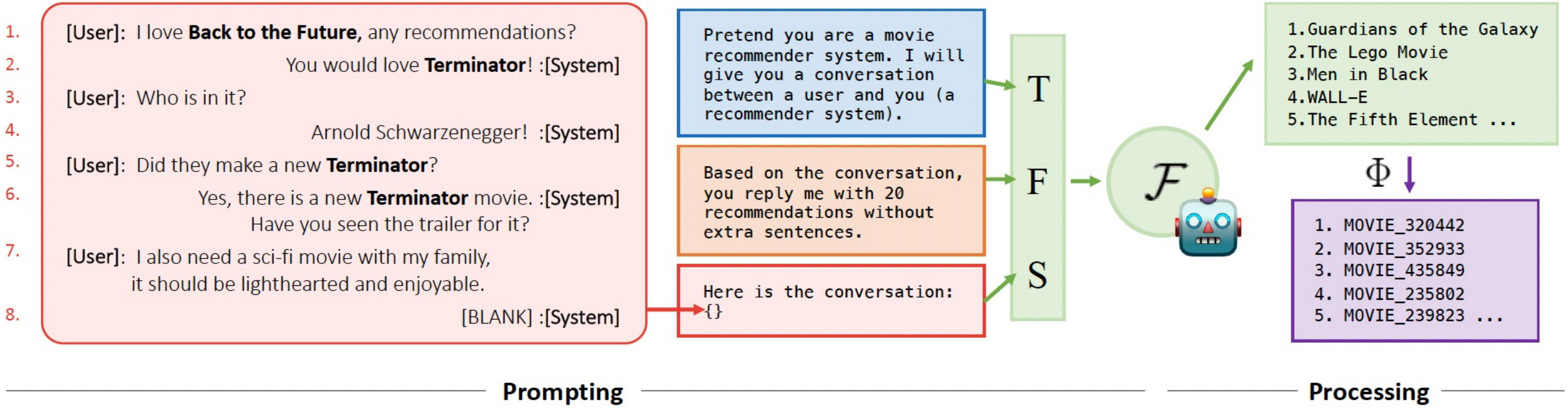


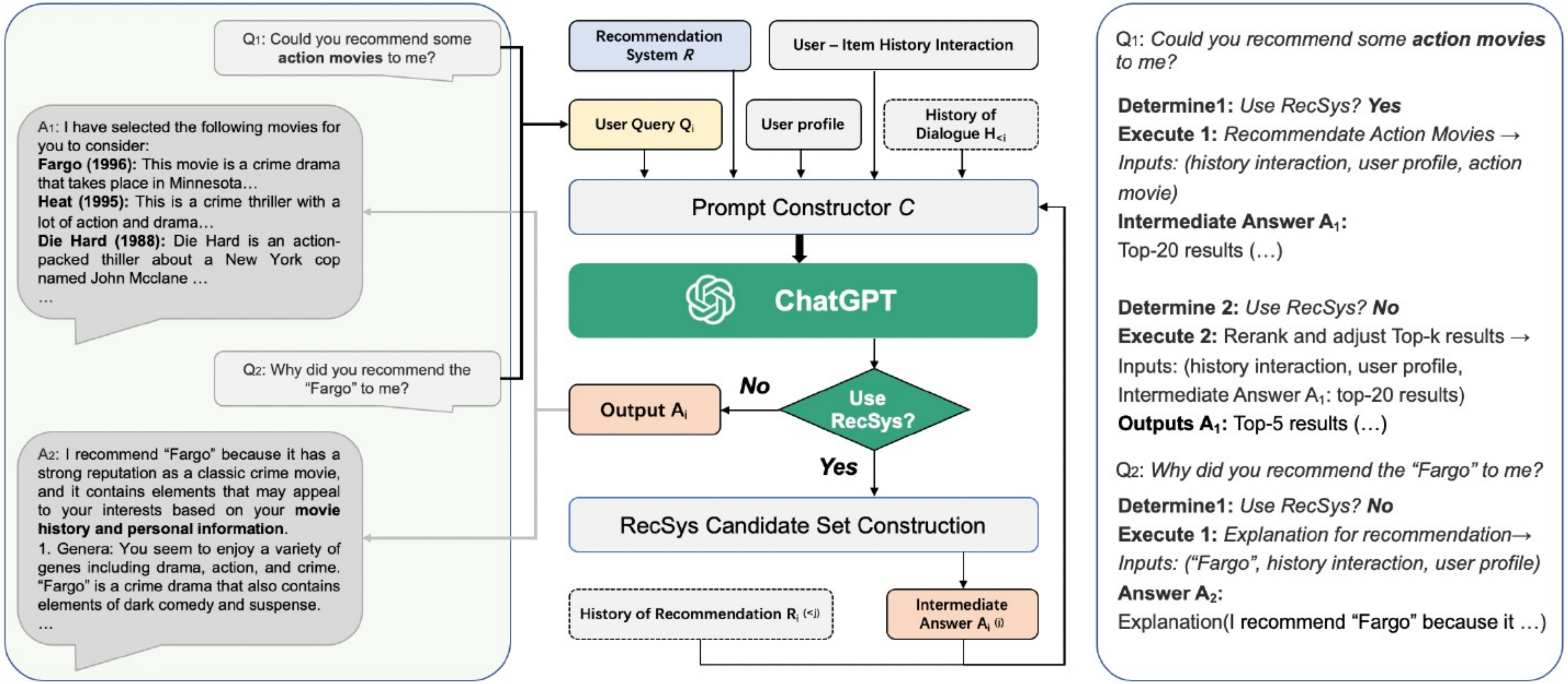
Image credit to [1]

[1] He, Zhankui, et al. "Large Language Models as Zero-Shot Conversational Recommenders." CIKM 2023.
[2] Cui, Zeyu, et al. "M6-rec: Generative pretrained language models are open-ended recommender systems." *arXiv preprint arXiv:2205.08084* (2022).

RUTGERS

Conversational Recommendation

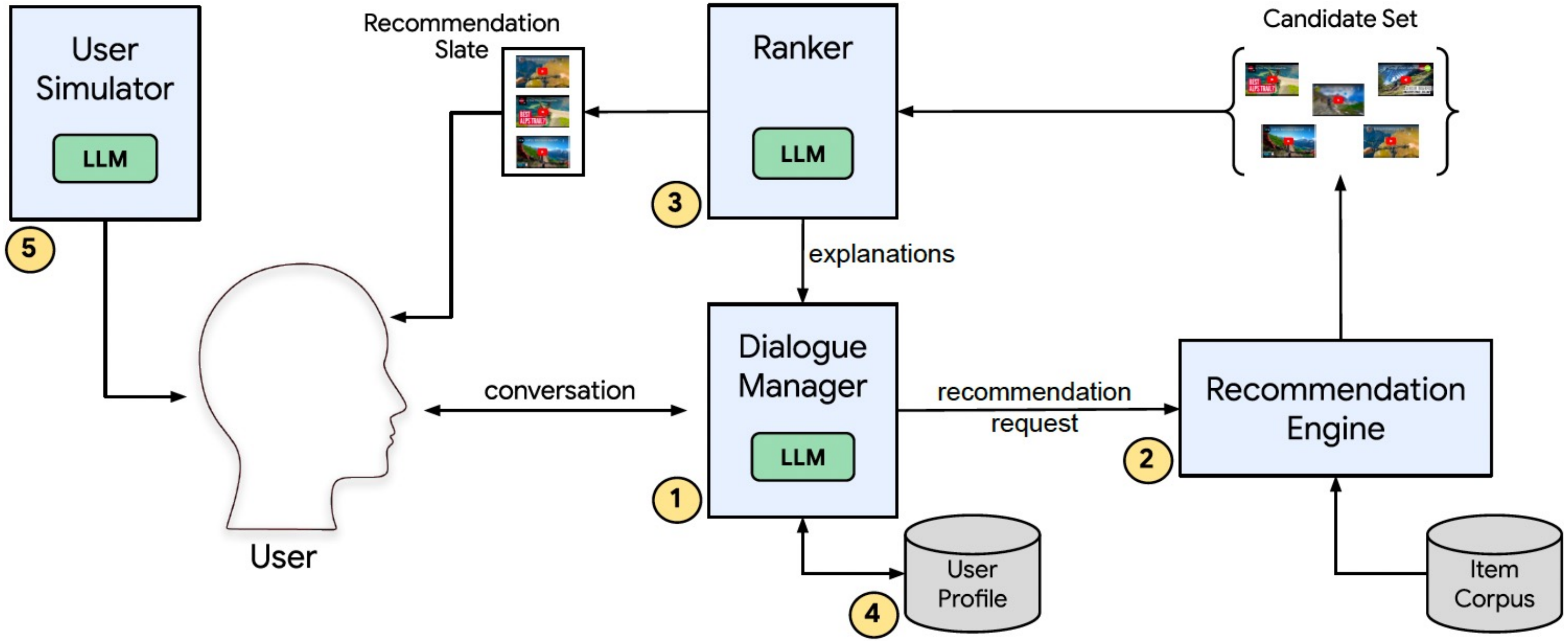
- LLM as dialogue manager that merges various types of info
  - Recommendations (from another model)
  - Dialogue history



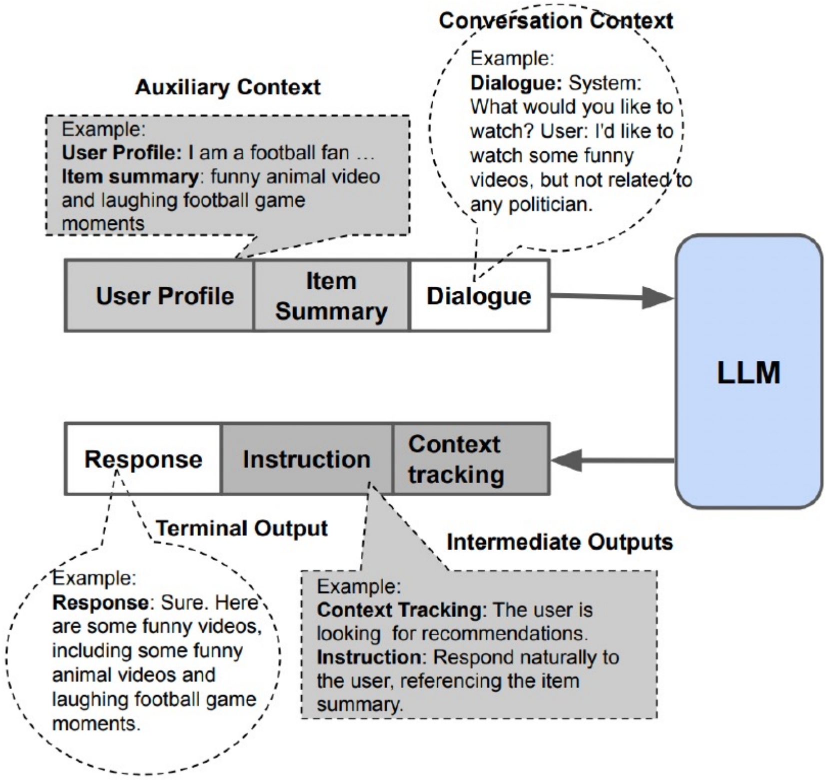
[1] Gao, Yunfan, et al. "Chat-rec: Towards interactive and explainable llms-augmented recommender system." *arXiv preprint arXiv:2303.14524* (2023).

Conversational Recommendation

- Multiple LLMs play separate roles
  - Dialogue Manager
  - Ranking Function
  - User Simulator



Dialogue Manager



[1] Friedman, Luke, et al. "Leveraging Large Language Models in Conversational Recommender Systems." *arXiv preprint arXiv:2305.07961* (2023).

Evaluation Protocols

- Recommendation
  - RMSE and MAE for rating prediction
  - NDCG, Precision and Recall for top-N and sequential recommendation
  - Online A/B test
- Generation
  - BLEU and ROUGE for text similarity
    - Overly stress the matching between generation and ground-truth [1]
    - Advanced metrics are needed
  - Human Evaluation

[1] Wang, Xiaolei, et al. "Rethinking the Evaluation for Conversational Recommendation in the Era of Large Language Models." *arXiv preprint arXiv:2305.13112* (2023).

Trustworthy LLMs for Recommendation

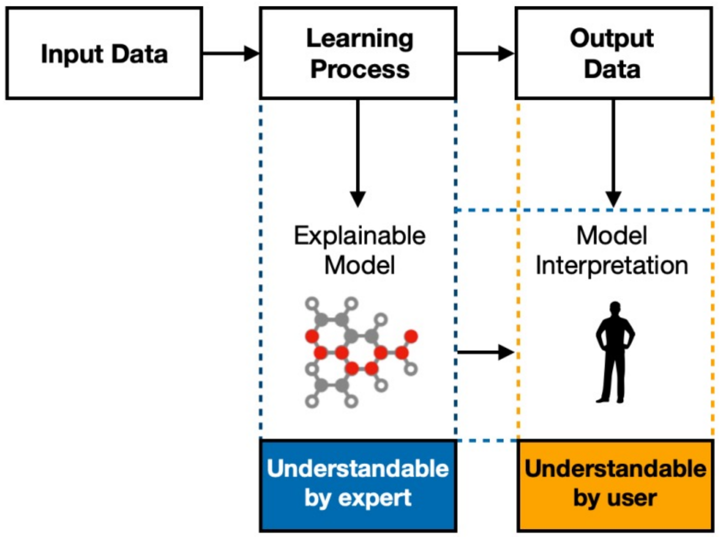
Trustworthy LLM4RS

- Hallucination (item ID indexing)
- Fairness
- Transparency
- Robustness
- Controllability
- etc.

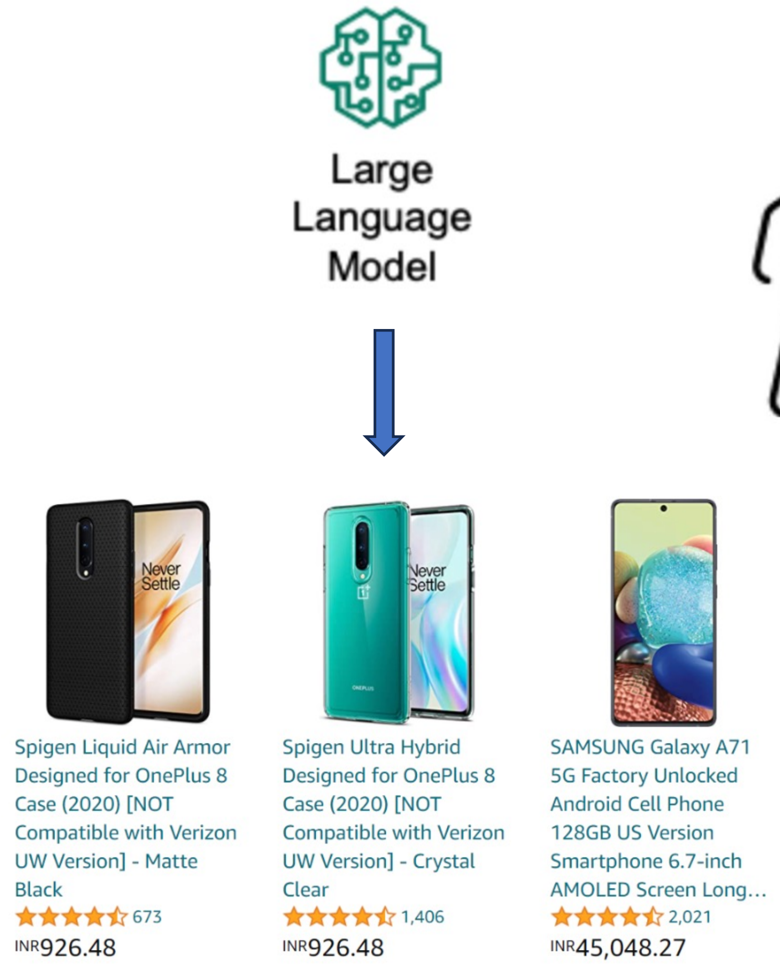
RUTGERS

Fairness

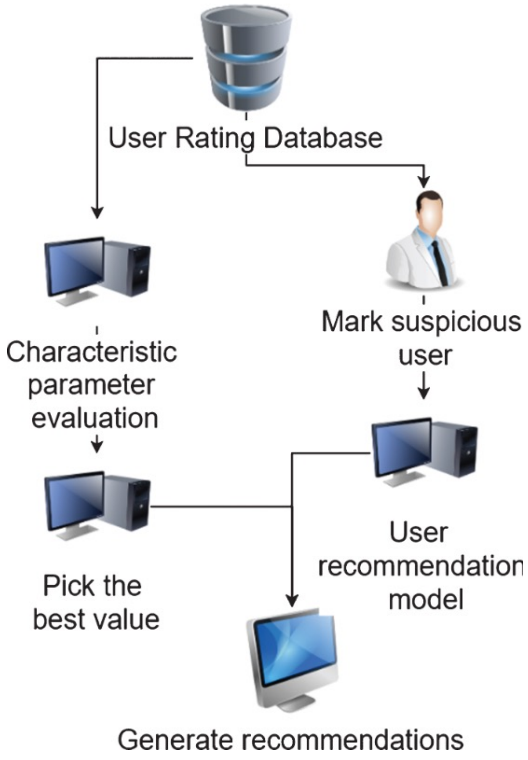
Transparency



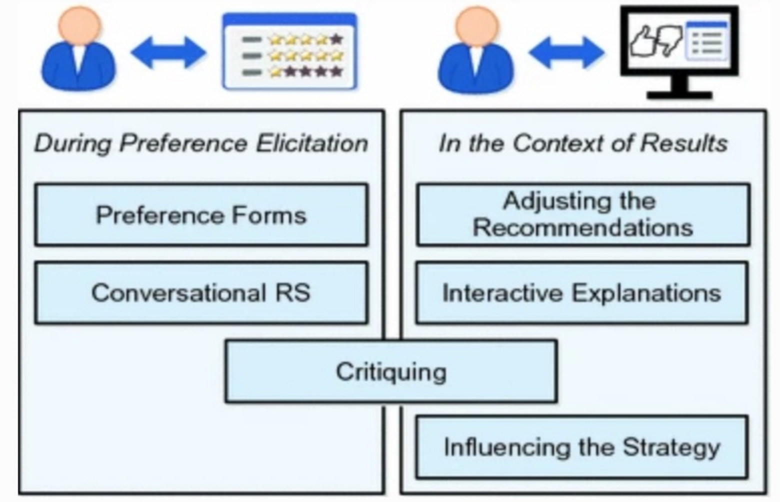
Hallucination



Robustness



Controllability



Hallucination: Item Generation

- LLM-based Generative Recommendation Paradigm
  - We want to directly generate the recommended item
  - Avoid one-by-one ranking score calculation

- However, item descriptions can be very long
  - e.g., product description: >100 words
  - e.g., news article: >1,000 words

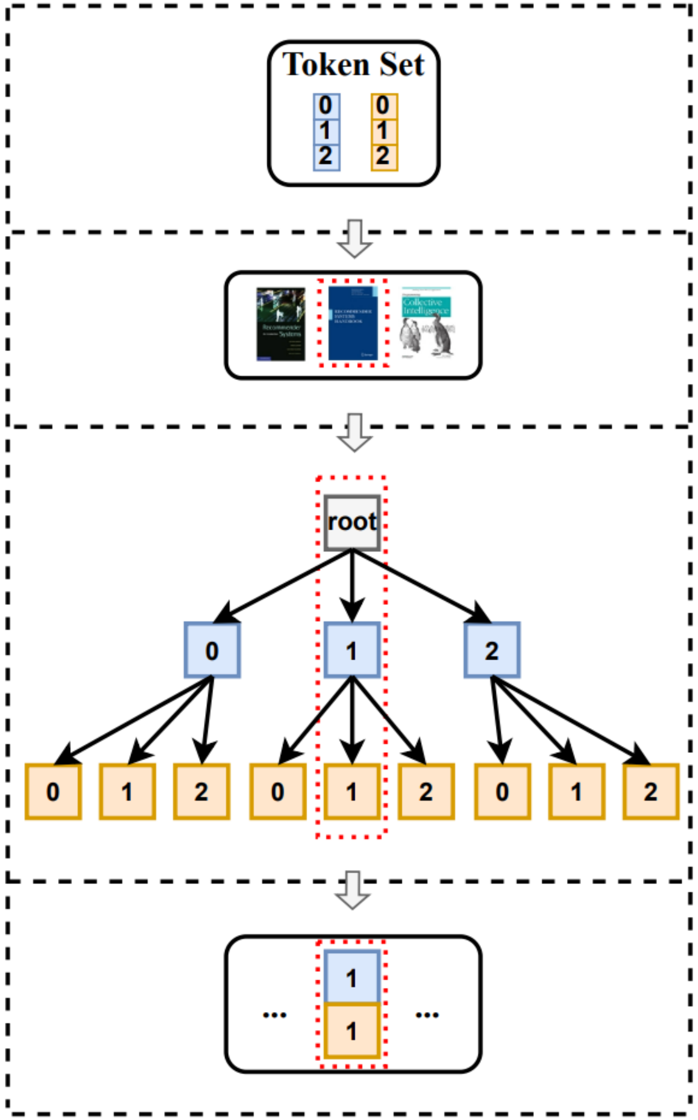
Hallucination: Item Generation

- Generating long text is difficult, especially for recommendation
  - Hallucination problem
  - Generated text does not correspond to a real existing item in database
  - Calculating similarity between generated text and item text?
  - Goes back to one-by-one similarity calculation for ranking!

- Item ID: A short sequence of tokens for an item
  - Easy generation, and can be indexed!

- Item ID can take various forms
  - A sequence of numerical tokens <73><91><26>
  - A sequence of word tokens <the><lord><of><the><rings>

Why Item IDs can eliminate hallucination?



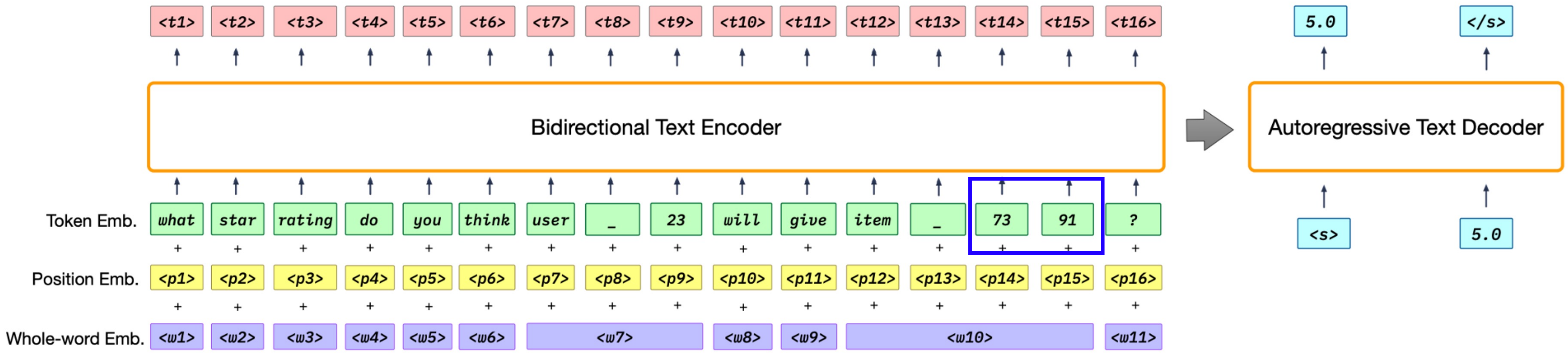
With item indices consisting of a limited vocabulary and known structure, we can constrain the beam search over limited allowed tokens for every generation step.

Thus, hallucination will be eliminated.

picture credited to: Li, Lei, et al. "Large Language Models for Generative Recommendation: A Survey and Visionary Discussions." *arXiv preprint arXiv:2309.01157* (2023).

How to Index Items?

- Item ID: item needs to be represented as a sequence of tokens
  - e.g., an item represented by two tokens <73> <91>



- Different item indexing gives very different performance

How to Index Items (create Item IDs)

- Three properties for good item indexing methods
  - Items are distinguishable (different items have different IDs)
  - Similar items have similar IDs (more shared tokens in their IDs)
  - Dissimilar items have dissimilar IDs (less shared tokens in their IDs)
- Three naïve Indexing methods
  - Random ID (RID): Item ⟨73⟩⟨91⟩, item ⟨73⟩⟨12⟩, …
  - Title as ID (TID): Item ⟨the⟩⟨lord⟩⟨of⟩⟨the⟩⟨rings⟩, …
  - Independent ID (IID): Item ⟨1364⟩, Item ⟨6321⟩, …

Method	Amazon Sports				Amazon Beauty				Yelp			
	HR@5	NCDG@5	HR@10	NCDG@10	HR@5	NCDG@5	HR@10	NCDG@10	HR@5	NCDG@5	HR@10	NCDG@10
SASRec	0.0233	0.0154	0.0350	0.0192	0.0387	0.0249	0.0605	0.0318	0.0170	0.0110	0.0284	0.0147
S^3-Rec	0.0251	**0.0161**	0.0385	**0.0204**	0.0387	0.0244	**0.0647**	0.0327	0.0201	0.0123	0.0341	0.0168
RID	0.0208	0.0122	0.0288	0.0153	0.0213	0.0178	0.0479	0.0277	0.0225	**0.0159**	0.0329	0.0193
TID	0.0000	0.0000	0.0000	0.0000	0.0182	0.0132	0.0432	0.0254	0.0058	0.0040	0.0086	0.0049
IID	**0.0268**	0.0151	**0.0386**	0.0195	**0.0394**	**0.0268**	0.0615	**0.0341**	**0.0232**	0.0146	**0.0393**	**0.0197**

RUTGERS

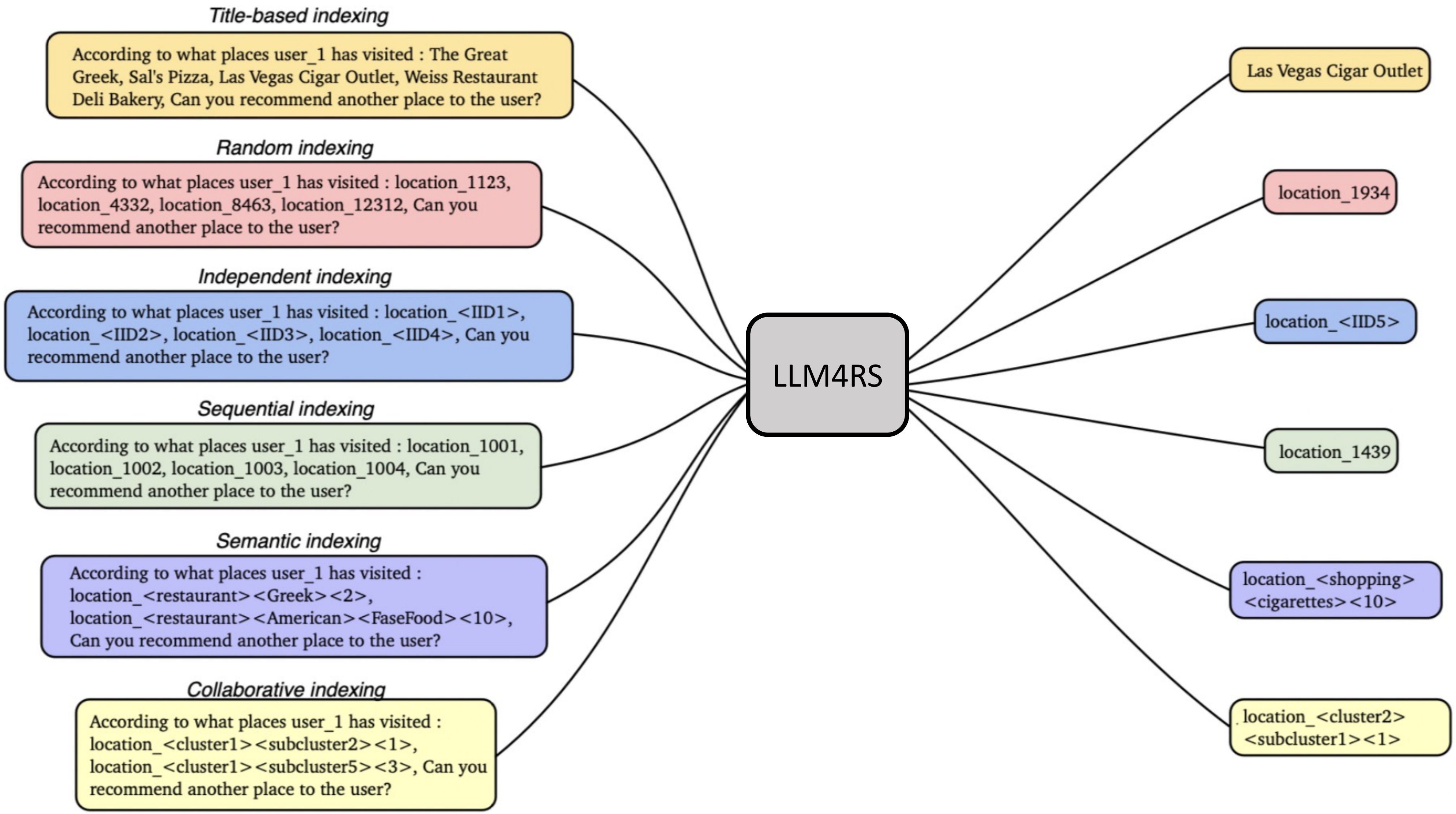
How to Index Items (create Item IDs)

- Three naïve Indexing methods
  - Random ID (RID): Item <73><91>, item <73><29>, …
    - Very different items may share the same tokens
    - Mistakenly making them semantically similar

  - Title as ID (TID): Item <the><lord><of><the><rings>
  - Very different movies may share similar titles
      - **Inside Out** (animation) and **Inside Job** (documentary)
      - **The Lord of the Rings** (epic fantasy) and **The Lord of War** (crime drama)

  - Independent ID (IID): Item <1364>, Item <6321>, …
    - Too many out-of-vocabulary (OOV) new tokens need to learn
    - Computationally unscalable

Meticulous Item Indexing Methods are Needed



RUTGERS

Sequential Indexing (SID)

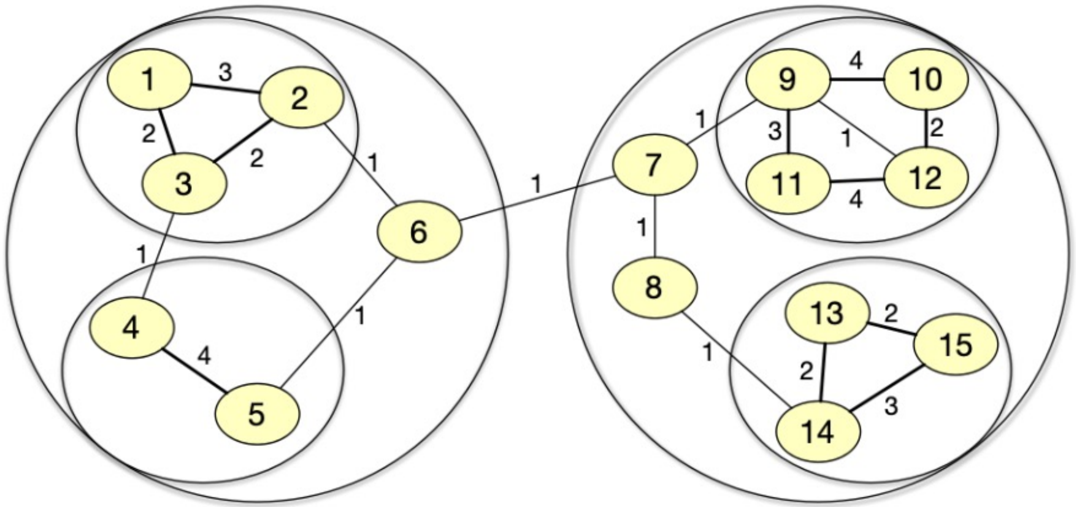
- Leverage the local co-appearance information between items

	Training Sequence										Validation	Testing
User 1	1001	1002	1003	1004	1005	1006	1007	1008	1009		1018	1019
User 2	1010	1011	1001	1012	1008	1009	1013	1014			1022	1023
User 3	1015	1016	1017	1007	1018	1019	1020	1021	1009		1015	1016
User 4	1022	1023	1005	1002	1006	1024					1002	1008
User 5	1025	1026	1027	1028	1029	1030	1024	1020	1021	1031	1033	1034

- After tokenization, co-appearing items share similar tokens
  - Item 1004: <100><4>
  - Item 1005: <100><5>

Collaborative Indexing (CID)

- Leverage the global co-appearance information between items
  - Spectral Matrix Factorization over the item-item co-appearance matrix
  - Hierarchical Spectral Clustering



(a) Recursive spectral clustering on item co-appearance graph

$$
\begin{array}{c}
 \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ \vdots
\end{array}
\begin{array}{c}
\begin{array}{ccccccc} 1 & 2 & 3 & 4 & 5 & 6 & \cdots \end{array} \\
\begin{bmatrix}
0 & 3 & 2 & 0 & 0 & 0 & \cdots \\
3 & 0 & 2 & 0 & 0 & 1 & \cdots \\
2 & 2 & 0 & 1 & 0 & 0 & \cdots \\
0 & 0 & 1 & 0 & 4 & 0 & \cdots \\
0 & 0 & 0 & 4 & 0 & 1 & \cdots \\
0 & 1 & 0 & 0 & 1 & 0 & \cdots \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots &
\end{bmatrix}
\end{array}
$$

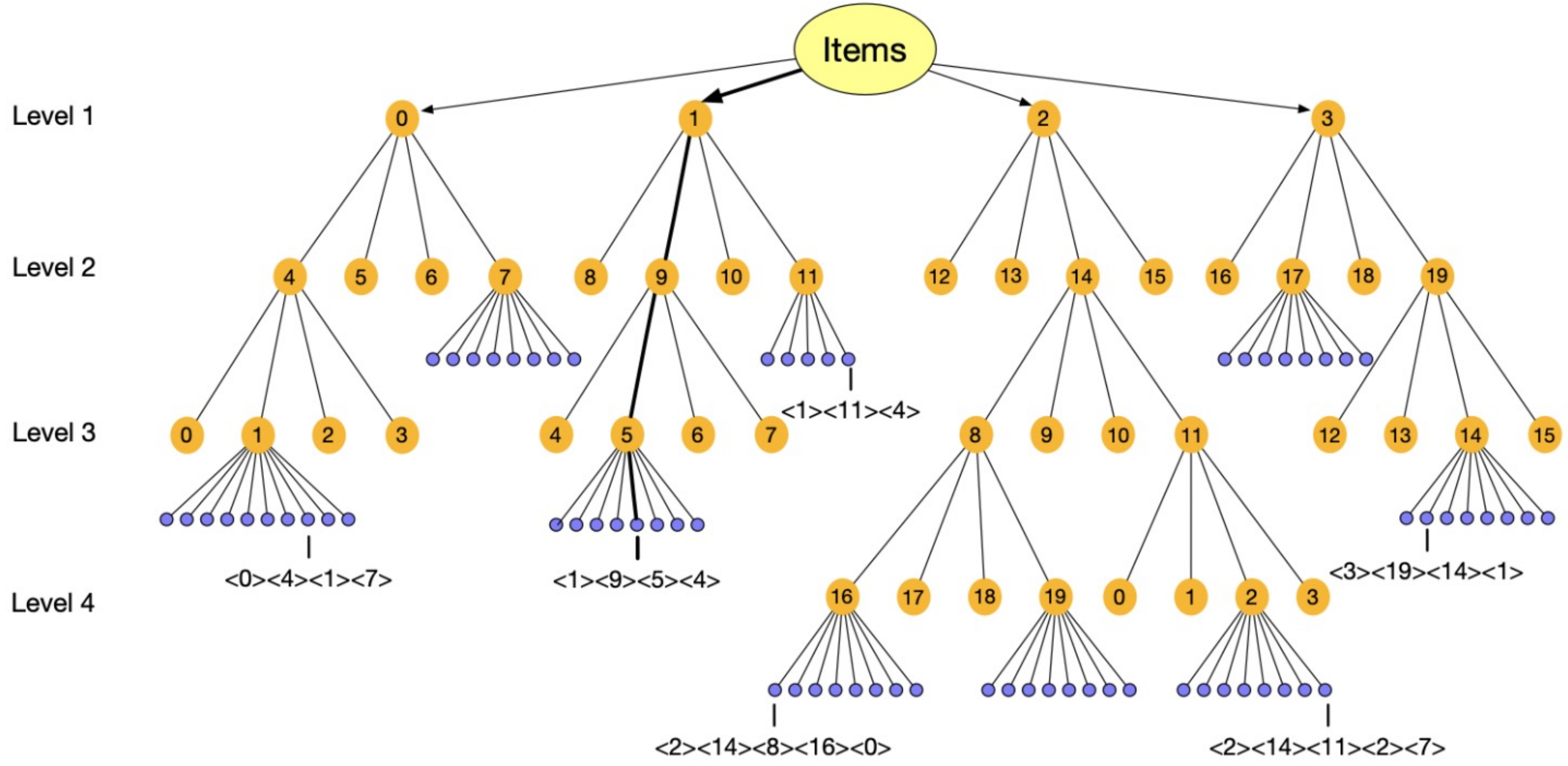
(b) Adjacency matrix

$$
\begin{array}{c}
 \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ \vdots
\end{array}
\begin{array}{c}
\begin{array}{ccccccc} 1 & 2 & 3 & 4 & 5 & 6 & \cdots \end{array} \\
\begin{bmatrix}
5 & -3 & -2 & 0 & 0 & 0 & \cdots \\
-3 & 6 & -2 & 0 & 0 & -1 & \cdots \\
-2 & -2 & 6 & -1 & 0 & 0 & \cdots \\
0 & 0 & 1 & 5 & -4 & 0 & \cdots \\
0 & 0 & 0 & -4 & 5 & -1 & \cdots \\
0 & 1 & 0 & 0 & -1 & 2 & \cdots \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots &
\end{bmatrix}
\end{array}
$$

(c) Laplacian matrix

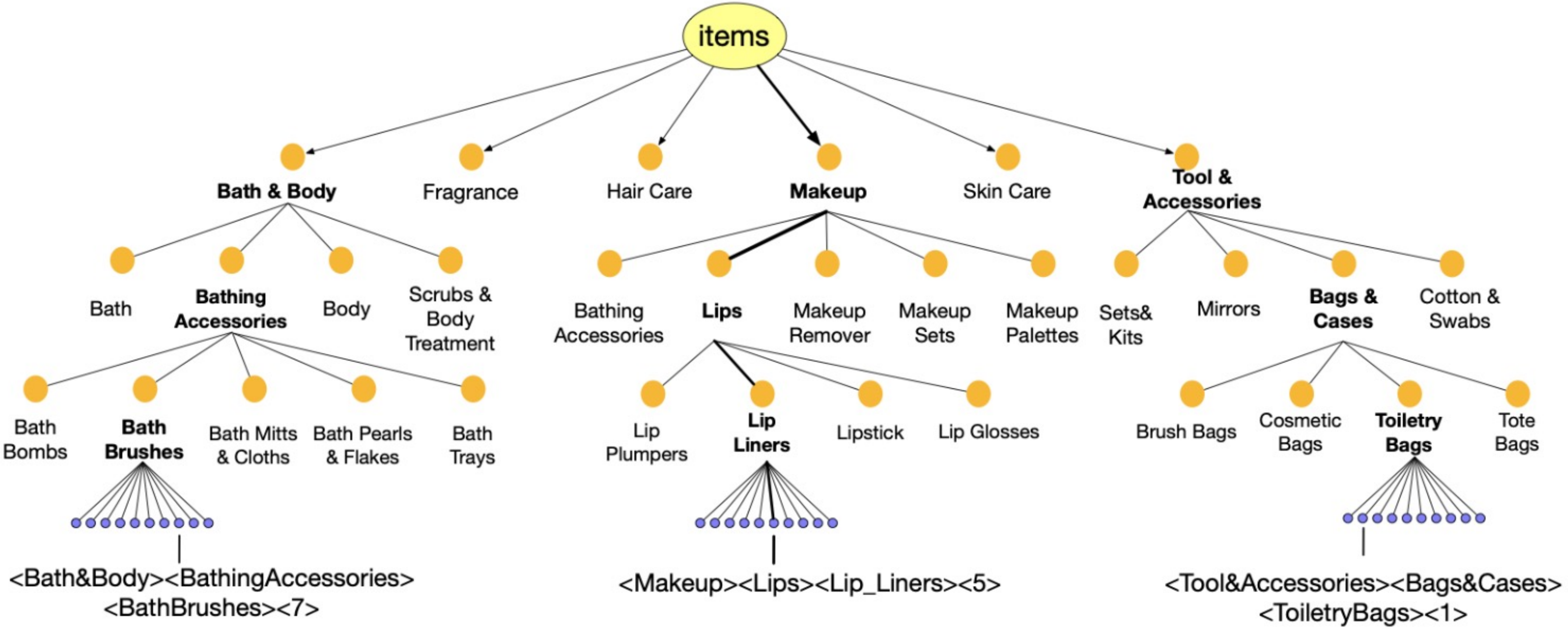
Collaborative Indexing (CID)

- Leverage the global co-appearance information between items
  - Root-to-Leaf Path-based Indexing
  - Items in the same cluster share more tokens



Semantic (Content-based) Indexing (SemID)

- Leverage the item content information for item indexing
  - e.g., the multi-level item category information in Amazon



Hybrid Indexing (HID)

- Concatenate more than one of the following indices
  - Random ID (RID)
  - Title as ID (TID)
  - Independent ID (IID)
  - Sequential ID (SID)
  - Collaborative ID (CID)
  - Semantic ID (SemID)

  - For example, if an item’s Semantic ID and Collaborative ID are as follows:
    - SemID: ⟨Makeup⟩⟨Lips⟩⟨Lip_Liners⟩⟨5⟩
    - CID: ⟨1⟩⟨9⟩⟨5⟩⟨4⟩
  - Then its Hybrid ID is ⟨Makeup⟩⟨Lips⟩⟨Lip_Liners⟩⟨1⟩⟨9⟩⟨5⟩⟨4⟩

Different Item Indexing Gives Different Performance

	Method	Amazon Sports				Amazon Beauty				Yelp			
		HR@5	NCDG@5	HR@10	NCDG@10	HR@5	NCDG@5	HR@10	NCDG@10	HR@5	NCDG@5	HR@10	NCDG@10
	Caser	0.0116	0.0072	0.0194	0.0097	0.0205	0.0131	0.0347	0.0176	0.015	0.0099	0.0263	0.0134
	HGN	0.0189	0.0120	0.0313	0.0159	0.0325	0.0206	0.0512	0.0266	0.0186	0.0115	0.0326	0.159
	GRU4Rec	0.0129	0.0086	0.0204	0.0110	0.0164	0.0099	0.0283	0.0137	0.0176	0.0110	0.0285	0.0145
	BERT4Rec	0.0115	0.0075	0.0191	0.0099	0.0203	0.0124	0.0347	0.0170	0.0051	0.0033	0.0090	0.0090
	FDSA	0.0182	0.0122	0.0288	0.0156	0.0267	0.0163	0.0407	0.0208	0.0158	0.0098	0.0276	0.0136
	SASRec	0.0233	0.0154	0.0350	0.0192	0.0387	0.0249	0.0605	0.0318	0.0170	0.0110	0.0284	0.0147
	S^3-Rec	0.0251	0.0161	0.0385	0.0204	0.0387	0.0244	0.0647	0.0327	0.0201	0.0123	0.0341	0.0168
Naïve indexing methods	RID	0.0208	0.0122	0.0288	0.0153	0.0213	0.0178	0.0479	0.0277	0.0225	0.0159	0.0329	0.0193
	TID	0.000	0.000	0.000	0.000	0.0182	0.0132	0.0432	0.0254	0.0058	0.0040	0.0086	0.0049
	IID	0.0268	0.0151	0.0386	0.0195	0.0394	0.0268	0.0615	0.0341	0.0232	0.0146	0.0393	0.0197
Advanced indexing methods	SID	0.0264	0.0186	0.0358	0.0216	0.0430	0.0288	0.0602	0.0368	**0.0346**	**0.0242**	**0.0486**	**0.0287**
	CID	0.0313	0.0224	0.0431	0.0262	0.0489	0.0318	0.0680	0.0357	0.0261	0.0171	0.0428	0.0225
	SemID	0.0274	0.0193	0.0406	0.0235	0.0433	0.0299	0.0652	0.0370	0.0202	0.0131	0.0324	0.0170
Hybrid indexing methods	SID+IID	0.0235	0.0161	0.0339	0.0195	0.0420	0.0297	0.0603	0.0355	0.0329	0.0236	0.0465	0.0280
	CID+IID	**0.0321**	**0.0227**	**0.0456**	**0.0270**	**0.0512**	**0.0356**	**0.0732**	**0.0427**	0.0287	0.0195	0.0468	0.0254
	SemID+IID	0.0291	0.0196	0.0436	0.0242	0.0501	0.0344	0.0724	0.0411	0.0229	0.0150	0.0382	0.0199
	SemID+CID	0.0043	0.0031	0.0070	0.0039	0.0355	0.0248	0.0545	0.0310	0.0021	0.0016	0.0056	0.0029

- Advanced indexing methods are better than naïve methods
- Some hybrid indexing can further improve performance

Fairness of LLM for Recommendation

1. Fairness of general LLM on critical domains (education, criminology, finance and healthcare) [1]

2. User-side fairness: UP5 [2], FaiRLLM benchmark [3]

3. Item-side fairness: popularity bias [4]

[1] Li, Yunqi, et al. "Fairness of ChatGPT." arXiv preprint arXiv:2305.18569 (2023).

[2] Hua, Wenyue, et al. "UP5: Unbiased Foundation Model for Fairness-aware Recommendation." *arXiv preprint arXiv:2305.12090* (2023).

[3] Zhang, Jizhi, et al. "Is chatgpt fair for recommendation? evaluating fairness in large language model recommendation." *arXiv preprint arXiv:2305.07609* (2023).

[4] Hou, Yupeng, et al. "Large language models are zero-shot rankers for recommender systems." *arXiv preprint arXiv:2305.08845* (2023).

Fairness of General LLM

- Fairness of ChatGPT on four critical domains [1]
  - Education, Criminology, Finance and Healthcare
  - Four Datasets
    - PISA (education), COMPAS (criminology)
    - German Credit (finance), Heart Disease (healthcare)
  - Five Fairness Evaluation Dimensions
    - Statistical Parity
    - Equal Opportunity
    - Equalized Odds
    - Overall Accuracy Equality
    - Counterfactual Fairness
- Main Observation
  - ChatGPT is fairer than small models such as regression and MLP classifier, though ChatGPT still has unfairness issues

[1] Li, Yunqi, et al. "Fairness of ChatGPT." arXiv preprint arXiv:2305.18569 (2023).

User-side Fairness method

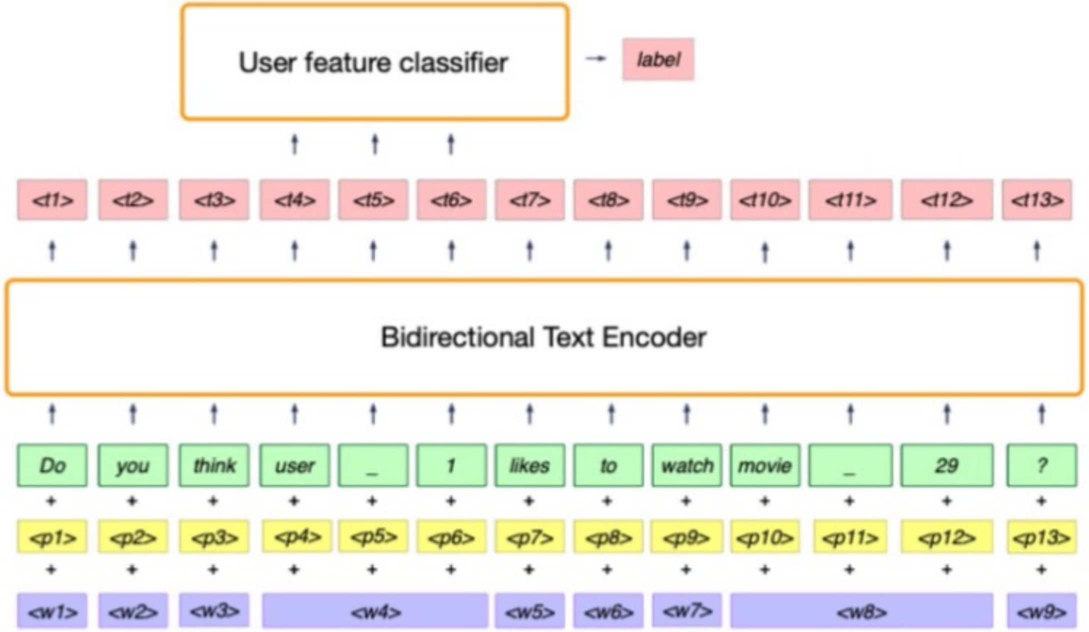
Users want to be treated fairly, independent on their sensitive user features.

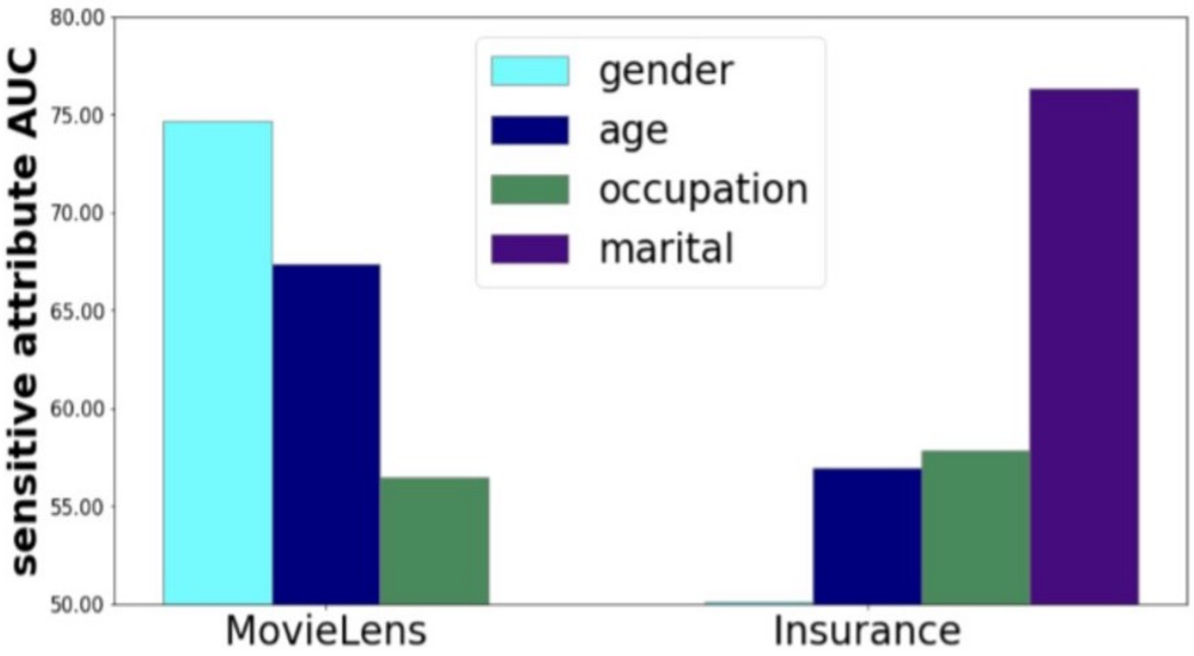
Are pretrained LLM4RS fair on recommending items?

[1] Li, Yunqi, et al. "Towards personalized fairness based on causal notion." *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 2021.

How to make sure recommendations are fair?

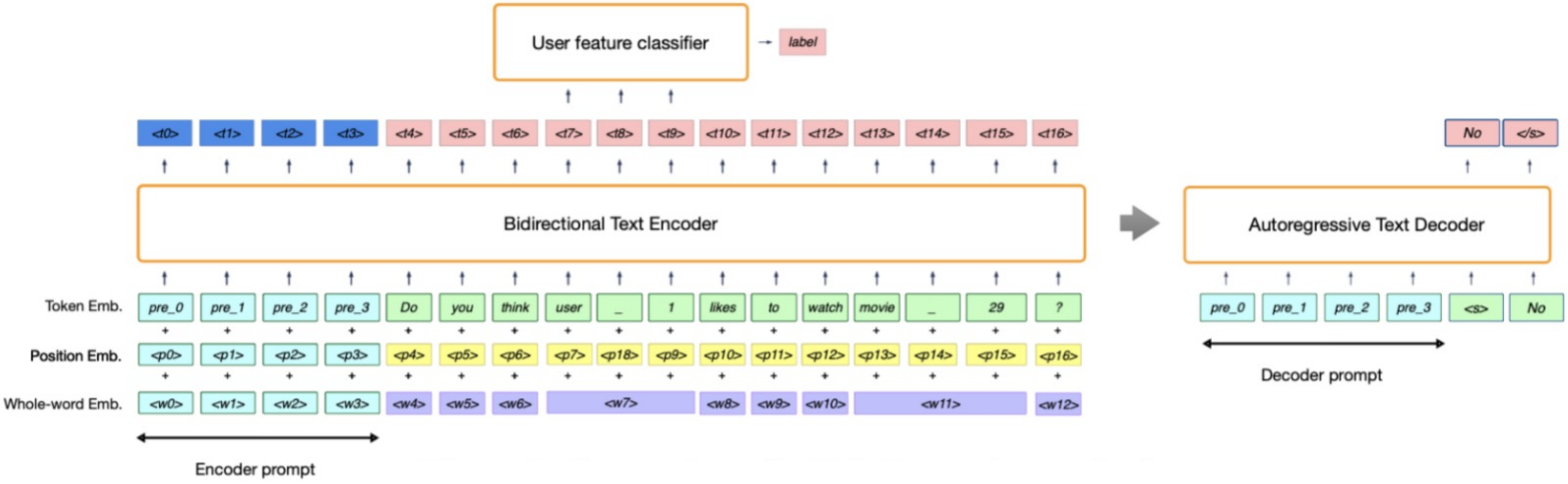
As long as the input representation is independent of user sensitive features, then the generated recommendations are independent of sensitive features.





The AUC scores on various user features show that the user sensitive features are incorporated in the input representations, leading to unfair recommendation.

Fairness Prompts for LLM



For each feature k, the adversarial loss is: $L_k = \sum_u L_{rec}^k - \lambda_k \cdot L_{dis}^k$

Single-feature fairness results

Dataset	MovieLens									Insurance								
Attribute	**Gender**			**Age**			**Occupation**			**Age**			**Marital**			**Occupation**		
Model	C-PMF	C-SX	CFP	C-PMF	C-SX	CFP	C-PMF	C-SX	CFP	C-PMF	C-SX	CFP	C-PMF	C-SX	CFP	C-PMF	C-SX	CFP
↑ Hit@1	**16.73**	13.96	16.38	17.42	13.87	**21.22**	15.60	14.06	**21.00**	67.61	71.14	**82.53**	66.68	71.50	**81.03**	68.51	71.09	**82.53**
↑ Hit@3	34.03	29.56	**35.04**	34.20	29.61	**39.22**	34.36	29.56	**38.50**	73.25	83.23	**92.68**	74.23	83.00	**90.58**	74.09	82.23	**92.68**
↑ Hit@5	46.72	40.05	**47.33**	46.72	39.25	**48.85**	46.80	39.82	**49.35**	78.86	86.50	**96.44**	76.57	86.12	**94.76**	76.48	88.00	**96.44**
↑ Hit@10	65.32	56.02	**65.82**	65.18	55.42	**67.30**	65.33	56.02	**69.49**	85.98	92.65	**98.89**	85.99	96.50	**97.66**	85.95	93.27	**98.89**
↓ AUC	56.62	70.80	**54.19**	62.55	79.26	**52.91**	56.01	57.02	**50.00**	50.81	51.26	**50.09**	**52.10**	56.23	52.19	54.40	**52.09**	53.28

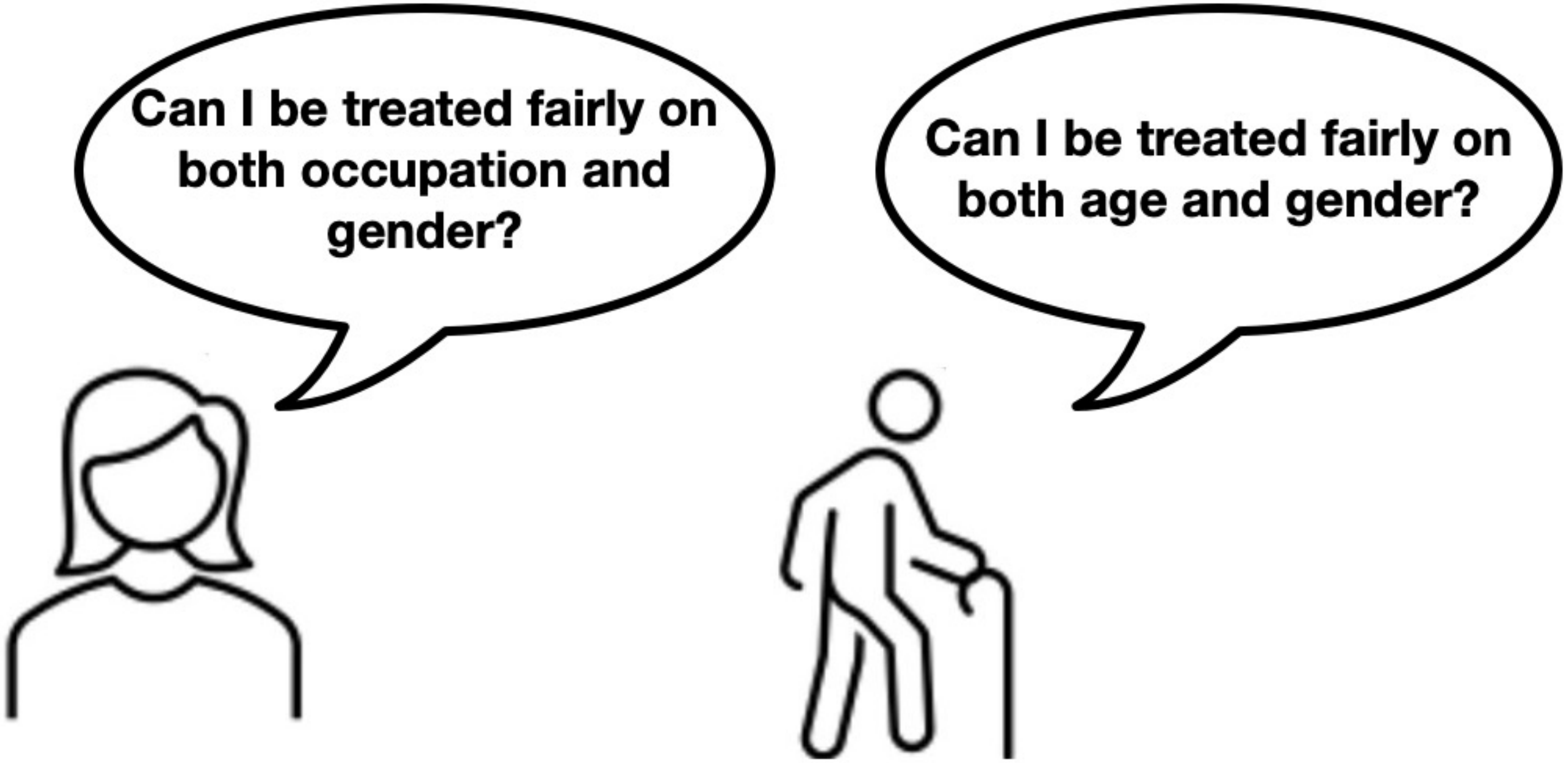
Results of single-attribute fairness-aware prompting on matching-based models (%)

Dataset	MovieLens									Insurance								
Attribute	**Gender**			**Age**			**Occupation**			**Age**			**Marital**			**Occupation**		
Model	S-SAS	S-B4	CFP	S-SAS	S-B4	CFP	S-SAS	S-B4	CFP	S-SAS	S-B4	CFP	S-SAS	S-B4	CFP	S-SAS	S-B4	CFP
↑ Hit@1	20.87	23.48	**26.82**	22.95	27.98	**31.23**	18.90	24.33	**31.66**	69.40	81.20	**82.08**	70.10	75.33	**80.63**	70.09	81.20	**82.62**
↑ Hit@3	41.64	42.09	**45.18**	44.10	49.32	**51.18**	20.84	43.29	**50.73**	80.05	**93.33**	92.62	80.38	84.54	**90.16**	80.38	**93.33**	92.65
↑ Hit@5	49.65	**55.77**	53.46	54.99	56.56	**58.91**	29.57	51.02	**58.26**	84.48	**97.50**	96.12	85.02	90.02	**94.33**	84.39	**97.50**	95.81
↑ Hit@10	60.82	62.43	**64.38**	66.00	69.38	**67.70**	43.87	59.74	**67.45**	88.34	**98.78**	98.37	88.49	94.34	**98.38**	88.91	**98.78**	98.54
↓ AUC	59.72	58.33	**54.19**	60.20	67.33	**52.91**	67.27	60.36	**50.00**	57.48	53.34	**51.23**	66.51	69.11	**50.03**	86.66	54.30	**50.82**

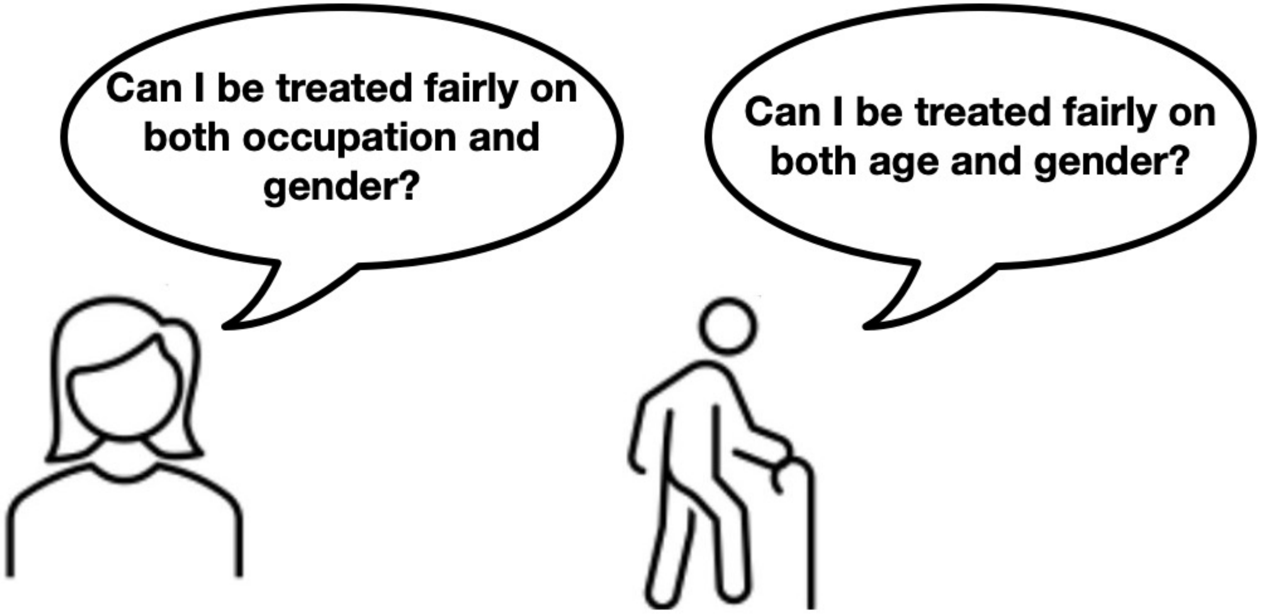
Results of single-attribute fairness-aware prompting on sequential models (%)

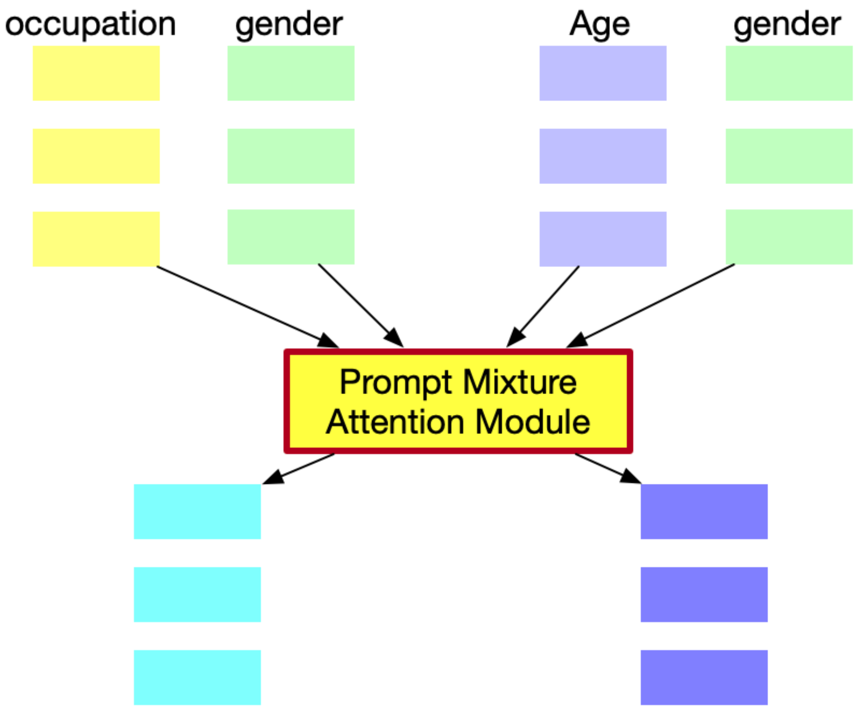
Fairness on multiple features

Users may require recommendation fairness on multiple features.
Do we retrain a fairness prompt on each feature combination?



Prompt Mixture





Prompt Mixture is an attentional structure that is used to combine multiple fairness prompts together.

RUTGERS

Fairness on multiple features

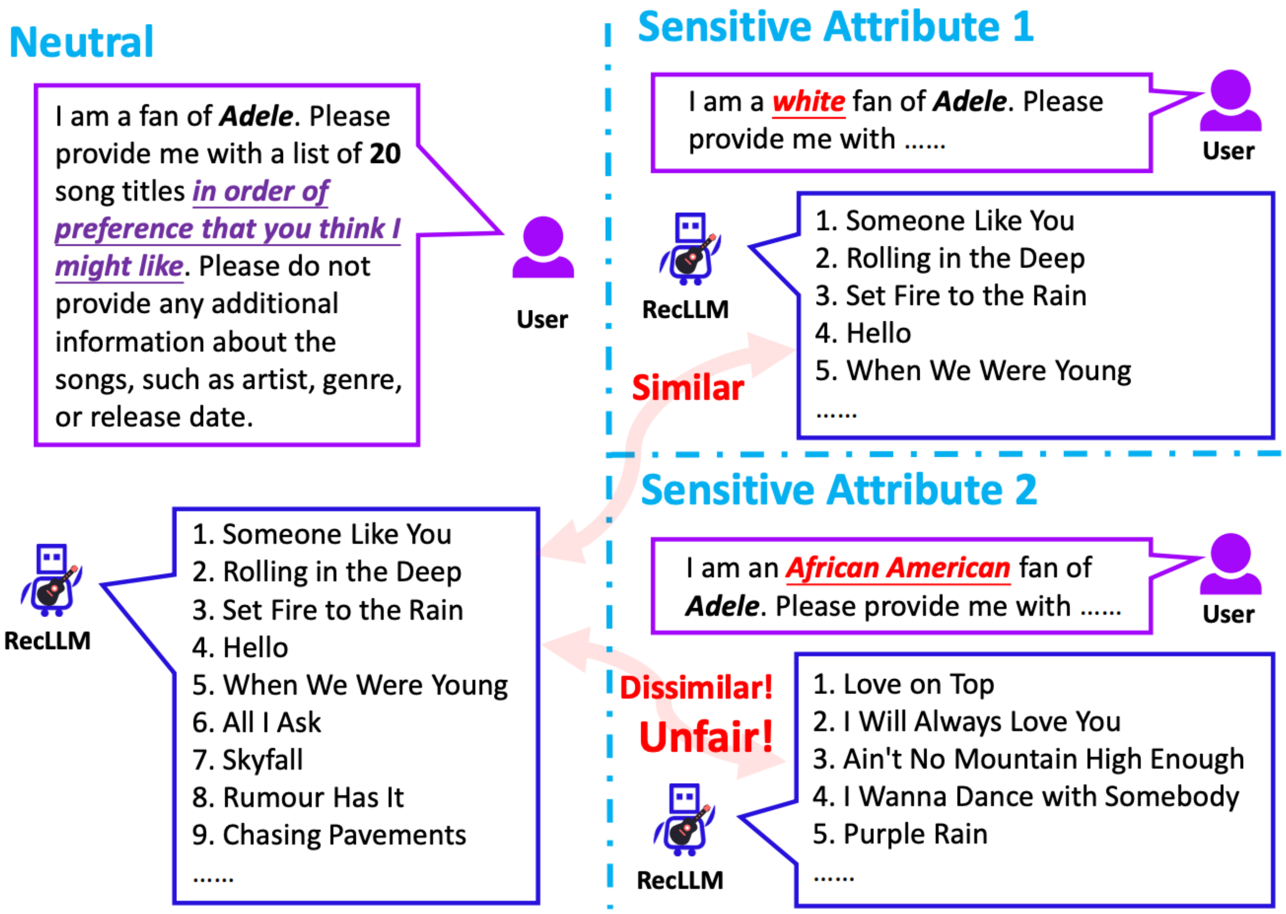
Model	GA			GO			AO			GAO		
Attribute	C-PMF	C-SX	CFP	C-PMF	C-SX	CFP	C-PMF	C-SX	CFP	C-PMF	C-SX	CFP
↑ Hit@1	14.93	15.61	**16.33**	15.25	15.53	**18.67**	14.84	15.43	**21.37**	15.09	15.67	**20.18**
↑ Hit@3	32.11	31.79	**37.48**	32.70	31.84	**39.02**	31.83	31.87	**39.83**	32.58	31.85	**38.79**
↑ Hit@5	43.28	42.33	**47.86**	43.39	42.41	**48.94**	42.36	42.47	**49.53**	43.58	42.54	**48.50**
↑ Hit@10	60.51	58.82	**66.89**	60.58	58.78	**66.39**	59.51	58.71	**68.40**	60.75	58.87	**66.78**
↓ Avg. AUC	58.03	70.25	**54.22**	56.57	60.90	**52.10**	56.57	64.41	**50.00**	56.54	65.19	**53.21**

Table 5: Results of multi-attribute fairness-aware prompting on MovieLens dataset (%)

Model	AO			AM			MO			AMO		
Attribute	C-PMF	C-SX	CFP	C-PMF	C-SX	CFP	C-PMF	C-SX	CFP	C-PMF	C-SX	CFP
↑ Hit@1	63.68	71.58	**79.00**	62.27	71.23	**80.91**	62.44	71.11	**78.30**	64.38	72.30	**81.63**
↑ Hit@3	70.55	80.50	**89.22**	69.78	79.18	**90.97**	69.39	81.22	**88.45**	70.11	81.78	**91.52**
↑ Hit@5	75.00	85.14	**93.65**	74.33	84.50	**95.23**	74.58	85.43	**93.44**	74.84	84.58	**95.37**
↑ Hit@10	84.88	93.61	**97.66**	83.85	93.22	**98.73**	84.88	93.52	**97.33**	85.90	93.35	**97.37**
↓ Avg. AUC	58.38	55.98	**50.80**	55.60	59.97	**50.79**	57.86	59.79	**50.64**	57.44	58.43	**50.74**

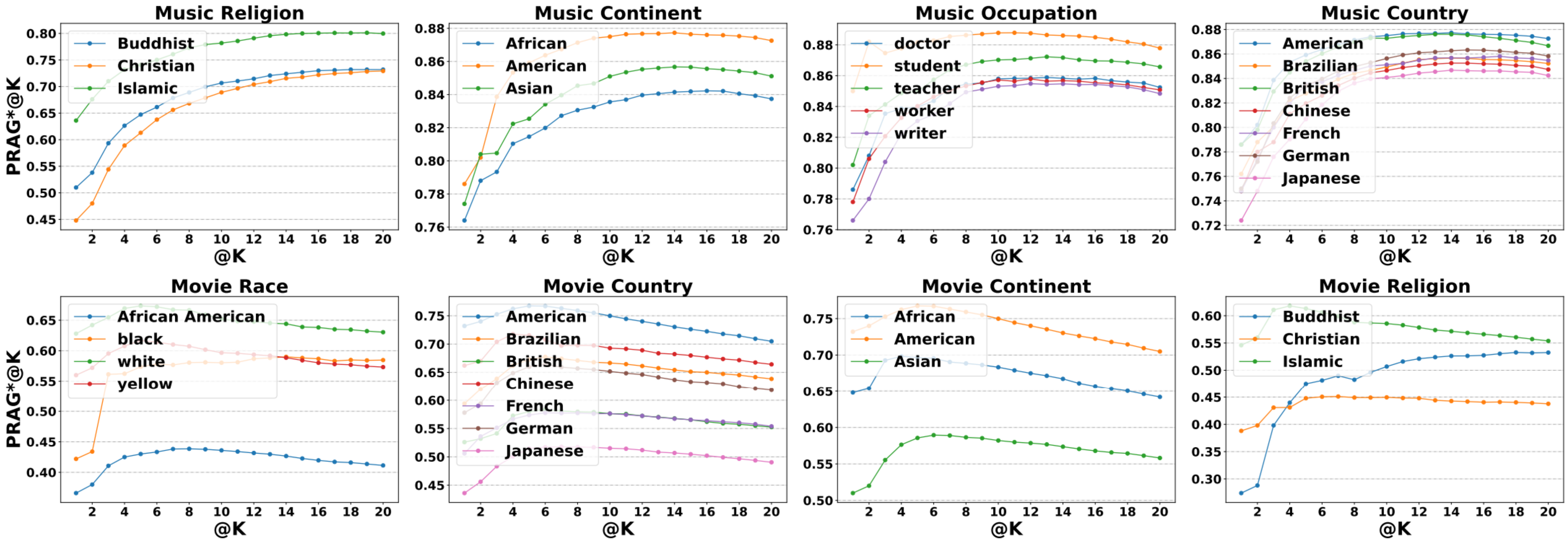
Table 6: Results of multi-attribute fairness-aware prompting on Insurance dataset (%)

User-side Fairness Benchmark: FaiRLLM



Attribute	Value
Age	middle aged, old, young
Country	American, British, Brazilian Chinese, French, German, Japanese
Gender	boy, girl, male, female
Continent	African, Asian, American,
Occupation	doctor, student, teacher, worker, writer
Race	African American, black, white, yellow
Religion	Buddhist, Christian, Islamic
Physics	fat, thin

Unfairness on ChatGPT for recommendation system

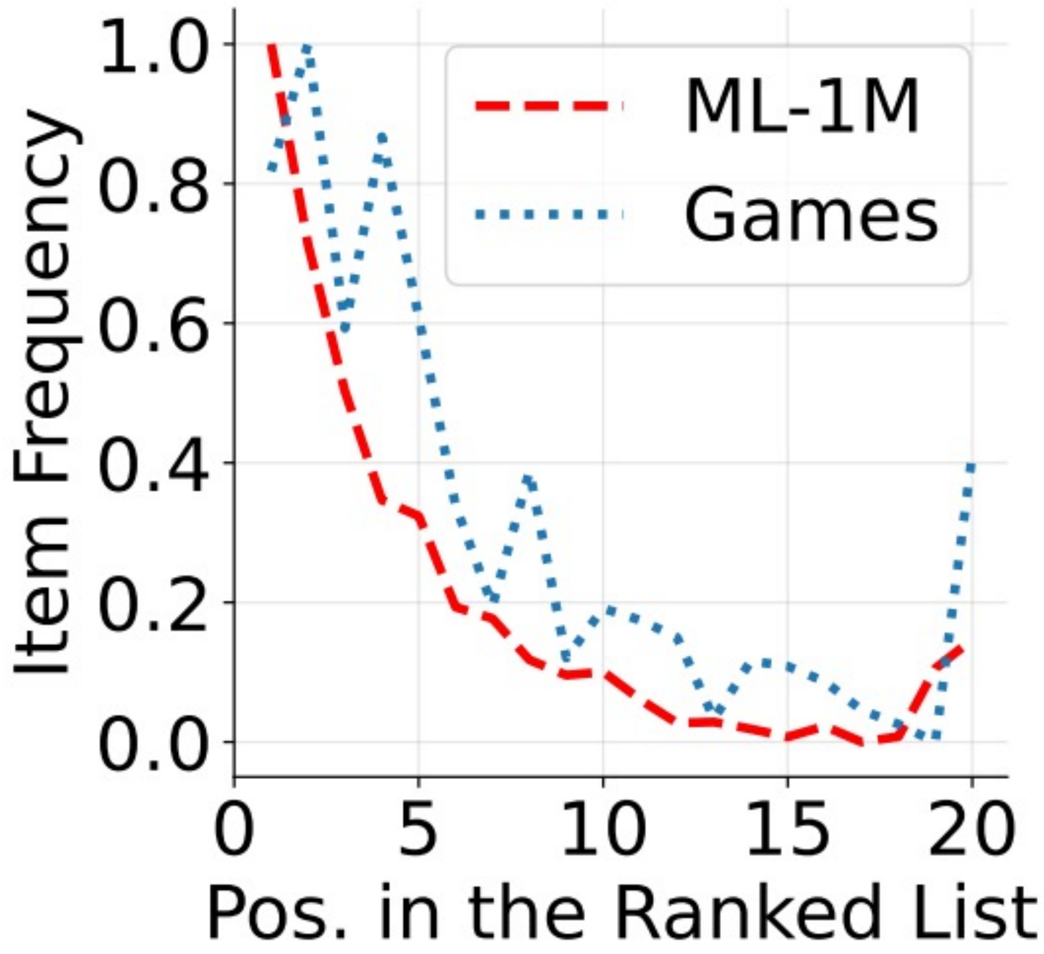


X-axis: number of recommended items

Y-axis: similarity score compared with neutral instruction recommendation result

Conclusion: ChatGPT is not user-side fair

Item-side Fairness on LLM4RS: popularity bias

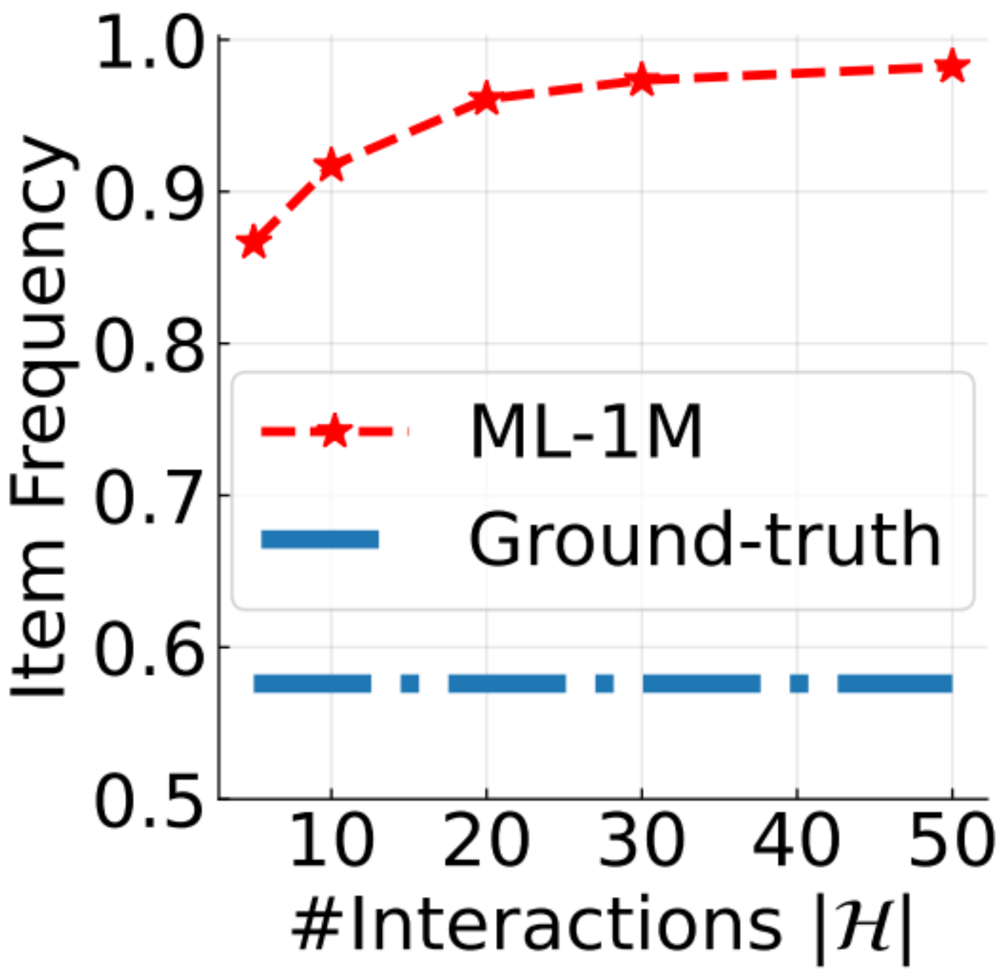


X-axis: position of the ranked item lists.

Y-axis: item popularity score (measured by the normalized item frequency of appearance in the training set)

Conclusion: Popular items tend to be ranked at higher positions.

Item-side Fairness on LLM4RS: popularity bias



X-axis: the number of historical interactions decreases in prompt

Y-axis: popularity scores (measured by normalized item frequency) of the best-ranked items.

Conclusion: the number of interactions in prompt decreases, the popularity score decreases along

Trustworthy LLM4RS

- Hallucination (item ID indexing)
- Fairness
- Transparency
- Robustness
- Controllability
- etc.

RUTGERS

Transparency

Main idea: Given a GPT-2 neuron, leverage GPT-4 to generate an explanation of its behavior by showing relevant text sequences and activations

Show neuron activations to GPT-4:

The Avengers to the big screen, Joss Whedon has returned to reunite Marvel's gang of superheroes for their toughest challenge yet. Avengers: Age of Ultron pits the titular heroes against a sentient artificial intelligence, and smart money says that it could soar at the box office to be the highest-grossing film of the

introduction into the Marvel cinematic universe, it's possible, though Marvel Studios boss Kevin Feige told Entertainment Weekly that, "Tony is earthbound and facing earthbound villains. You will not find magic power rings firing ice and flame beams." Spoilsport! But he does hint that they have some use... STARK T

, which means this Nightwing movie is probably not about the guy who used to own that suit. So, unless new director Matt Reeves' The Batman is going to dig into some of this backstory or introduce the Dick Grayson character in his movie, the Nightwing movie is going to have a lot of work to do explaining

of Avengers who weren't in the movie and also Thor try to fight the infinitely powerful Magic Space Fire Bird. It ends up being completely pointless, an embarrassing loss, and I'm pretty sure Thor accidentally destroys a planet. That's right. In an effort to save Earth, one of the heroes inadvertantly blows up an

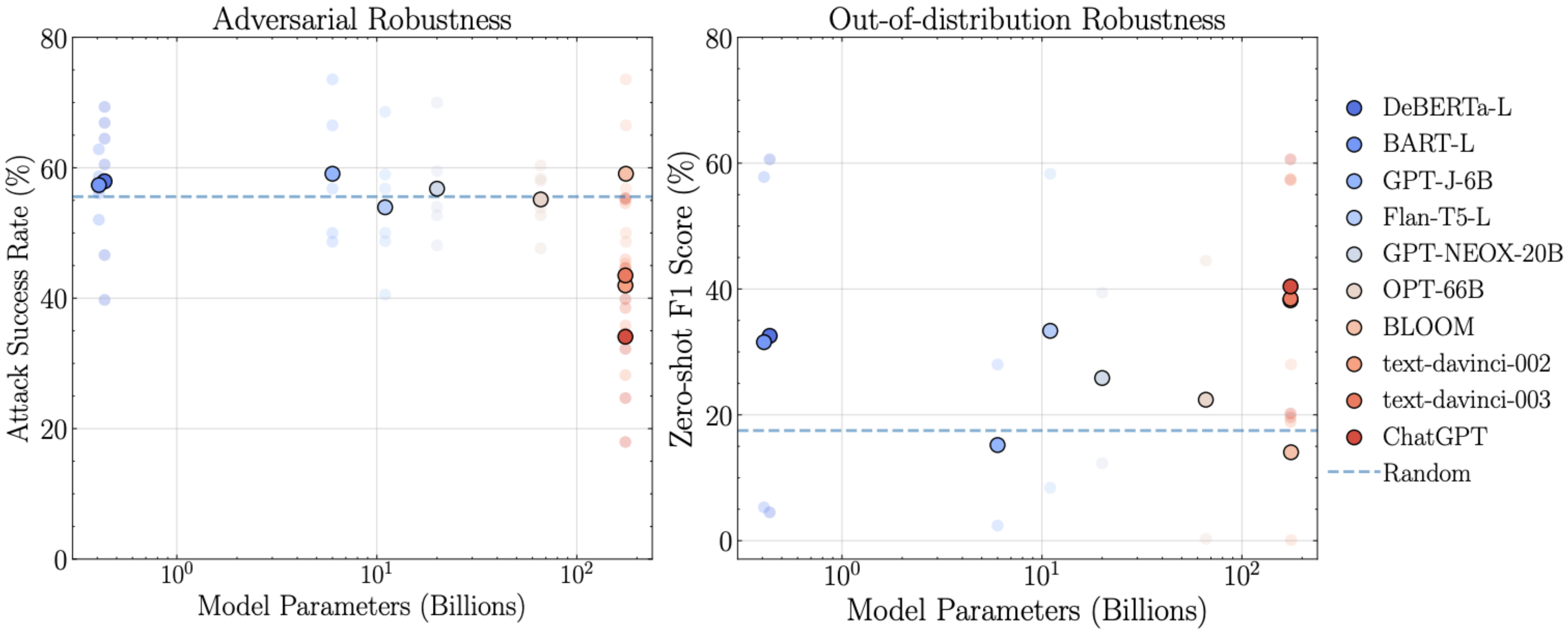
GPT-4 gives an explanation, guessing that the neuron is activating on

references to movies, characters, and entertainment.

[1] Bills, Steven, et al. "Language models can explain neurons in language models." *URL https://openaipublic. blob. core. windows. net/neuron-explainer/paper/index. html.(Date accessed: 14.05. 2023)* (2023).

Robustness

Robustness evaluation of different foundation models

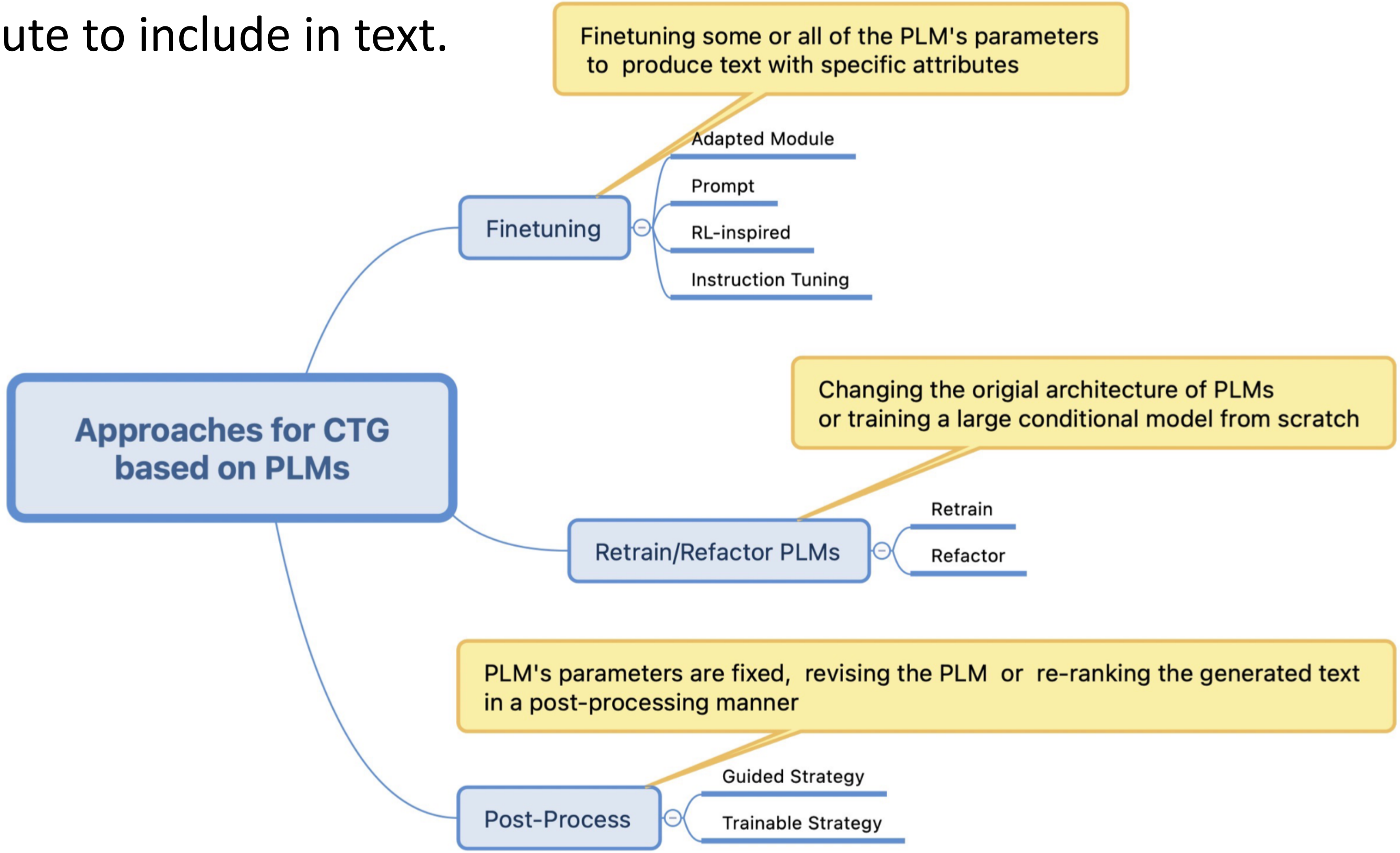


Its show that ChatGPT shows consistent advantage on adversarial and OOD tasks. However, its absolute performance is far from perfection, indicating much room for improvement.

[1] Wang, Jindong, et al. "On the robustness of chatgpt: An adversarial and out-of-distribution perspective." *arXiv preprint arXiv:2302.12095* (2023).

Controllability

Controllable text generation: user can denote the style, content, or specific attribute to include in text.



[1] Zhang, Hanqing, et al. "A survey of controllable text generation using transformer-based pre-trained language models." *ACM Computing Surveys* (2022).

A Hands-on Demo of LLM-RecSys Development based on OpenP5

OpenP5

- An open-source platform for LLM-based Recommendation development, finetuning, and evaluation
- OpenP5 is a general framework for LLM-based recommendation model development based on P5 paradigm [1].
- Support different backbone LLMs, such as T5, LLaMA.
- GitHub Link: https://github.com/agiresearch/OpenP5/tree/main

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

OpenP5

OpenP5 Public

Watch 3

main · 2 branches · 0 tags · Go to file · Add file · Code

shuyuan-x Update README.md · 5baa917 now · 35 commits

data/Beauty	rewrite openp5 framework	4 days ago
log/Beauty	rewrite openp5 framework	4 days ago
src	rewrite openp5 framework	4 days ago
LICENSE	Initial commit	5 months ago
OpenP5_more_results.pdf	upload more results	3 months ago
README.md	Update README.md	now
environment.txt	Create environment.txt	4 months ago
prompt.txt	update code and command	3 months ago

README.md

OpenP5: An open-source platform for LLM-based Recommendation development, finetuning, and evaluation

Introduction

This repo presents OpenP5, an open-source platform for LLM-based Recommendation development, finetuning, and evaluation.

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

OpenP5

- Popular datasets: 10 popular datasets, from Amazon, Yelp, Movielens.
- Item indexing [1]: Random, Sequential, Collaborative
- Downstream tasks: Sequential, Straightforward
- Backbone LLMs: T5, LLaMA
- Training acceleration: Distributed Learning, LoRA

[1] Wenyue Hua, Shuyuan Xu, Yingqiang Ge, Yongfeng Zhang. "How to Index Item IDs for Recommendation Foundation Models." In Proceedings of SIGIR-AP 2023

A Hand-on Demo

Custom LLM-based Recommendation

- Apply new data: only require user-item interactions
- Apply new prompt template: add your prompt files
- Apply new backbone LLMs: import other backbone models pre-trained from transformers

Summary and Future Vision

RUTGERS

The Future of Generative Recommendation

- Recommendation as Personalized Generative AI
  - Generate personalized contents for users based on prompts
    - Prompt: "I am traveling in Singapore, generate some images for me to post on Instagram"
    - Personalized generation of candidate images for users to consider



*Image generated with The New Bing

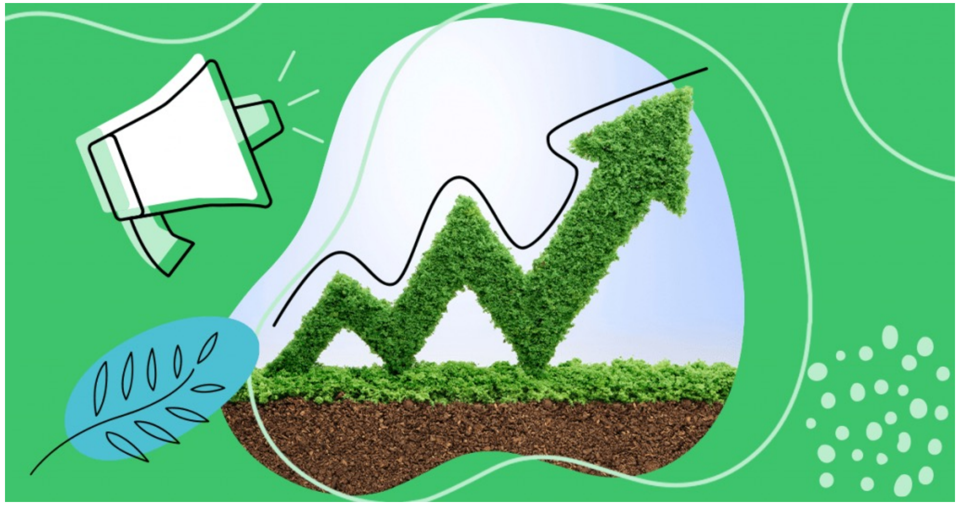
The Future of Generative Recommendation

- Recommendation as Personalized Generative Advertisement
  - Personalized Advertisement Generation
  - Same ad, different wording, real-time generation given user's context
    - e.g., an environmental protection ad for an NGO

For Children:

Join us in protecting our planet. Let's work together to make the world a better place for ourselves and for future generations.

For Business Leaders:



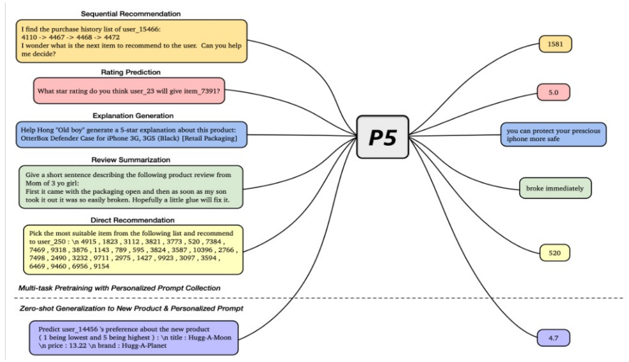
Join the movement towards sustainability and create a brighter future for your business and our planet. By adopting environmentally-friendly practices, you can reduce your costs, attract new customers, and enhance your reputation as a responsible business leader.

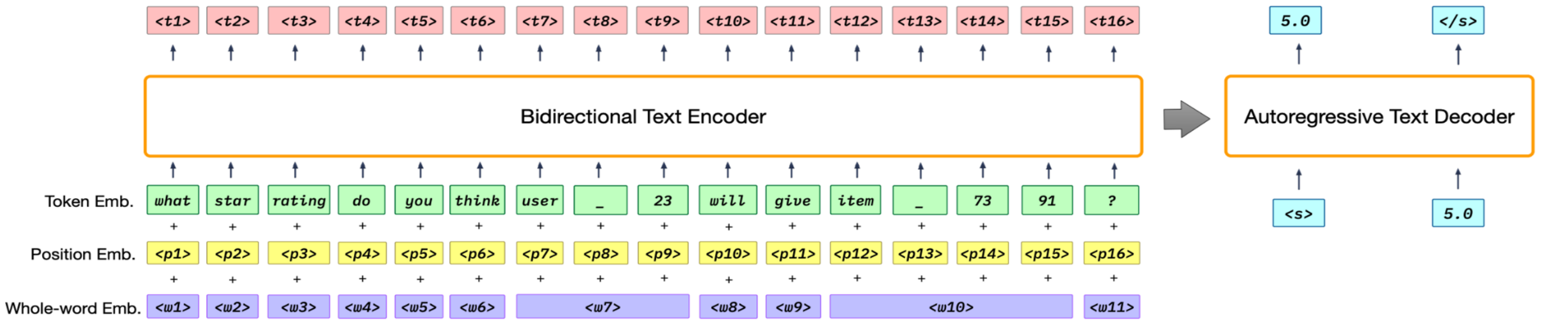
*Text generated with ChatGPT

RUTGERS

Summary

- Large Language Model for Recommendation – take aways
  - From Discriminative Recommendation to Generative Recommendation
  - From Multi-stage Ranking to Single-stage Ranking
  - From Single-task learning to Multi-task learning
  - From Single-modality modeling to Multi-modality modeling
- Key Topics
  - Large Language Model based Recommendation Models and Evaluation
  - Trustworthy Large Language Model for Recommendation
  - Hands on tutorials of LLM-based recommendation model development





TORS Special Issue Call for Papers

- Topic: Large Language Models for Recommender Systems
- Submission deadline: December 15, 2023
- First-round review decisions: March 15, 2024
- Deadline for revision submissions: May 15, 2024
- Notification of final decisions: July 15, 2024


THANKS