CSCE 670 - Information Storage and Retrieval

Lecture 24: Large Language Models for Recommendation

Yu Zhang

yuzhang@tamu.edu

November 18, 2025

Course Website: https://yuzhang-teaching.github.io/CSCE670-F25.html

Recap: How do LLMs help search engines?

Please write a question based on this passage.
Passage: {{passage}}
Query:

{{query}}

Pointwise

Passage: {{passage}}
Query: {{query}}
Does the passage answer the query?

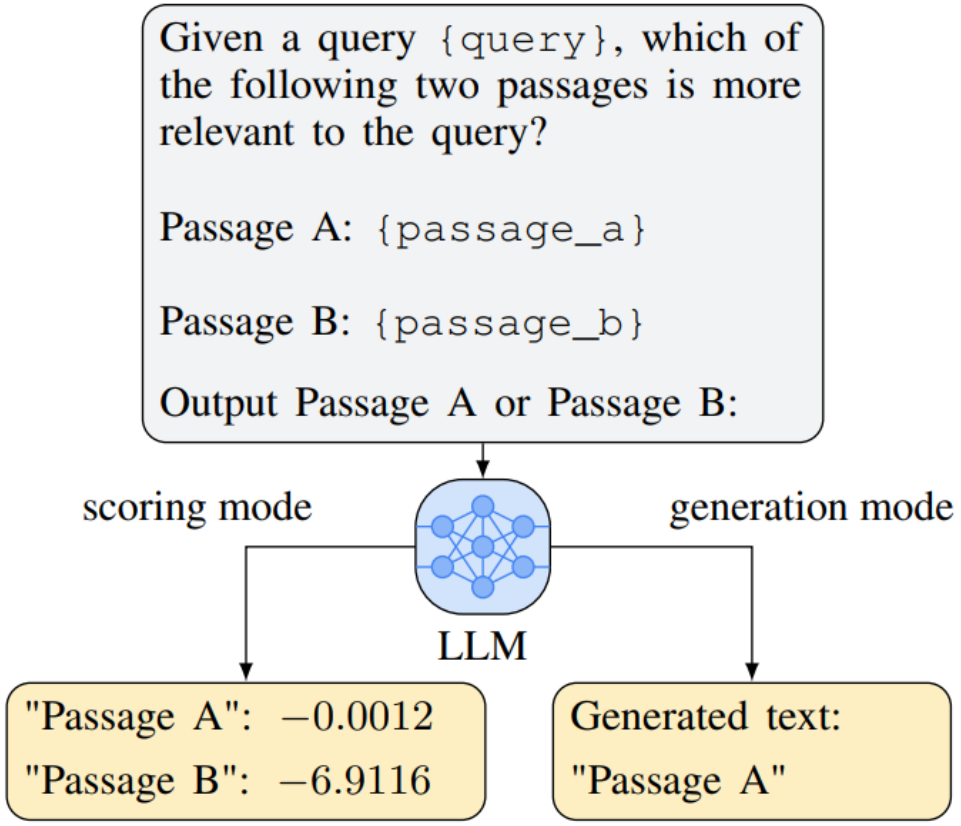
Yes (or No)

Pointwise

The following are passages related to query {{query}}
[1] {{passage_1}}
[2] {{passage_2}}
(more passages)
Rank these passages based on their relevance to the query.

[2] > [3] > [1] > [...]

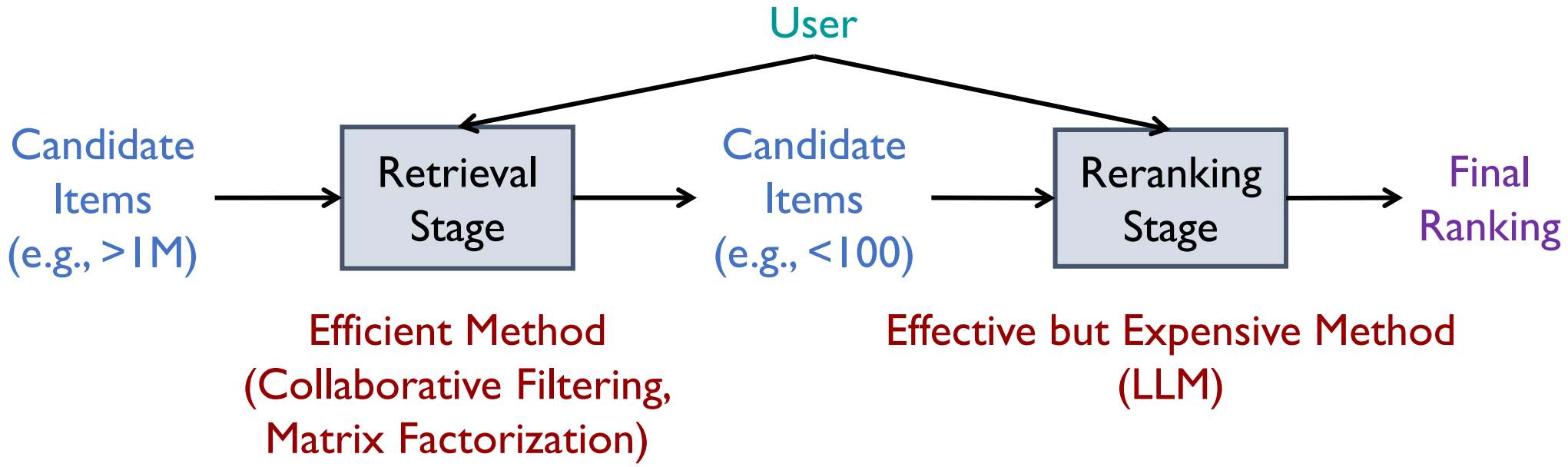
Listwise



Pairwise

Today: How do LLMs help recommender systems?

- The retrieval-reranking paradigm (again!)



- What is the difference between using LLMs for reranking in search vs. in recommendation?

Liu et al. [arXiv 2023]

Is ChatGPT a Good Recommender? A Preliminary Study

Junling Liu*
william.liuj@gmail.com
Alibaba Group
China

Chao Liu*
chize.lc@antgroup.com
Ant Group
China

Peilin Zhou*
zhoupalin@gmail.com
Hong Kong University of Science and Technology(Guangzhou)
China

Renjie Lv
lvrenjie.lrj@antgroup.com
Ant Group
China

Kang Zhou
kangbeyond89@163.com
Alibaba Group
China

Yan Zhang
yanbest0117@163.com
Alibaba Group
China

ABSTRACT

Recommendation systems have witnessed significant advancements and have been widely used over the past decades. However, most traditional recommendation methods are task-specific and therefore lack efficient generalization ability. Recently, the emergence of ChatGPT has significantly advanced NLP tasks by enhancing the capabilities of conversational models. Nonetheless, the application of ChatGPT in the recommendation domain has not been thoroughly investigated. In this paper, we employ ChatGPT as a general-purpose recommendation model to explore its potential for transferring extensive linguistic and world knowledge acquired from large-scale corpora to recommendation scenarios. Specifically, we design a set of prompts and evaluate ChatGPT's performance

KEYWORDS

Large-Language Model, ChatGPT, Recommendation System

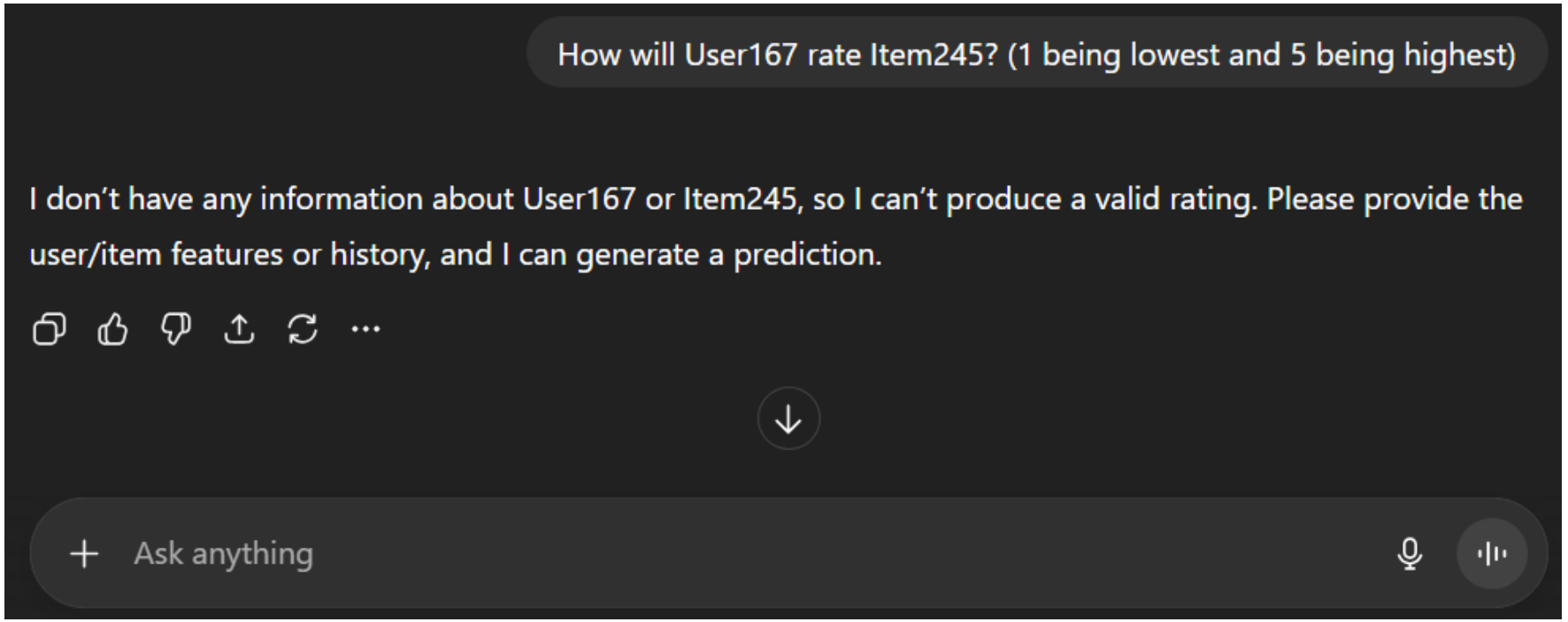
ACM Reference Format:
Junling Liu, Chao Liu, Peilin Zhou, Renjie Lv, Kang Zhou, and Yan Zhang. 2023. Is ChatGPT a Good Recommender? A Preliminary Study. In *The 1st workshop on recommendation with generative models, October 21–25, 2023, Birmingham, United Kingdom.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

As a crucial technique for addressing information overload and enhancing user experience, recommendation systems have witnessed significant advancements over the past decade and have

A Naïve Idea

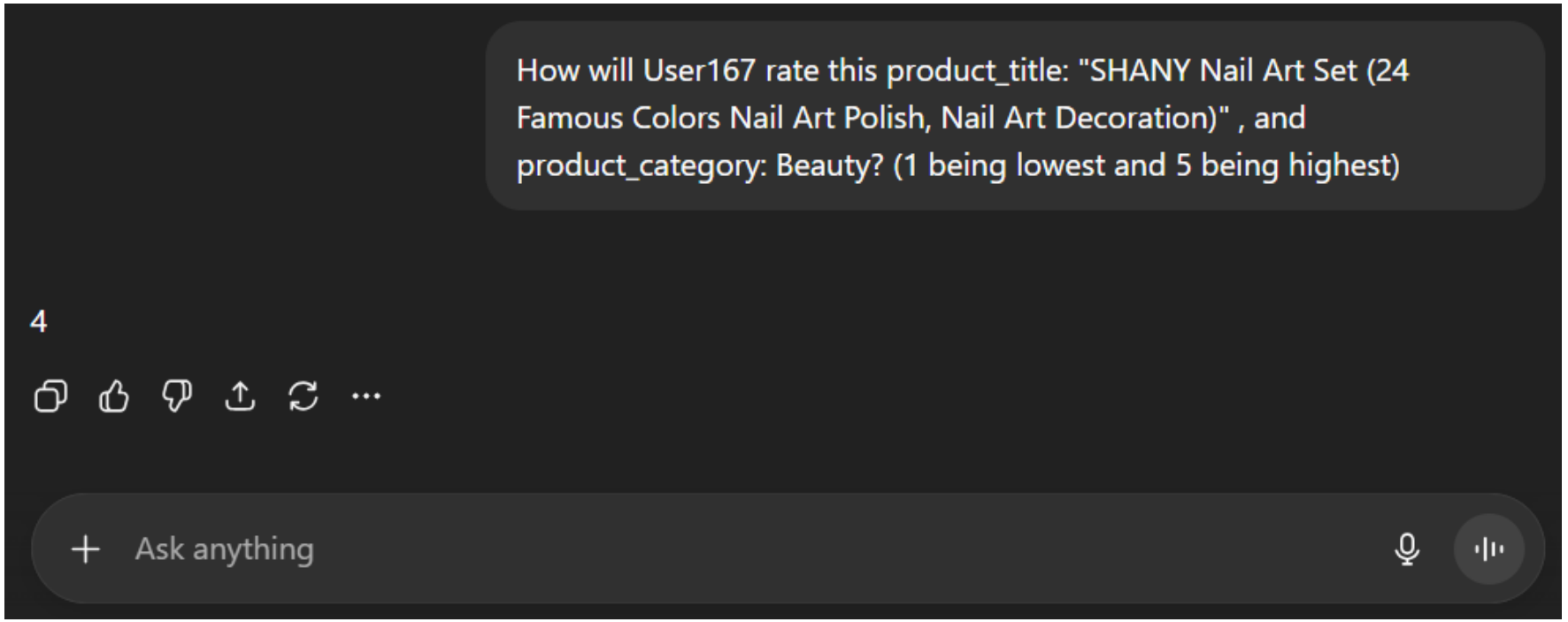
- Pointwise: Feed the user and each candidate item to the LLM, and let it output the predicted rating



We need to describe the item!

A Naïve Idea (Version 2)

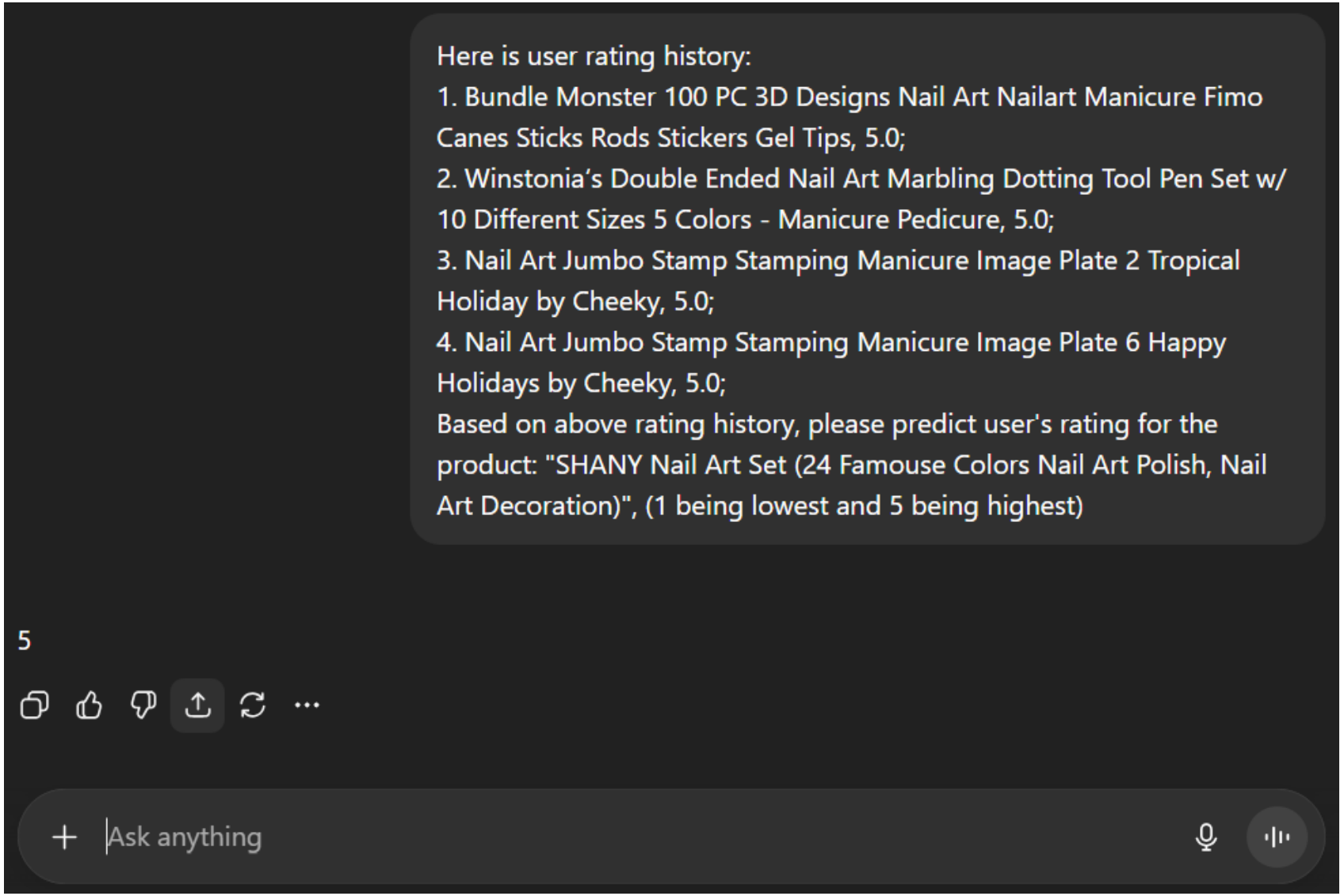
- Pointwise: Feed the user and each candidate item (described by its content) to the LLM, and let it output the predicted rating



Method	RMSE
MF	1.1973
MLP	1.3078
gpt-3.5-turbo (w/ this idea)	1.4059

We need to describe the user!

Incorporating User's Rating History



Method	RMSE
MF	1.1973
MLP	1.3078
gpt-3.5-turbo (w/ previous idea)	1.4059
gpt-3.5-turbo (w/ this idea)	1.0751

Incorporating User's Rating History

- How can we understand this solution?
  - "Zero-shot" → "Few-shot"
  - Content-based approach: We are having the LLM infer the user's profile from items the user has already rated, BUT this inference process is not based on a formula we explicitly wrote in advance

Rating Prediction

zero-shot	How will user rate this product_title: "SHANY Nail Art Set (24 Famous Colors Nail Art Polish, Nail Art Decoration)" , and product_category: Beauty? (1 being lowest and 5 being highest) Attention! Just give me back the exact number a result , and you don't need a lot of text.
few-shot	Here is user rating history: 1. Bundle Monster 100 PC 3D Designs Nail Art Nailart Manicure Fimo Canes Sticks Rods Stickers Gel Tips, 5.0; 2. Winstonia's Double Ended Nail Art Marbling Dotting Tool Pen Set w/ 10 Different Sizes 5 Colors - Manicure Pedicure, 5.0; 3. Nail Art Jumbo Stamp Stamping Manicure Image Plate 2 Tropical Holiday by Cheeky&reg, 5.0 ; 4.Nail Art Jumbo Stamp Stamping Manicure Image Plate 6 Happy Holidays by Cheeky&reg, 5.0; Based on above rating history, please predict user's rating for the product: "SHANY Nail Art Set (24 Famouse Colors Nail Art Polish, Nail Art Decoration)", (1 being lowest and5 being highest,The output should be like: (x stars, xx%), do not explain the reason.)

Incorporating User's Rating History

- If the user has already rated 1,000 items, and it is not possible to feed all of them into the LLM, what can be done?
  - First select the 10 items that are most similar to the candidate item (e.g., using the BERT embedding of the product title)
  - Collaborative Filtering: We are having the LLM aggregate the (user, candidate item) rating from the (user, similar item) ratings, BUT this aggregation process is not based on a formula we explicitly wrote in advance

Here is user rating history:
1. Bundle Monster 100 PC 3D Designs Nail Art Nailart Manicure Fimo Canes Sticks Rods Stickers Gel Tips, 5.0;
2. Winstonia's Double Ended Nail Art Marbling Dotting Tool Pen Set w/ 10 Different Sizes 5 Colors - Manicure Pedicure, 5.0;
3. Nail Art Jumbo Stamp Stamping Manicure Image Plate 2 Tropical Holiday by Cheeky&reg, 5.0 ;
4.Nail Art Jumbo Stamp Stamping Manicure Image Plate 6 Happy Holidays by Cheeky&reg, 5.0;
Based on above rating history, please predict user's rating for the product: "SHANY Nail Art Set (24 Famouse Colors Nail Art Polish, Nail Art Decoration)", (1 being lowest and5 being highest,The output should be like: (x stars, xx%), do not explain the reason.)

What if we only have implicit feedback?

- Pointwise supervision is no longer available
- Listwise: Feed the user (described by the items the user has interacted with) and all candidate items (described by their content) to the LLM, and let it output the ranking

Requirements: you must choose 10 items for recommendation and sort them in order of priority, from highest to lowest. Output format: a python list. Do not explain the reason or include any other words.
The user has interacted with the following items (in no particular order): [""Skin Obsession Jessner's Chemical Peel Kit Anti-aging and Anti-acne Skin Care Treatment"", 'Xtreme Brite Brightening Gel 1oz.',….., 'Reviva - Light Skin Peel, 1.5 oz cream']. From the candidates listed below, choose the top 10 items to recommend to the user and rank them in order of priority from highest to lowest. Candidates: ['Rogaine for Women Hair Regrowth Treatment 3- 2 ounce bottles', 'Best Age Spot Remover', ….."L'Oreal Kids Extra Gentle 2-in-1 Shampoo With a Burst of Cherry Almond, 9.0 Fluid Ounce""].

- Instruction
- User interaction history (a list of product titles, without any particular order)
- Candidate items

What if we only have implicit feedback?

- Listwise: Feed the user (described by the items the user has interacted with) and all candidate items (described by their content) to the LLM, and let it output the ranking

Requirements: you must choose 10 items for recommendation and sort them in order of priority, from highest to lowest. Output format: a python list. Do not explain the reason or include any other words.
The user has interacted with the following items (in no particular order): ['Maybelline New York Eye Studio Lasting Drama Gel Eyeliner, Eggplant 956, 0.106 Ounce', ""L'Oreal Paris Healthy Look Hair Color, 8.5 Blonde/White Chocolate"", ……, 'Duo Lash Adhesive, Clear, 0.25 Ounce']. Given that the user has interacted with 'WAWO 15 Color Professionl Makeup Eyeshadow Camouflage Facial Concealer Neutral Palette' from a pool of candidates: ['MASH Bamboo Reusable Cuticle Pushers Remover / Manicure Pedicure Stick', 'Urban Decay All Nighter Long-Lasting Makeup Setting Spray 4 oz', ...,'Classic Cotton Balls Jumbo Size, 100 Count'], please recommend the best item from a new candidate pool, ['Neutrogena Ultra Sheer Sunscreen SPF 45 Twin Pack 6.0 Ounce', 'Blinc Eyeliner Pencil - Black', ...,'Skin MD Natural + SPF15 combines the benefits of a shielding lotion and a sunscreen lotion']. Note that the candidates in the new pool are not ordered in any particular way.

  - Instruction
  - User interaction history (hold out one item as a demonstration example)
  - Demonstration example (the held-out item + a dummy candidate pool (items the user has not interacted with))
  - Candidate items

What if we want to perform sequential recommendation?

- Listwise: Feed the user (described by the items the user has interacted with) and all candidate items (described by their content) to the LLM, and let it output the ranking

Requirements: you must choose 10 items for recommendation and sort them in order of priority, from highest to lowest. Output format: a python list. Do not explain the reason or include any other words.
The user has interacted with the following items in chronological order: ['Better Living Classic Two Chamber Dispenser, White', 'Andre Silhouettes Shampoo Cape, Metallic Black', ….., 'John Frieda JFHA5 Hot Air Brush, 1.5 inch'].Please recommend the next item that the user might interact with.

  - Instruction
  - User interaction history (a list of product titles, in chronological order)
  - Candidate items

What if we want to perform sequential recommendation?

- Listwise: Feed the user (described by the items the user has interacted with) and all candidate items (described by their content) to the LLM, and let it output the ranking

Requirements: you must choose 10 items for recommendation and sort them in order of priority, from highest to lowest. Output format: a python list. Do not explain the reason or include any other words.
Given the user's interaction history in chronological order: ['Avalon Biotin B-Complex Thickening Conditioner, 14 Ounce', 'Conair 1600 Watt Folding Handle Hair Dryer', ….., 'RoC Multi-Correxion 4-Zone Daily Moisturizer, SPF 30, 1.7 Ounce'], the next interacted item is ['Le Edge Full Body Exfoliator - Pink']. Now, if the interaction history is updated to ['Avalon Biotin B-Complex Thickening Conditioner, 14 Ounce', 'Conair 1600 Watt Folding Handle Hair Dryer',….., 'RoC Multi-Correxion 4-Zone Daily Moisturizer, SPF 30, 1.7 Ounce', 'Le Edge Full Body Exfoliator - Pink'] and the user is likely to interact again, recommend the next item.

- Instruction
- User interaction history (hold out the last item as a demonstration example)
- Demonstration example (the held-out item), which is also the updated interaction history
- Candidate items

Performance

- LLMs still underperform non-LLM baselines (fully supervised) by a notable margin!
  - The follow-up work we will introduce also does not overcome this limitation!

Recommendation with Implicit Feedback

Method	nDCG@5	nDCG@10
BPR-MF	0.0857	0.1224
BPR-MLP	0.0848	0.1225
gpt-3.5-turbo (w/ demonstration)	0.0216	0.0398

Sequential Recommendation

Method	nDCG@5	nDCG@10
SASRec	0.0249	0.0318
S^3-Rec	0.0244	0.0327
gpt-3.5-turbo (w/ demonstration)	0.0135	0.0135

Wang and Lim [arXiv 2023]

Zero-Shot Next-Item Recommendation using Large Pretrained Language Models

Lei Wang
Ee-Peng Lim*
lei.wang.2019@phdcs.smu.edu.sg
eplim@smu.edu.sg
Singapore Management University
Singapore

ABSTRACT

Large language models (LLMs) have achieved impressive zero-shot performance in various natural language processing (NLP) tasks, demonstrating their capabilities for inference without training examples. Despite their success, no research has yet explored the potential of LLMs to perform next-item recommendations in the zero-shot setting. We have identified two major challenges that must be addressed to enable LLMs to act effectively as recommenders. First, the recommendation space can be extremely large for LLMs, and LLMs do not know about the target user's past interacted items and preferences. To address this gap, we propose a prompting strategy called **Zero-Shot Next-Item Recommendation (NIR)** prompting that directs LLMs to make next-item recom-

ACM Reference Format:
Lei Wang and Ee-Peng Lim*. 2023. Zero-Shot Next-Item Recommendation using Large Pretrained Language Models. In *Proceedings of ACM Conference (Conference'17).* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

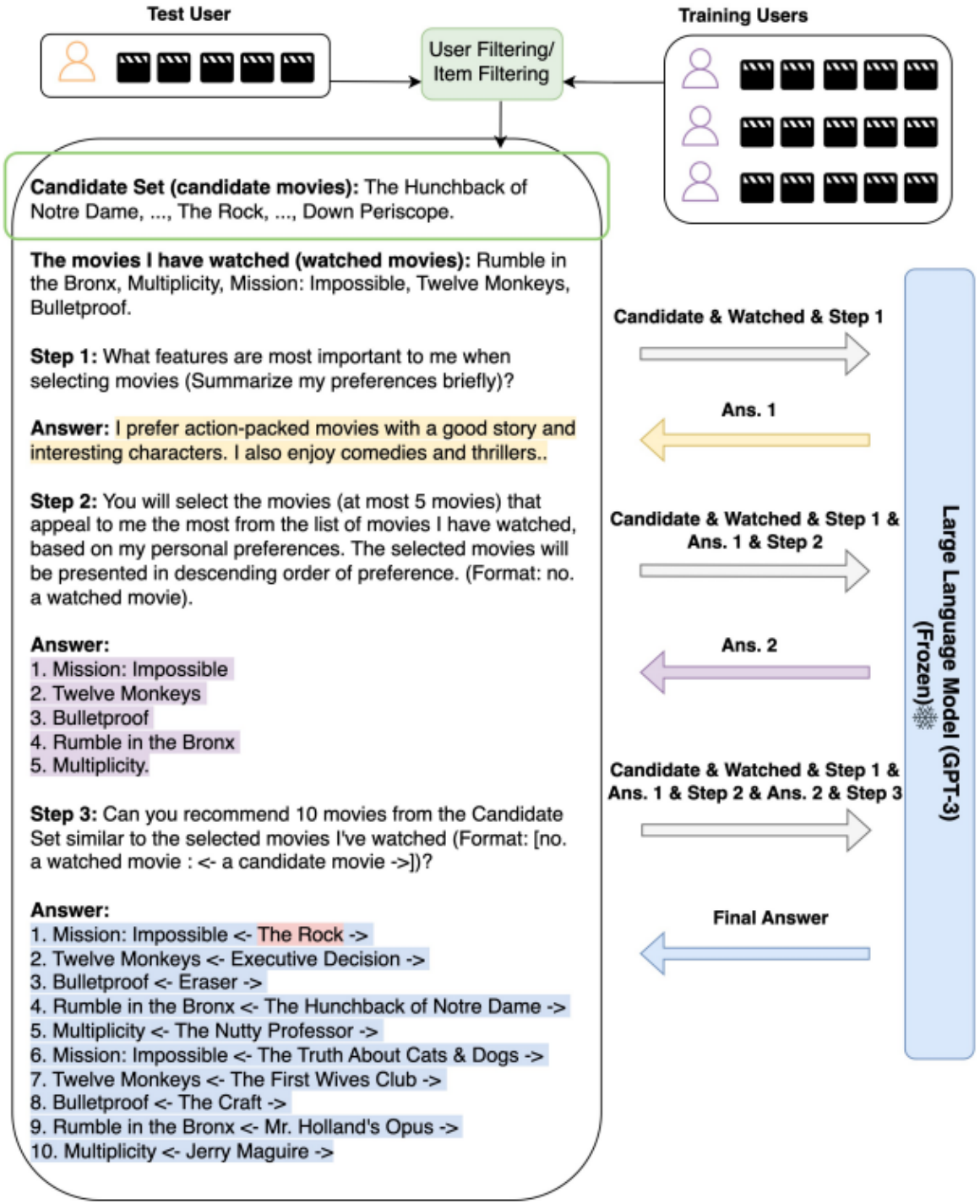
1 INTRODUCTION

Large language models (LLMs) [1, 3, 25], such as GPT-3 [1], have achieved impressive results in various natural language processing (NLP) tasks. Nevertheless, LLMs are also very large and often accessible only via some API service. Hence, they cannot be fine-tuned like the earlier pre-trained language models (PTMs) [5, 15].

Step-by-Step Prompting

- Step 1: Summarize the user's profile
- Step 2: Pick the top-k items the user likes most from their previously interacted items
- Step 3: Pick unwatched movies that are closest to the user's favorite

Method	nDCG@10
SASRec	0.0573
CL4SRec	0.0617
text-davinci-003	0.0546



Hou et al. [ECIR 2024]

Large Language Models are Zero-Shot Rankers for Recommender Systems

Yupeng Hou[1,2], Junjie Zhang[1], Zihan Lin[3], Hongyu Lu[4], Ruobing Xie[4], Julian McAuley[2], and Wayne Xin Zhao[1(✉)]

[1] Gaoling School of Artificial Intelligence, Renmin University of China, Beijing, China
yphou@ucsd.edu, junjie.zhang@ruc.edu.cn, batmanfly@gmail.com
[2] UC San Diego, San Diego, USA
[3] School of Information, Renmin University of China, Beijing, China
[4] WeChat, Tencent, Shenzhen, China

Abstract. Recently, large language models (LLMs) (*e.g.,* GPT-4) have demonstrated impressive general-purpose task-solving abilities, including the potential to approach recommendation tasks. Along this line of research, this work aims to investigate the capacity of LLMs that act as the ranking model for recommender systems. We first formalize the recommendation problem as a conditional ranking task, considering sequential interaction histories as *conditions* and the items retrieved by

Prompting Strategies

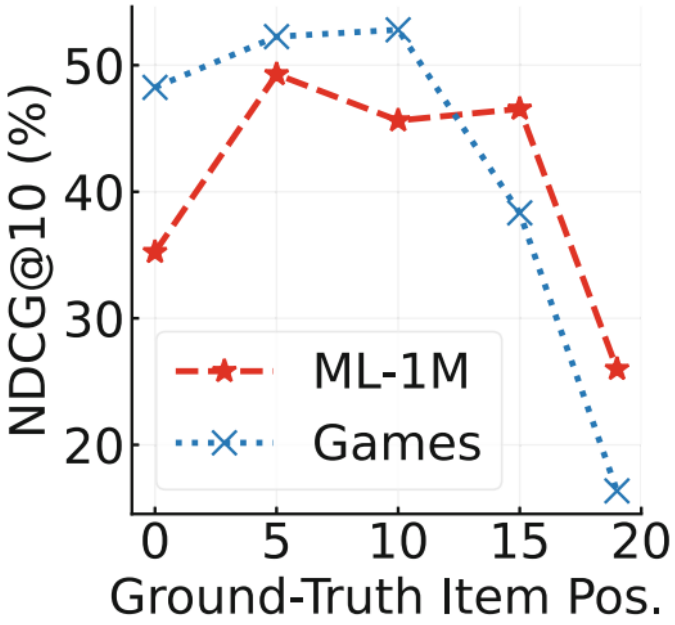
- Sequential prompting: Arrange the historical interactions in chronological order (already introduced in [Liu et al., arXiv 2023])
  - "*I've watched the following movies in the past in order: '0. Multiplicity', '1. Jurassic Park', ...*"
- Recency-focused prompting: In addition to the sequential interaction records, we can add an additional sentence to emphasize the most recent interaction
  - "*I've watched the following movies in the past in order: '0. Multiplicity', '1. Jurassic Park', ... Note that my most recently watched movie is Dead Presidents. ...*"
- In-context learning: Holding out the last item in the interaction history as a demonstration example (already introduced in [Liu et al., arXiv 2023])
  - "*If I've watched the following movies in the past in order: '0. Multiplicity', '1. Jurassic Park', ..., then you should recommend Dead Presidents to me and now that I've watched Dead Presidents, then ...*"

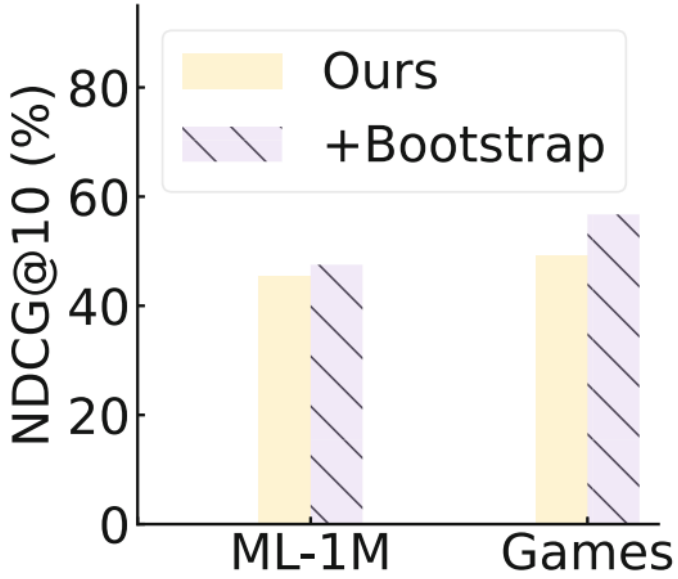
Performance

	Method	ML-1M				Games			
		N@1	N@5	N@10	N@20	N@1	N@5	N@10	N@20
full	Pop	22.91	45.16	52.33	55.36	28.35	47.42	52.96	57.45
	BPRMF [49]	34.60	59.87	64.29	65.39	44.92	62.33	66.27	68.94
	SASRec [33]	61.39	76.39	78.89	79.79	56.90	73.19	75.92	77.14
zero-shot	BM25 [50]	4.70	12.68	17.88	33.19	13.92	28.81	34.61	44.35
	UniSRec [30]	7.37	18.80	26.67	37.93	18.95	33.99	40.71	48.42
	VQ-Rec [29]	5.98	15.48	23.74	35.85	7.28	18.28	26.21	37.62
	Sequential	18.28	36.35	42.85	49.02	30.28	45.48	50.57	56.55
	Recency-Focused	19.57	37.73	44.23	50.01	**34.03**	**48.77**	**53.50**	**59.01**
	In-Context Learning	**21.77**	**39.59**	**45.83**	**51.62**	33.95	48.44	53.10	58.92

LLMs suffer from position bias

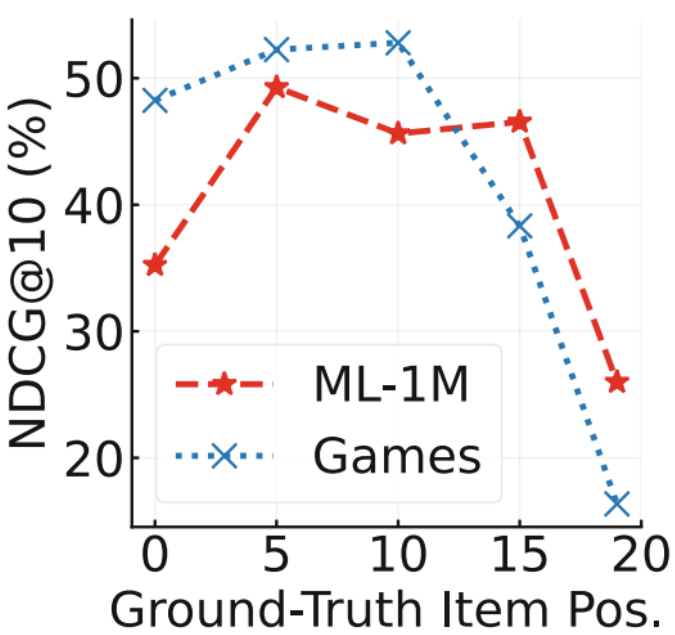
- The ranking performance drops significantly when the ground-truth items appear at the last few positions
- Although they both exhibit position bias, the finding here is somewhat different from the "lost in the middle" effect, possibly due to the different tasks (retrieval-augmented generation vs. recommendation)

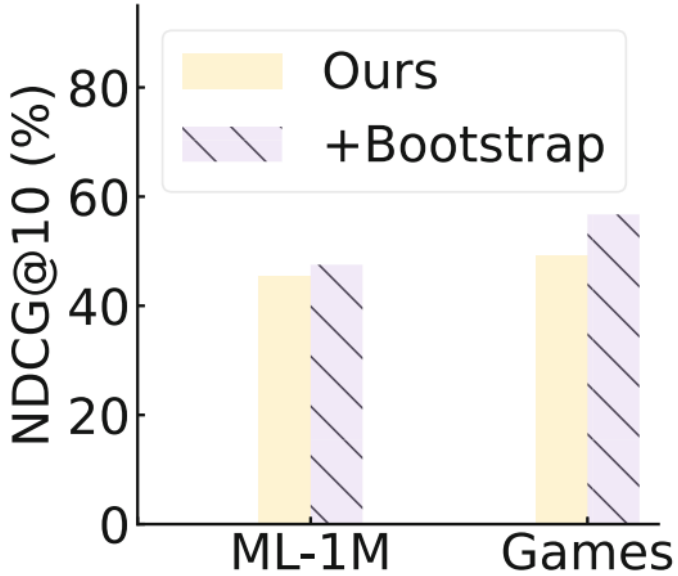




LLMs suffer from position bias

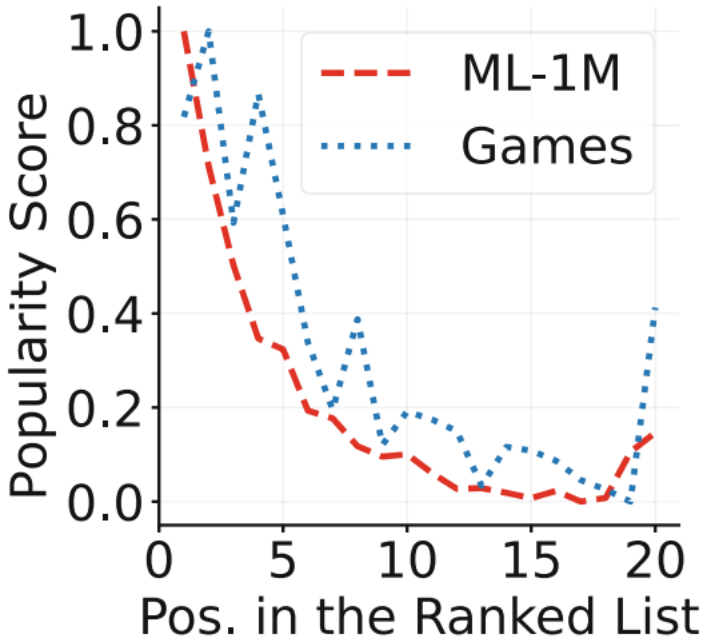
- Solution in this paper: Rank the candidate set repeatedly for multiple times, with candidates randomly shuffled at each round. Then merge the results of each round to derive the final ranking





LLMs suffer from popularity bias

- Popular items tend to be ranked at higher positions by LLMs
- Why?
  - For popular items, the associated text may also appear frequently in the pre-training corpora of LLMs
  - E.g., a best-selling book would be widely discussed on the Web



Fine-Tuning a Language Model for Recommendation

Recformer [Li et al., KDD 2023]

Text Is All You Need: Learning Language Representations for Sequential Recommendation

Jiacheng Li
University of California, San Diego
j9li@eng.ucsd.edu

Ming Wang
Amazon, United States
mingww@amazon.com

Jin Li
Amazon, United States
jincli@amazon.com

Jinmiao Fu
Amazon, United States
jinnmiaof@amazon.com

Xin Shen
Amazon, United States
xinshen@amazon.com

Jingbo Shang
University of California, San Diego
jshang@eng.ucsd.edu

Julian McAuley
University of California, San Diego
jmcauley@eng.ucsd.edu

ABSTRACT

Sequential recommendation aims to model dynamic user behavior from historical interactions. Existing methods rely on either explicit item IDs or general textual features for sequence modeling to understand user preferences. While promising, these approaches still struggle to model cold-start items or transfer knowledge to new datasets. In this paper, we propose to model user preferences and item features as language representations that can be generalized to new items and datasets. To this end, we present a novel framework, named RECFORMER, which effectively learns language representations for sequential recommendation. Specifically, we propose to formulate an item as a "sentence" (word sequence) by flattening item key-value attributes described by text so that an item sequence

ACM Reference Format:
Jiacheng Li, Ming Wang, Jin Li, Jinmiao Fu, Xin Shen, Jingbo Shang, and Julian McAuley. 2023. Text Is All You Need: Learning Language Representations for Sequential Recommendation. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '23), August 6–10, 2023, Long Beach, CA, USA.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3580305.3599519

1 INTRODUCTION

Sequential recommender systems model historical user interactions as temporally-ordered sequences to recommend potential items that users are interested in. Sequential recommenders [11, 14, 25, 27] can capture both short-term and long-term preferences of users

Item Key-Value Attribute Pairs

Item sequence

Item ID sequence

315 → 235 → 822

Key-value attribute pair sequence

Title	2020 MacBook Air Laptop M1 Chip
Brand	Apple
Color	Gold

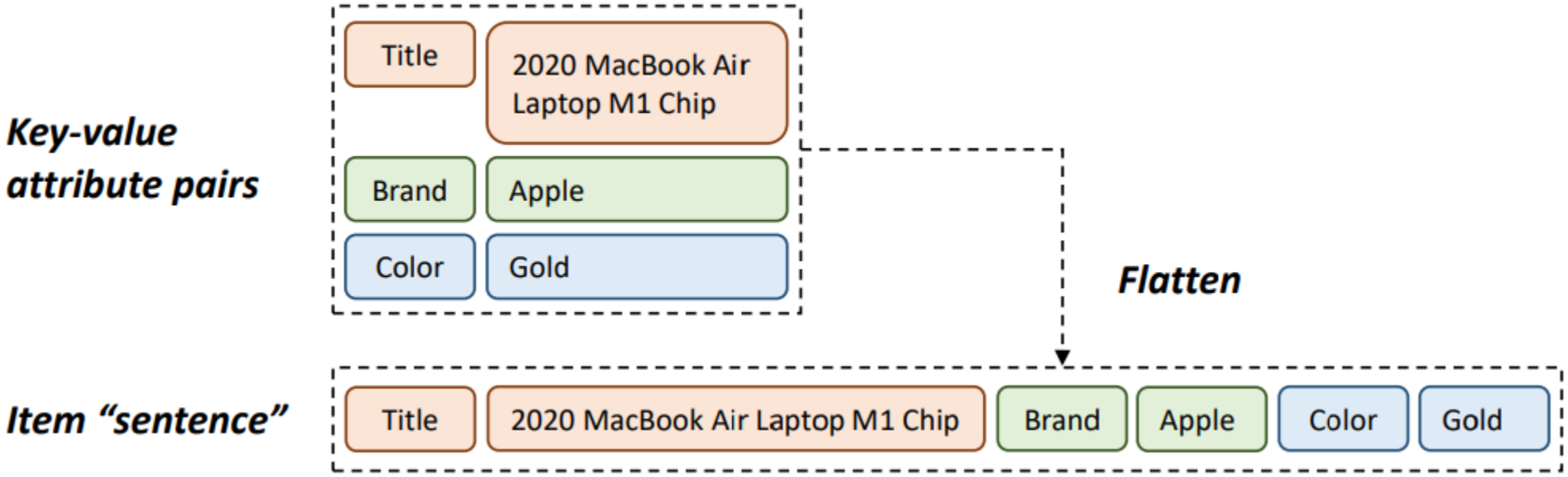
→

Title	Wireless Computer Mouse Compatible with MacBook
Brand	Amazon Basics
Color	Black

→

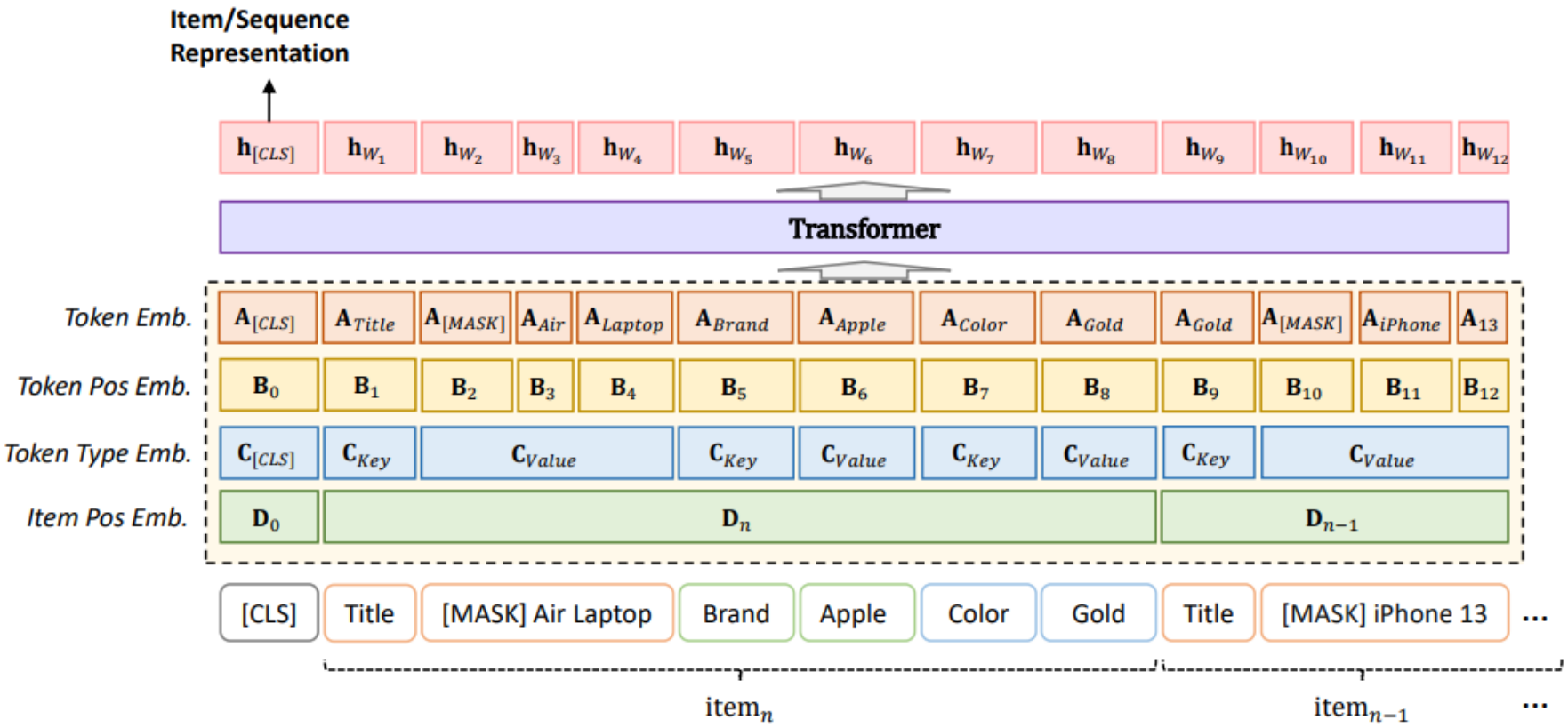
Title	Apple iPhone 13 (256GB, Pink)
Brand	Apple
Color	Pink

Flatten Key-Value Attribute Pairs into a "Sentence"



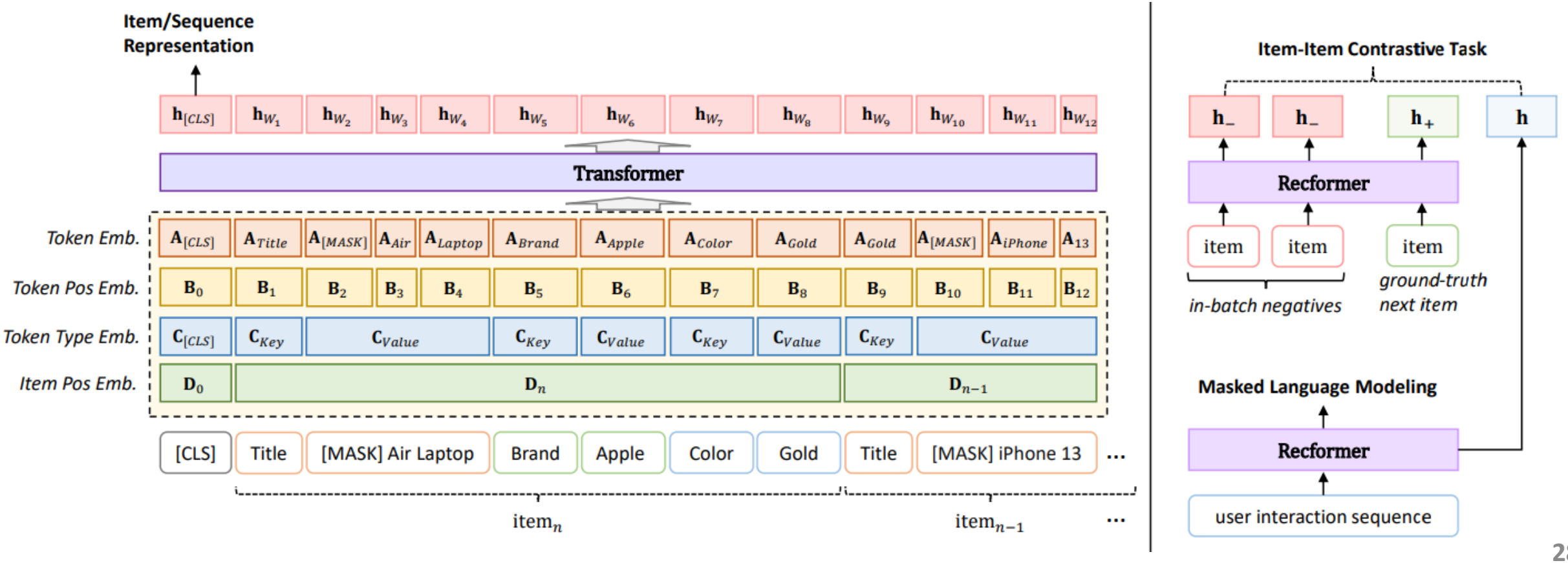
Feed "Sentences" into a Transformer Encoder

- Putting the most recent items on the left helps preserve them when the interaction history exceeds the model's maximum input length



Contrastive Learning

- Putting the most recent items on the left helps preserve them when the interaction history exceeds the model's maximum input length
- Contrastive learning helps the model better learn item representation vectors



Performance

Dataset	Metric	ID-Only Methods				ID-Text Methods		Text-Only Methods			Improv.
		GRU4Rec	SASRec	BERT4Rec	RecGURU	FDSA	S^3-Rec	ZESRec	UniSRec	RECFORMER	
Scientific	NDCG@10	0.0826	0.0797	0.0790	0.0575	0.0716	0.0451	0.0843	0.0862	**0.1027**	19.14%
	Recall@10	0.1055	0.1305	0.1061	0.0781	0.0967	0.0804	0.1260	0.1255	**0.1448**	10.96%
	MRR	0.0702	0.0696	0.0759	0.0566	0.0692	0.0392	0.0745	0.0786	**0.0951**	20.99%
Instruments	NDCG@10	0.0633	0.0634	0.0707	0.0468	0.0731	0.0797	0.0694	0.0785	**0.0830**	4.14%
	Recall@10	0.0969	0.0995	0.0972	0.0617	0.1006	0.1110	0.1078	**0.1119**	0.1052	-
	MRR	0.0707	0.0577	0.0677	0.0460	0.0748	0.0755	0.0633	0.0740	**0.0807**	6.89%
Arts	NDCG@10	0.1075	0.0848	0.0942	0.0525	0.0994	0.1026	0.0970	0.0894	**0.1252**	16.47%
	Recall@10	0.1317	0.1342	0.1236	0.0742	0.1209	0.1399	0.1349	0.1333	**0.1614**	15.37%
	MRR	0.1041	0.0742	0.0899	0.0488	0.0941	0.1057	0.0870	0.0798	**0.1189**	12.49%
Office	NDCG@10	0.0761	0.0832	0.0972	0.0500	0.0922	0.0911	0.0865	0.0919	**0.1141**	17.39%
	Recall@10	0.1053	0.1196	0.1205	0.0647	0.1285	0.1186	0.1199	0.1262	**0.1403**	9.18%
	MRR	0.0731	0.0751	0.0932	0.0483	0.0972	0.0957	0.0797	0.0848	**0.1089**	12.04%
Games	NDCG@10	0.0586	0.0547	0.0628	0.0386	0.0600	0.0532	0.0530	0.0580	**0.0684**	8.92%
	Recall@10	0.0988	0.0953	0.1029	0.0479	0.0931	0.0879	0.0844	0.0923	**0.1039**	0.97%
	MRR	0.0539	0.0505	0.0585	0.0396	0.0546	0.0500	0.0505	0.0552	**0.0650**	11.11%
Pet	NDCG@10	0.0648	0.0569	0.0602	0.0366	0.0673	0.0742	0.0754	0.0702	**0.0972**	28.91%
	Recall@10	0.0781	0.0881	0.0765	0.0415	0.0949	0.1039	0.1018	0.0933	**0.1162**	11.84%
	MRR	0.0632	0.0507	0.0585	0.0371	0.0650	0.0710	0.0706	0.0650	**0.0940**	32.39%

Next Lecture

- Logistics for the final project presentation and final exam
- Quiz 4!
  - All policies are the same as Quiz 1 (number of questions, time limit, grading, etc.)
  - Scope:
    - Lecture 19 (Neural Collaborative Filtering)
    - Lecture 20 (Sequential Recommendation)
    - Lecture 21 (Large Language Models Basics)
    - Lecture 23 (Large Language Models for Ranking)
    - Lecture 24 (Large Language Models for Recommendation)
    - Homework 3
  - Guest Lecture 22 will NOT appear in Quiz 4 or the final exam!

CSCE 670 - Information Storage and Retrieval

Information Retrieval for Science
(will not appear in quizzes or the exam)

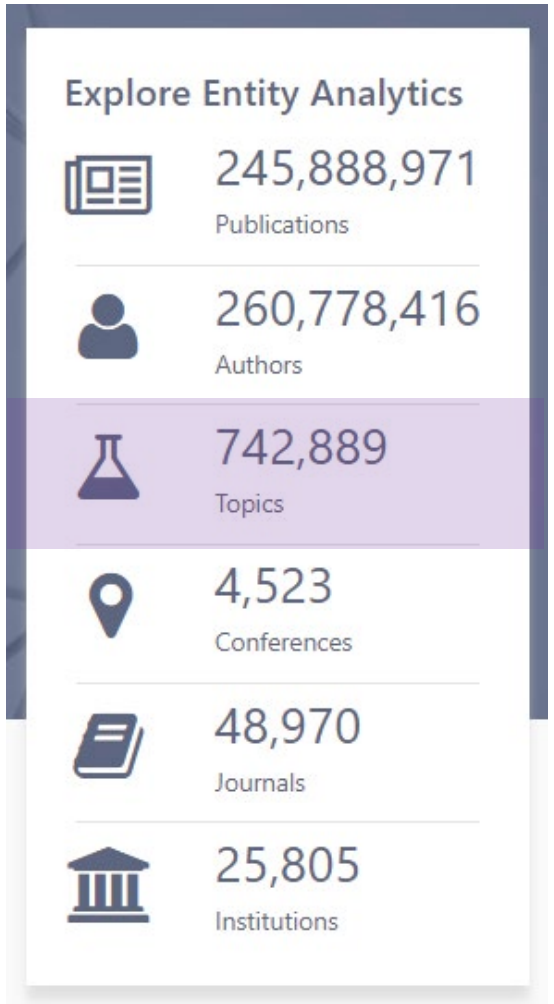
Yu Zhang

yuzhang@tamu.edu

November 18, 2025

Course Website: https://yuzhang-teaching.github.io/CSCE670-F25.html

Fine-Grained Scientific Paper Classification



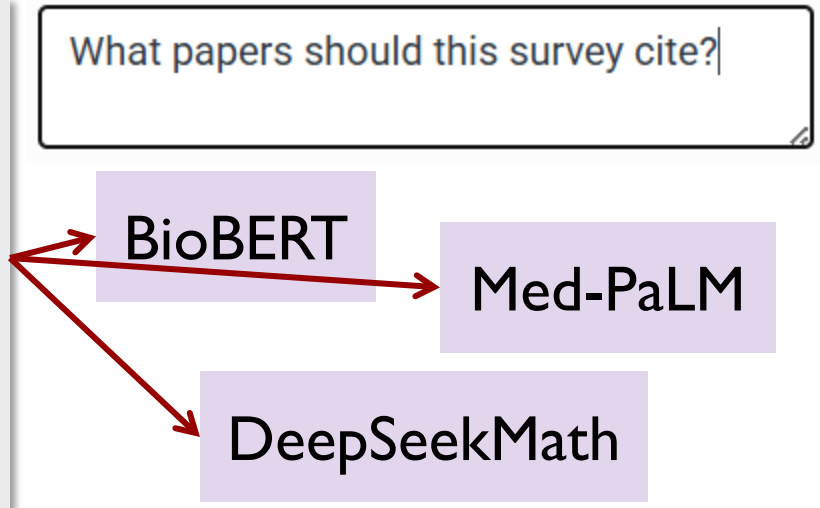
- The Microsoft Academic Graph has 740K+ categories.
- The Medical Subject Headings (MeSH) for indexing PubMed papers contain 30K+ categories.
- Each paper can be relevant to more than one category (5-15 categories for most papers).

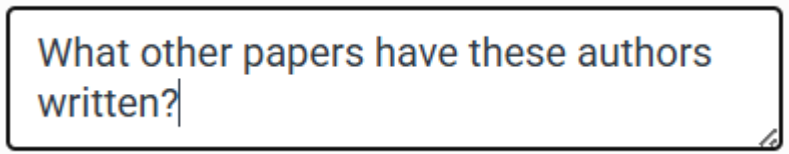
Clinical course and risk factors for mortality of adult inpatients with COVID-19 in Wuhan, China: a retrospective cohort study.

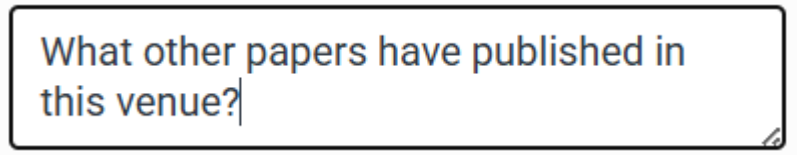
- Relevant categories: Betacoronavirus, Cardiovascular Diseases, Comorbidity, Coronavirus Infections, Fibrin Fibrinogen Degradation Products, Mortality, Pandemics, Patient Isolation, Pneumonia, …

Fine-grained classification can be viewed as a retrieval task.
Query: Paper; Candidates: Category Names

Link Prediction

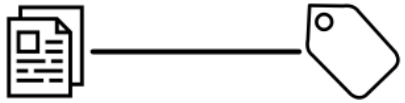
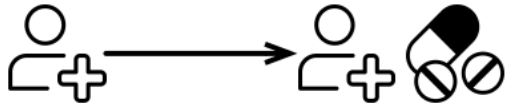
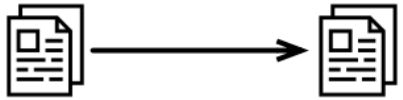
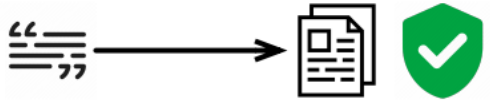
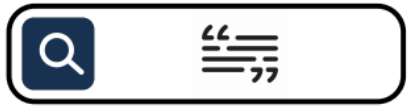
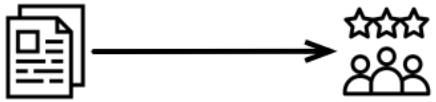






Link prediction can be viewed as a retrieval task.
Query: Paper; Candidates: Papers

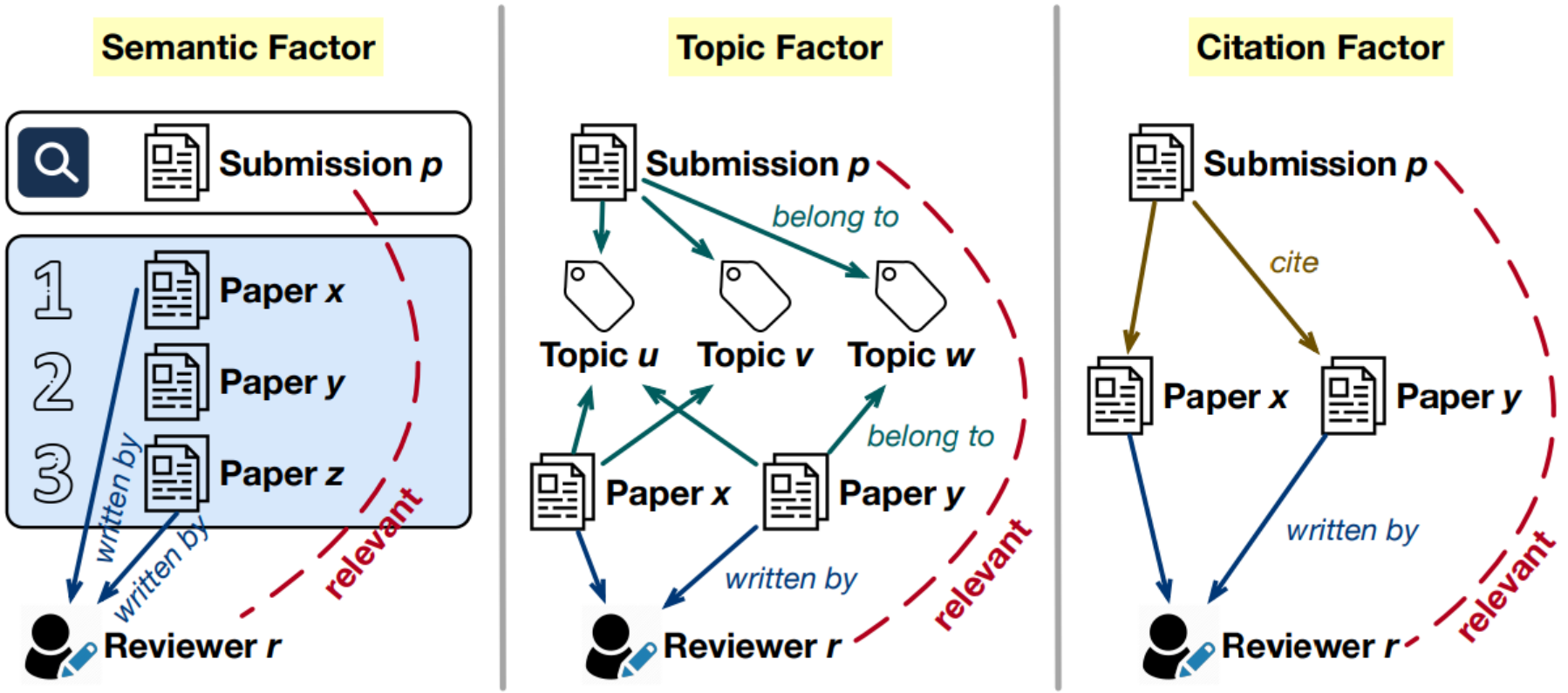
Fundamental Retrieval Tasks vs. Advanced Retrieval Applications

Fundamental Retrieval Tasks	Advanced Retrieval Applications	
Paper Classification 	Patient-to-Patient Matching 	Given a patient summary, find similar patients/clinical case reports.
Link Prediction 	Claim Verification 	Given a scientific claim, find relevant papers (and predict their stance).
Literature Retrieval 	Peer Review Assignment 	Given a paper submission, find expert reviewers.

- Why are some tasks more complex?
  - Multiple factors should be considered when judging the relevance.

Multiple Factors for Judging Relevance

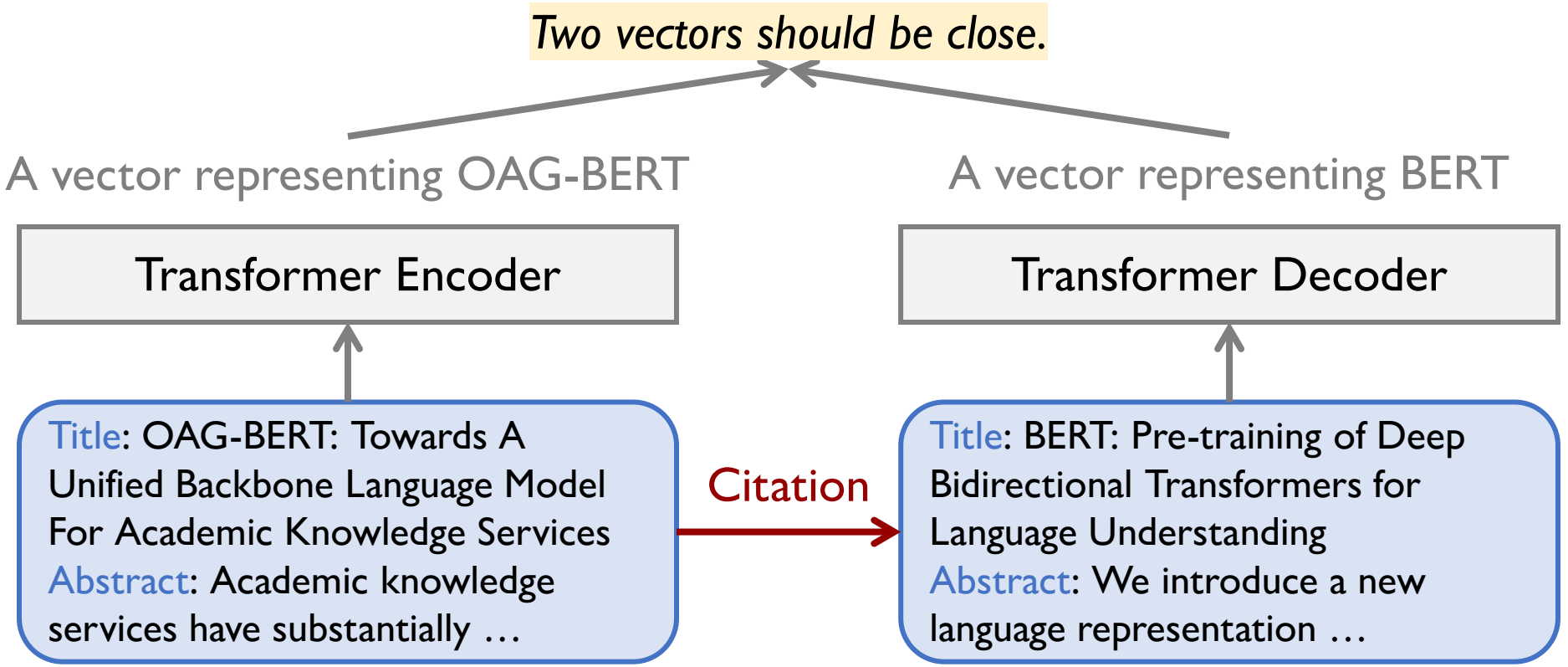
- Example: Paper-Reviewer Matching
  - Why is a pair of (Paper, Reviewer) relevant?



- Multiple factors exist in other tasks (e.g., Patient-to-Article Matching) as well.

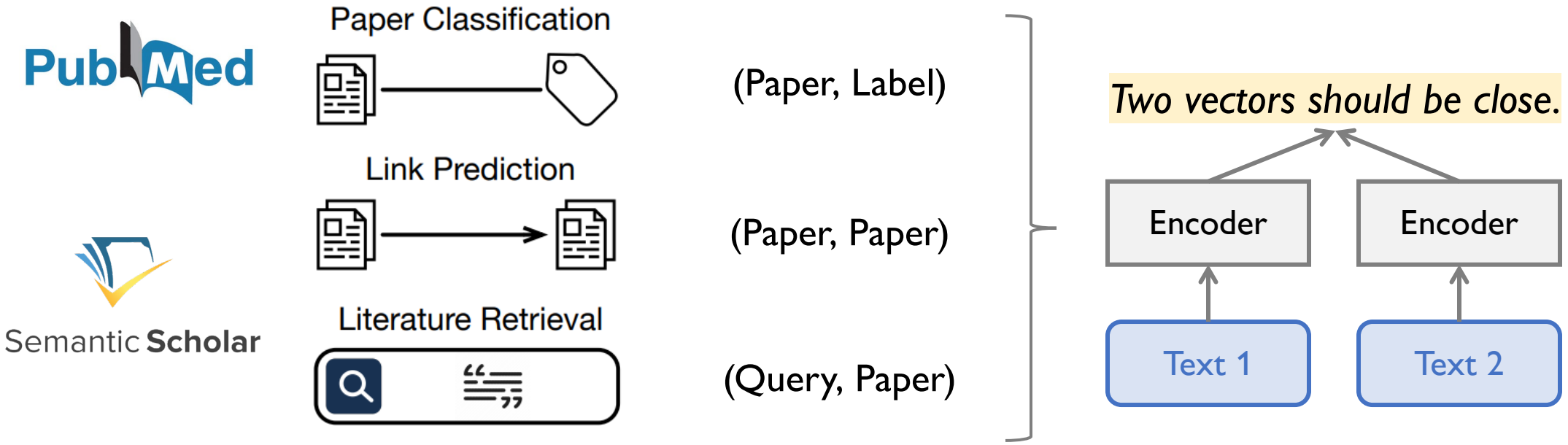
Contrastive Learning for a Fundamental Task

- E.g., Link Prediction
  - Step 1: Collect a large number of papers with citation information.
  - Step 2: Train an LLM with such citation information.



Contrastive Learning for an Advanced Task – A Naïve Way

- Each factor (topic, citation, and semantic) relies on one fundamental retrieval task.
- Directly combining pre-training data from different tasks to train a model?

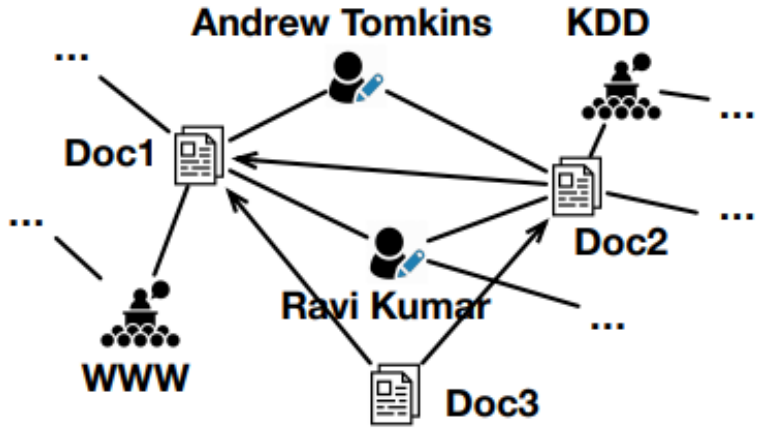


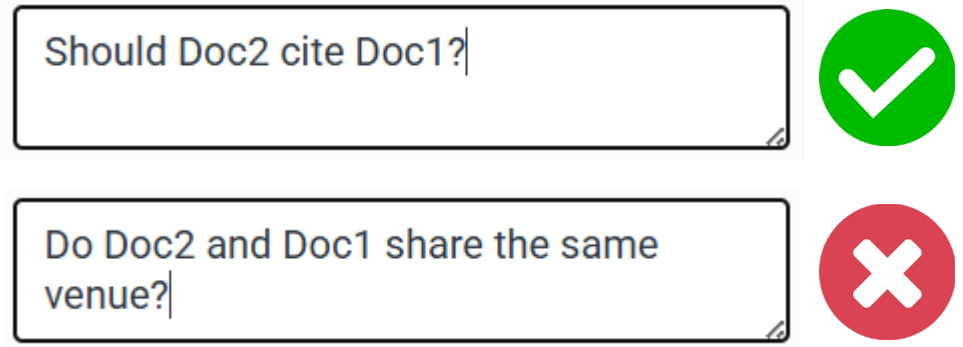
- Task Interference: The model is confused by different types of "relevance".

A Toy Example of Task Interference

- Imagine you have two "tasks".
  - Task 1: Given Paper1 and Paper2, predict if Paper1 should cite Paper2.
  - Task 2: Given Paper1 and Paper2, predict if Paper1 and Paper2 share the same venue.

- What if we directly merge the collected relevant (paper, paper) pairs for these two tasks?
  - Is (Doc2, Doc1) relevant?
  - The model does not know which task you are referring to, so it will get confused!





SciMult [Zhang et al., EMNLP 2023]

Pre-training Multi-task Contrastive Learning Models for Scientific Literature Understanding

Yu Zhang[♣]†*, **Hao Cheng**[♠]*, **Zhihong Shen**[♡], **Xiaodong Liu**[♠], **Ye-Yi Wang**[♡], **Jianfeng Gao**[♠]
[♣] University of Illinois at Urbana-Champaign [♠] Microsoft Research
[♡] Microsoft Search, Assistant and Intelligence
yuz9@illinois.edu {chehao,zhihosh,xiaodl,yeyiwang,jfgao}@microsoft.com

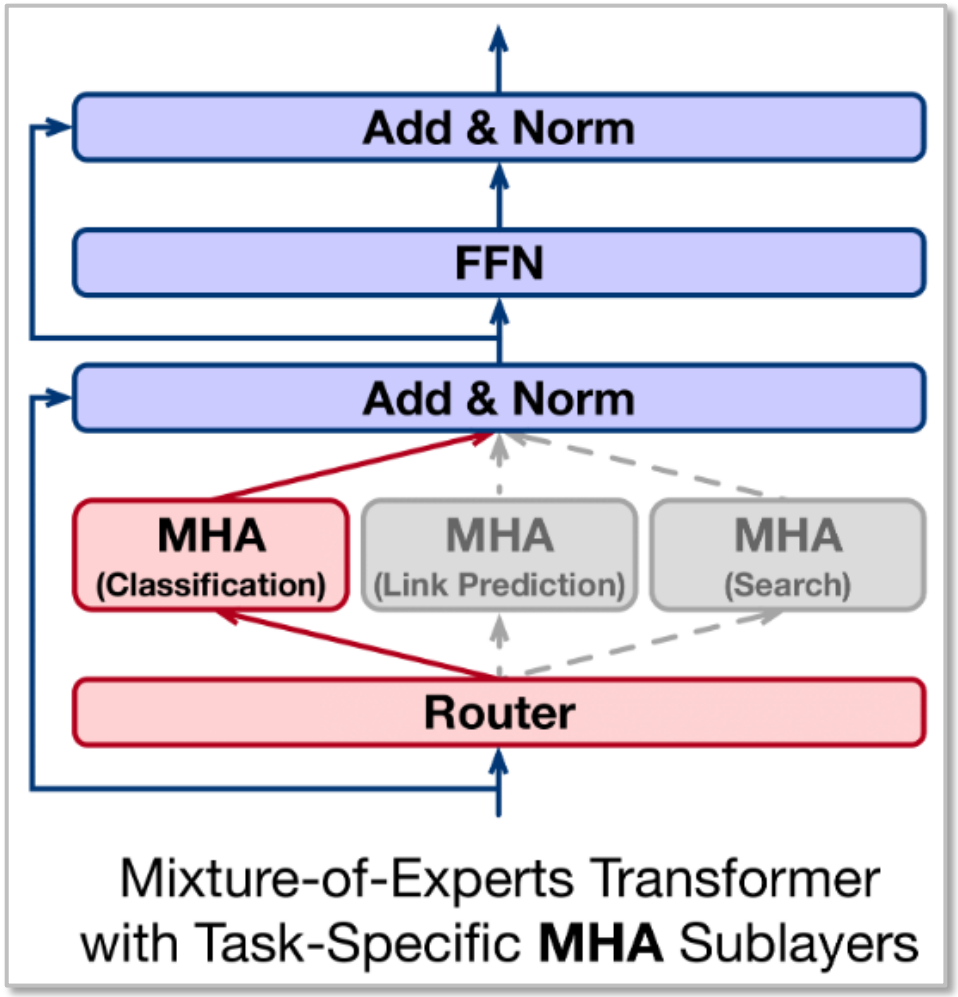
Abstract

Scientific literature understanding tasks have gained significant attention due to their potential to accelerate scientific discovery. Pre-trained language models (LMs) have shown effectiveness in these tasks, especially when tuned via contrastive learning. However, jointly utilizing pre-training data across multiple heterogeneous tasks (*e.g.*, extreme multi-label paper classification, citation prediction, and literature search) remains largely unexplored. To

models (LMs) (Beltagy et al., 2019; Gu et al., 2021; Liu et al., 2022) in these tasks as they generate high-quality scientific text representations, especially when the LMs are further tuned via contrastive learning. For example, MICoL (Zhang et al., 2022) proposes a metadata-induced contrastive learning that can perform extreme multi-label paper classification with more than 10,000 classes; SPECTER (Cohan et al., 2020) and SciNCL (Ostendorff et al., 2022) leverage citation information to create train-

Tackling Task Interference: Mixture-of-Experts Transformer

- A typical Transformer layer
  - **1** Multi-Head Attention (MHA) sublayer
  - **1** Feed Forward Network (FFN) sublayer
- A Mixture-of-Experts (MoE) Transformer layer
  - Multiple MHA sublayers
  - **1** FFN sublayer
  - (Or 1 MHA & Multiple FFN)
- Specializing some parts of the architecture to be an "expert" of one task
- The model can learn both commonalities and characteristics of different tasks.

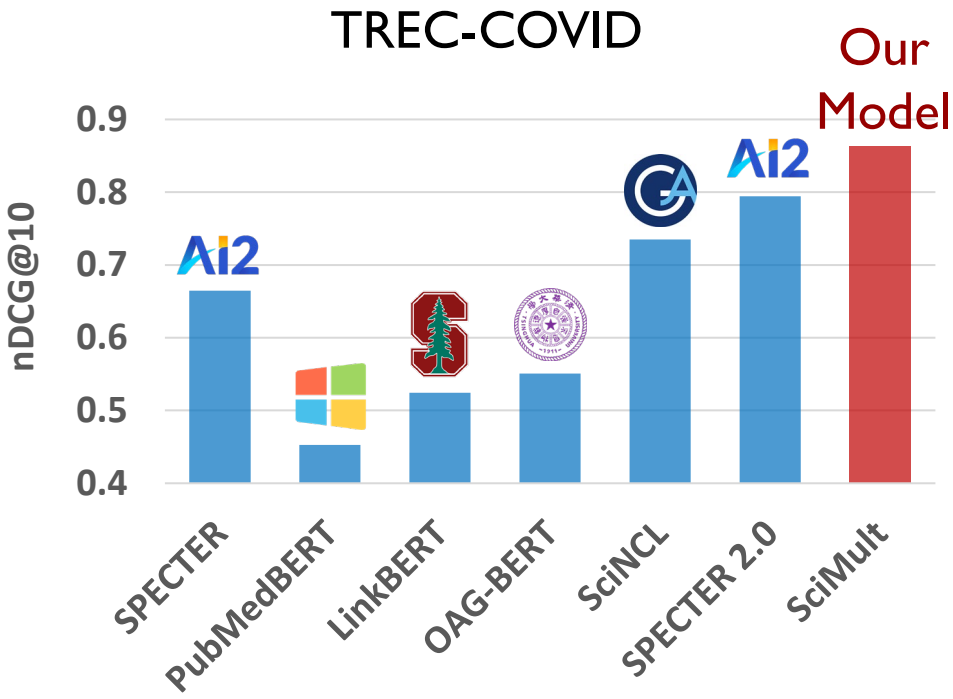


Comparison with Previous Approaches

- New SOTA on the PMC-Patients benchmark (patient-to-article retrieval)
- Outperforming previous scientific pre-trained language models in classification, link prediction, literature retrieval (TREC-COVID), paper recommendation, and claim verification (SciFact)

Patient-to-Article Retrieval (PAR) Leaderboard

	Model	MRR (%)	P@10 (%)	nDCG@10 (%)	R@1k (%)
Our Model 1 June 25, 2023	DPR (SciMult-MHAExpert) *UIUC/Microsoft* (Zhang et al. 2023)	29.89	9.35	13.79	53.71
2 Apr 5, 2023	RRF *Tsinghua University* (Zhao et al. 2023)	29.86	8.86	13.36	49.45



https://pmc-patients.github.io/

Chain-of-Factors [Zhang et al., WWW 2025]

Chain-of-Factors Paper-Reviewer Matching

Yu Zhang
Texas A&M University
College Station, TX, USA
yuzhang@tamu.edu

Yanzhen Shen
University of Illinois
Urbana-Champaign
Urbana, IL, USA
yanzhen4@illinois.edu

SeongKu Kang
Korea University
Seoul, South Korea
seongkukang@korea.ac.kr

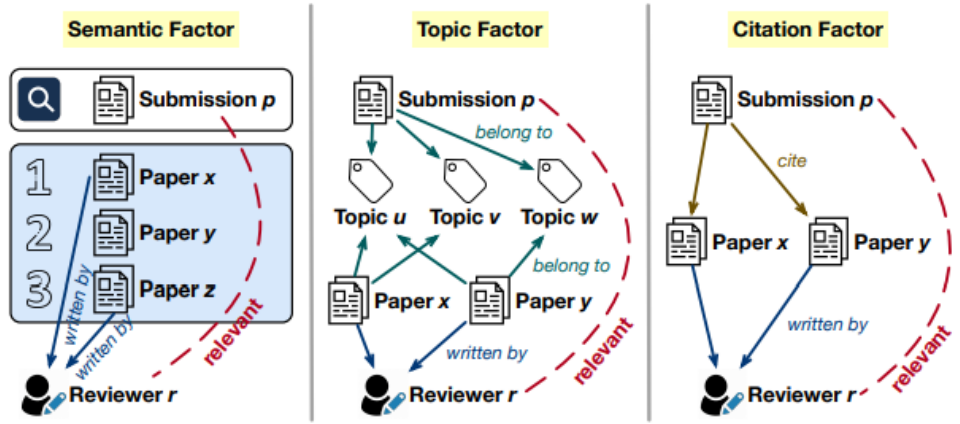
Xiusi Chen
University of Illinois
Urbana-Champaign
Urbana, IL, USA
xiusic@illinois.edu

Bowen Jin
University of Illinois
Urbana-Champaign
Urbana, IL, USA
bowenj4@illinois.edu

Jiawei Han
University of Illinois
Urbana-Champaign
Urbana, IL, USA
hanj@illinois.edu

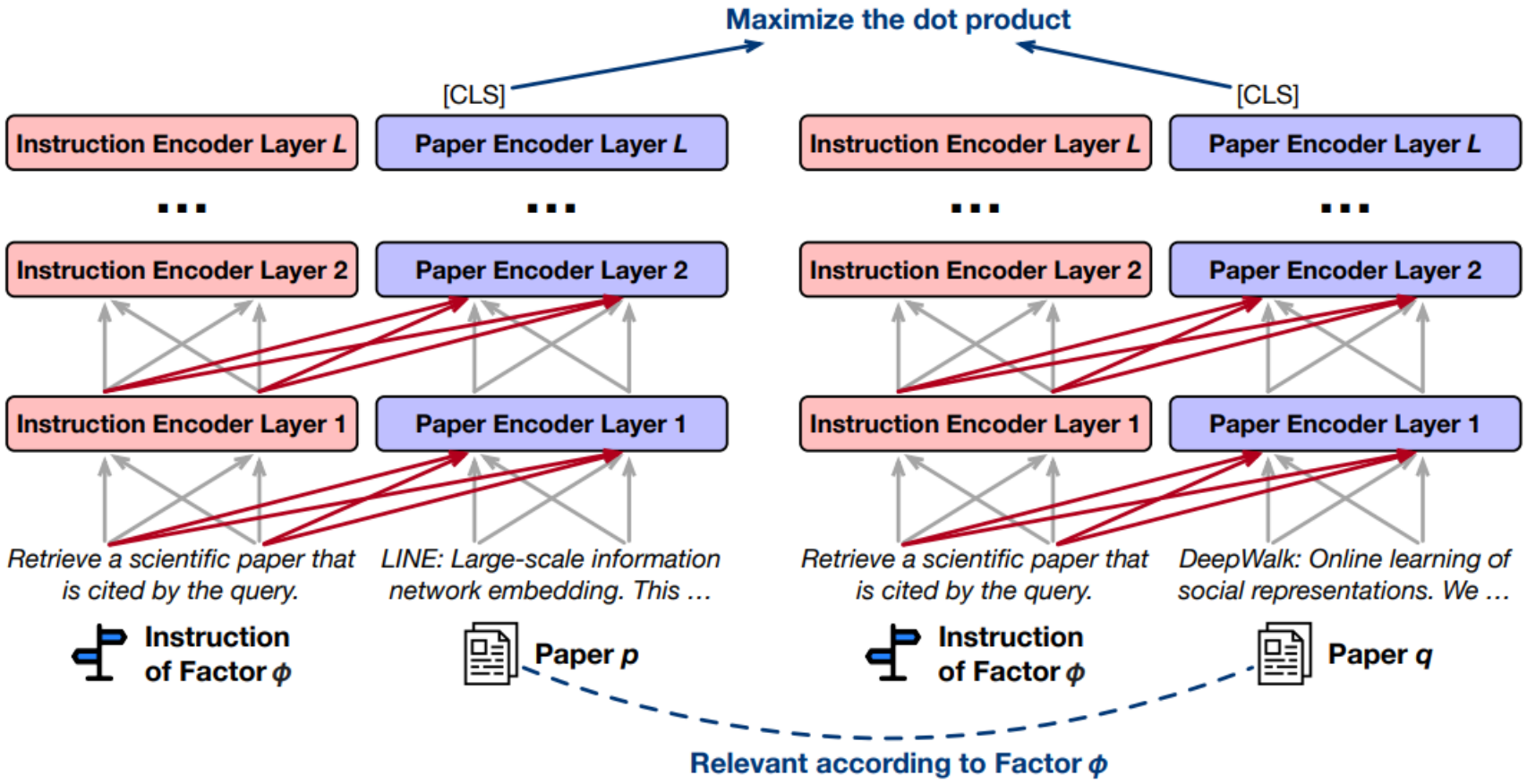
Abstract

With the rapid increase in paper submissions to academic conferences, the need for automated and accurate paper-reviewer matching is more critical than ever. Previous efforts in this area have considered various factors to assess the relevance of a reviewer's expertise to a paper, such as the semantic similarity, shared topics, and citation connections between the paper and the reviewer's previous works. However, most of these studies focus on only one factor, resulting in an incomplete evaluation of the paper-reviewer relevance. To address this issue, we propose a unified model for



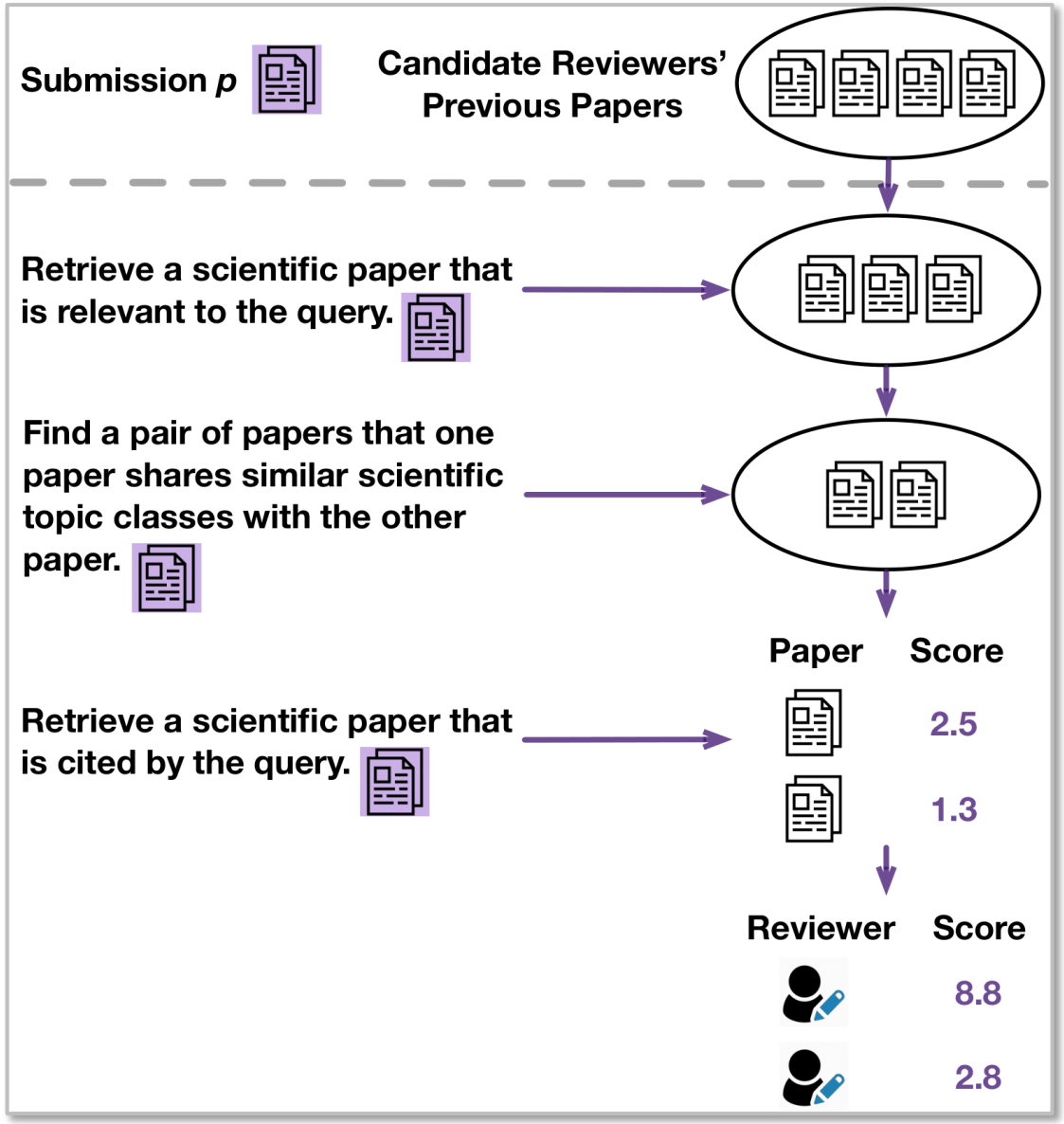
Tackling Task Interference: Instruction Tuning

- Using a factor-specific instruction to guide the paper encoding process
- The instruction serves as the context of the paper.
- The paper does NOT serve as the context of the instruction.



Chain-of-Factors Reasoning

- Consider semantic, topic, and citation factors in a step-by-step, coarse-to-fine manner.
- Step 1: Semantic relevance serves as the coarsest signal to filter totally irrelevant papers.
- Step 2: Then, we can classify each submission and each relevant paper to a fine-grained topic space and check if they share common topics.
- Step 3: After confirming that a submission and a reviewer's previous paper have common topics, the citation link between them will become an even stronger signal, indicating that the two papers may focus on the same task or datasets.



Comparison with Previous Approaches

- Public benchmark datasets
  - Expert C judges whether Reviewer A is qualified to review Paper B.
- Outperforming the Toronto Paper Matching System (TPMS, used by Microsoft CMT)

	SciRepEval [44]					SIGIR [19]					KDD				
	Soft P@5	Soft P@10	Hard P@5	Hard P@10	Average	Soft P@5	Soft P@10	Hard P@5	Hard P@10	Average	Soft P@5	Soft P@10	Hard P@5	Hard P@10	Average
TPMS [7]	62.06**	53.74**	31.40**	24.86**	43.02**	39.73**	38.36**	17.81**	17.12**	28.26**	17.01**	16.78**	6.78**	7.24**	11.95**
SciBERT [6]	59.63**	54.39**	28.04**	24.49**	41.64**	34.79**	34.79**	14.79**	15.34**	24.93**	28.51**	27.36**	12.64**	12.70**	20.30**
SPECTER [9]	65.23**	**56.07**	32.34**	25.42	44.77**	39.73**	40.00**	16.44**	16.71**	28.22**	34.94**	30.52**	15.17**	**13.28**	23.48**
SciNCL [34]	66.92**	55.42**	34.02*	25.33	45.42**	40.55**	39.45**	17.81**	17.40*	28.80**	36.21**	30.86**	15.06**	12.70**	23.71**
COCO-DR [56]	65.05**	55.14**	31.78**	24.67**	44.16**	40.00**	40.55*	16.71**	17.53	28.70**	35.06**	29.89**	13.68**	12.13**	22.69**
SPECTER 2.0 CLF [44]	64.49**	55.23**	31.59**	24.49**	43.95**	39.45**	38.63**	16.16**	16.30**	27.64**	34.37**	30.63**	14.48**	12.64**	23.03**
SPECTER 2.0 PRX [44]	66.36**	55.61**	34.21	**25.61**	45.45**	40.00**	38.90**	19.18**	16.85**	28.73**	37.13	31.03	15.86**	13.05*	24.27*
Our Model CoF	**68.47**	55.89	**34.52**	25.33	**46.05**	**45.57**	**41.69**	**22.47**	**17.76**	**31.87**	**37.63**	**31.09**	**16.13**	13.08	**24.48**

: semantic-based method : topic-based method : citation-based method

Thank You!

Course Website: https://yuzhang-teaching.github.io/CSCE670-F25.html