

# Recommender Systems in the Era of Large Language Models (LLMs)



Jiatong Li<sup>1</sup>, Zihuai Zhao<sup>1</sup>, Yunqing Liu<sup>1</sup>, Yiqi Wang<sup>2</sup>, Wenqi Fan<sup>1</sup>



<sup>1</sup>The Hong Kong Polytechnic University

<sup>2</sup>National University of Defense Technology

Website QR Code

Website (Slides): [https://advanced-recommender-systems.github.io/llms\\_rec\\_tutorial/](https://advanced-recommender-systems.github.io/llms_rec_tutorial/)

Survey Paper: "Recommender systems in the era of large language models (LLMs)." arXiv:2307.02046 (2023).

# Tutorial Outline

Introduction

Preliminaries

Pre-training

Fine-tuning

Prompting

Future Directions

- **Introduction** of RecSys in the era of LLMs (Dr. Wenqi Fan)
- **Preliminaries** of RecSys and LLMs (Yunqing Liu)
- **Pre-training** paradigms for adopting LLMs to RecSys (Jiatong Li)
- **Fine-tuning** paradigms for adopting LLMs to RecSys (Jiatong Li)
- **Prompting** paradigms for adopting LLMs to RecSys (Zihuai Zhao)
- **Future directions** of LLM-empowered RecSys (Dr. Yiqi Wang)



Website QR Code

# Recommender Systems (RecSys)

Age of Information Explosion



**Items** can be: Products, Friends, News, Movies, Videos, etc.

# Recommender Systems (RecSys)

- Recommendation has been widely applied in online services:
  - ❖ E-commerce, Content Sharing, Social Networking ...



## Product Recommendation

Frequently bought together



Total price: \$208.9

Add all three to Cart

Add all three to List



Amazon's recommendation algorithm drives **35%** of its sales [from McKinsey, 2012]

# Recommender Systems (RecSys)

- Recommendation has been widely applied in online services:
  - ❖ E-commerce, Content Sharing, Social Networking ...

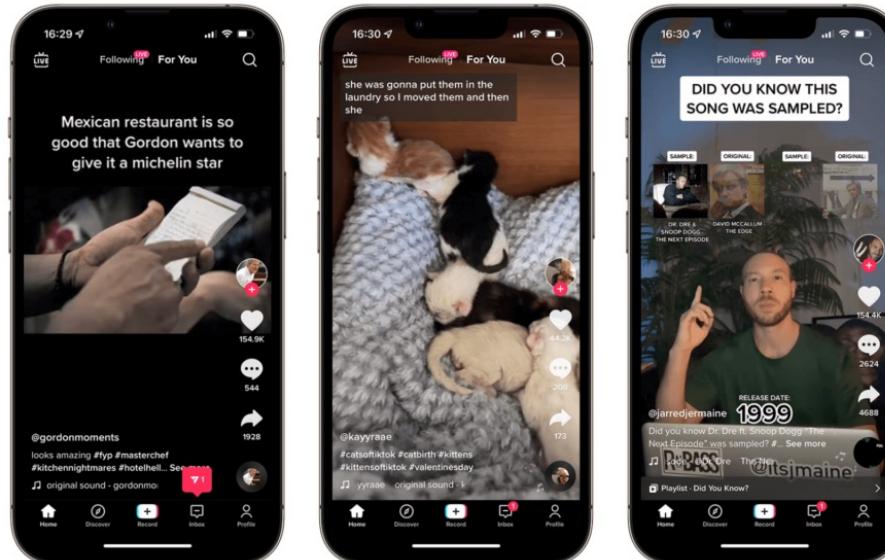


---

News/Video/Image Recommendation

TikTok's recommendation algorithm  
Top 10 Global Breakthrough  
Technologies in 2021

MIT  
Technology  
Review



# Large Language Models (LLMs) are Changing Our Lives



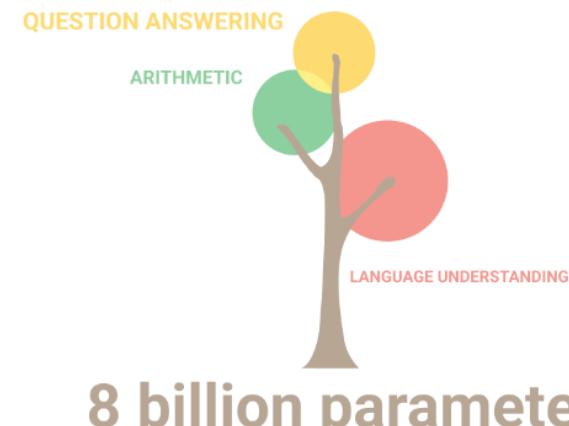
Meta



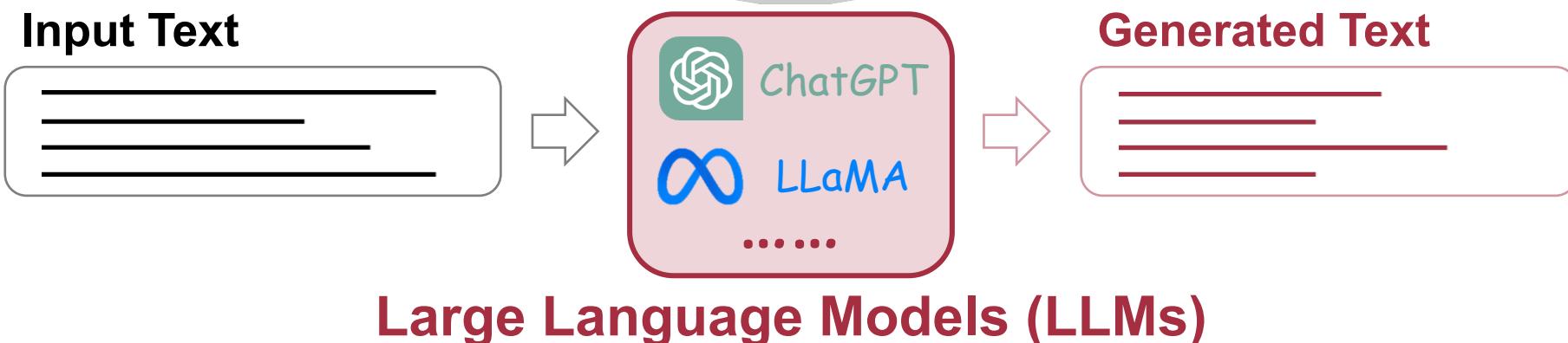
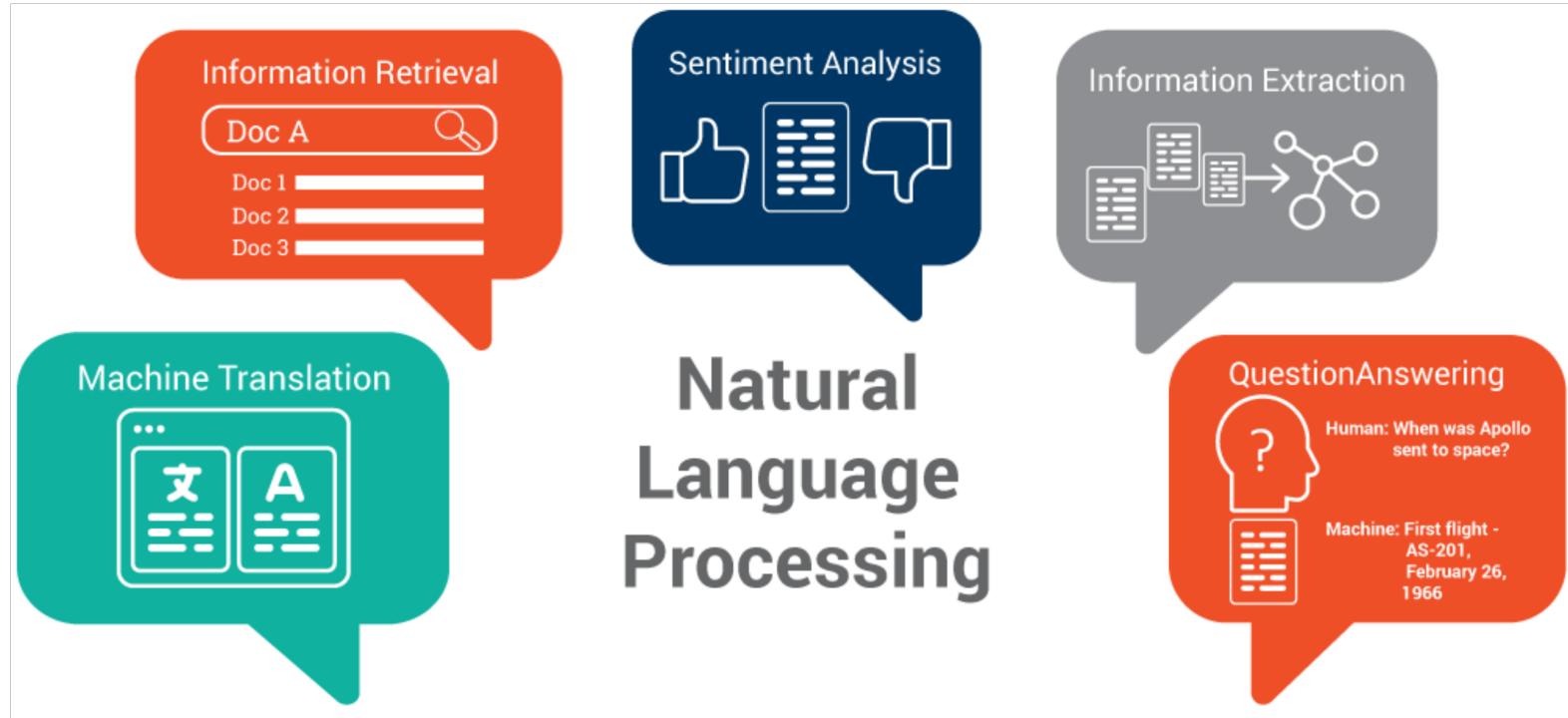
Google AI



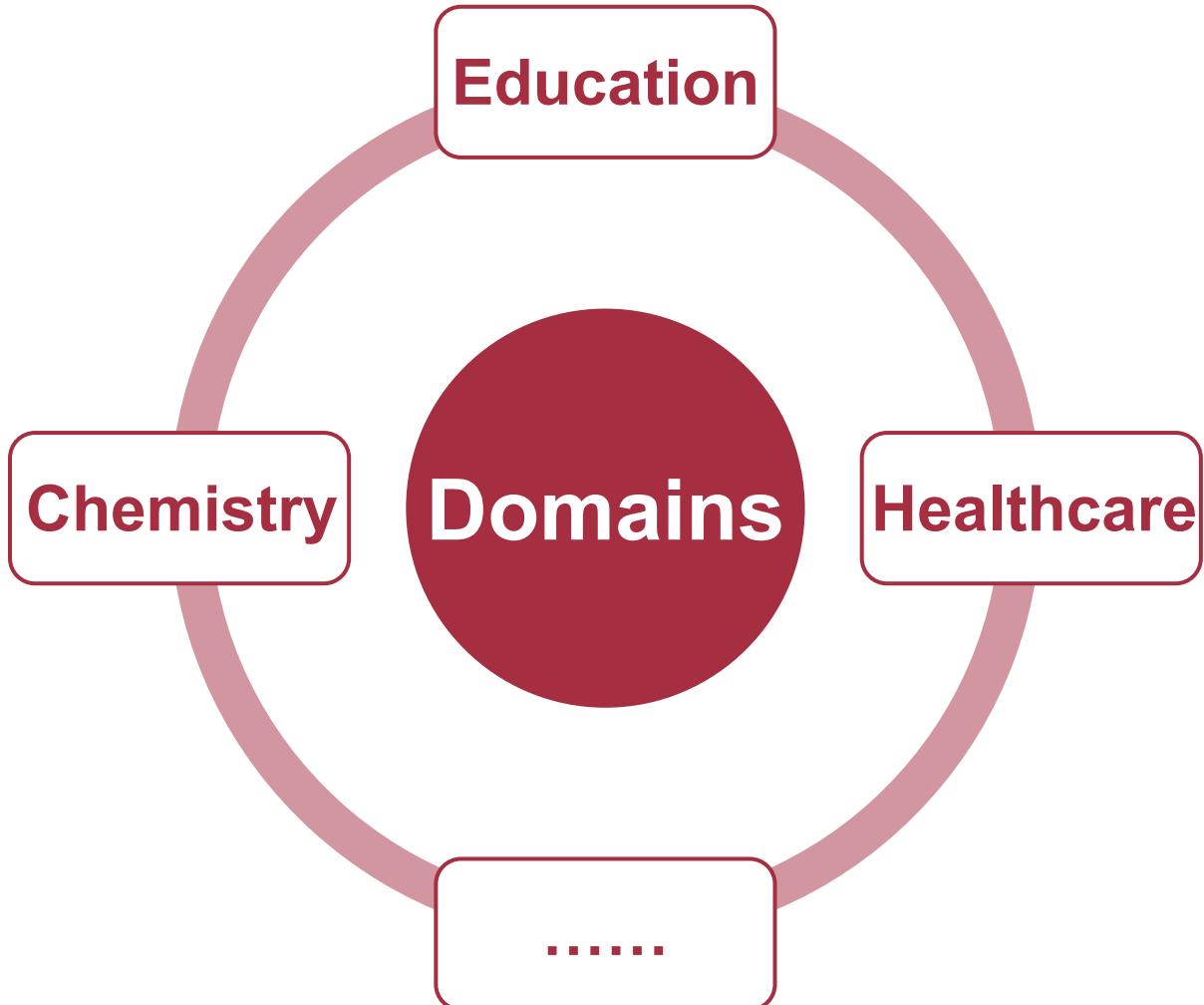
.....



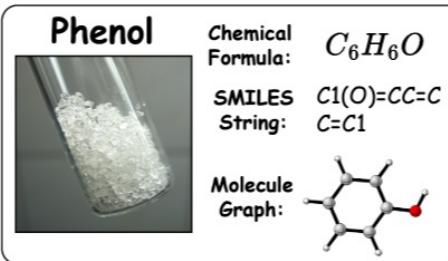
# LLMs in Natural Language Processing



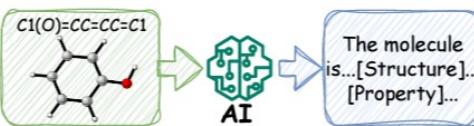
# LLMs in Downstream Domains



## □ Molecule discovery, etc.



(a) Molecule Representations.



(b) Molecule Captioning.



**ChatGPT**

**(a) Molecule Captioning**

Please show me a description of this molecule:  
"C1=CC=C(C=C1)OC2=CC=CC=C2"

The molecule is an aromatic ether in which the oxygen is attached to two phenyl substituents. It has been found in muscat grapes and vanilla. It has a role as a plant metabolite.

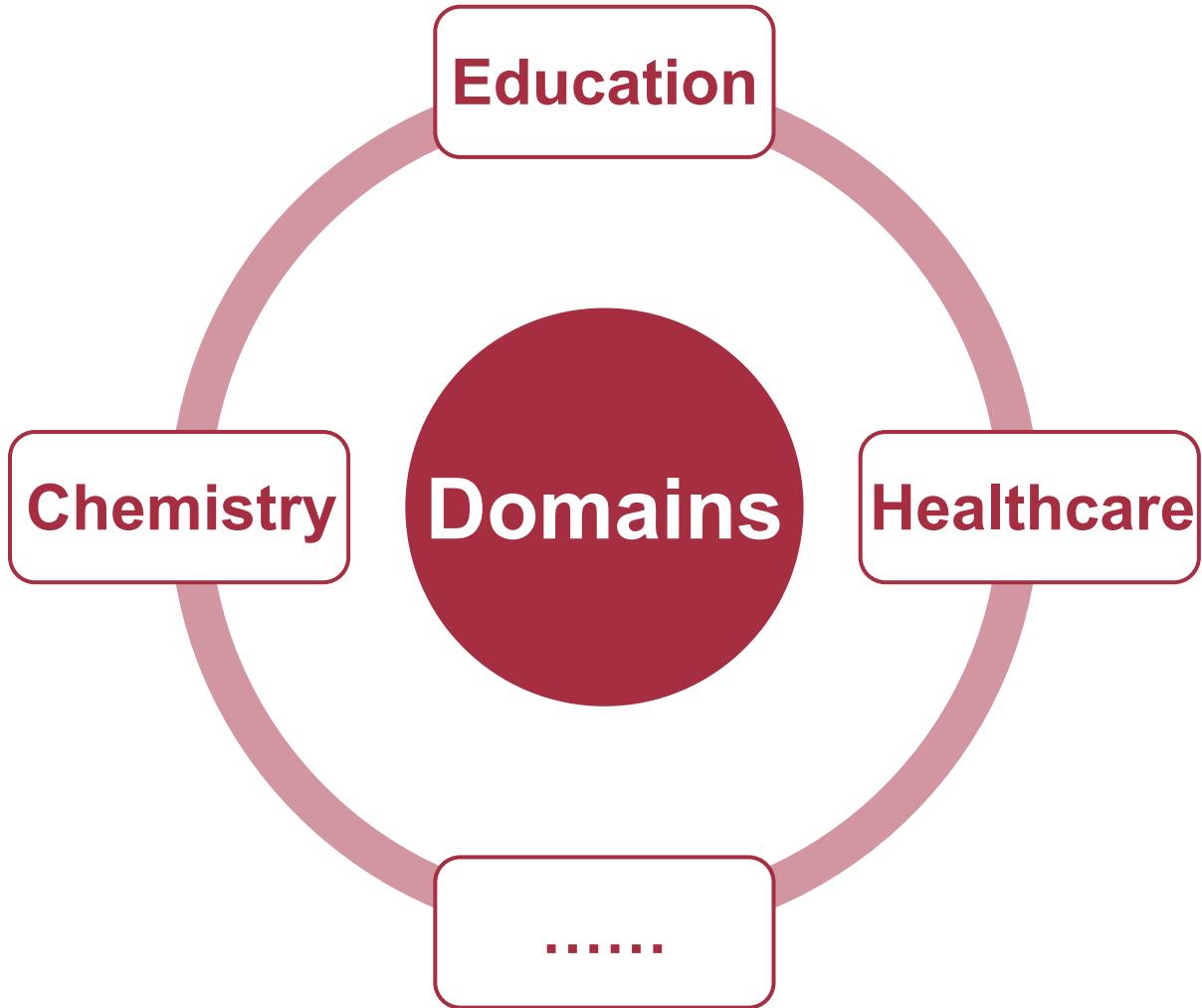
**(b) Text-based Molecule Generation**

Help me generate a molecule based on the given description:

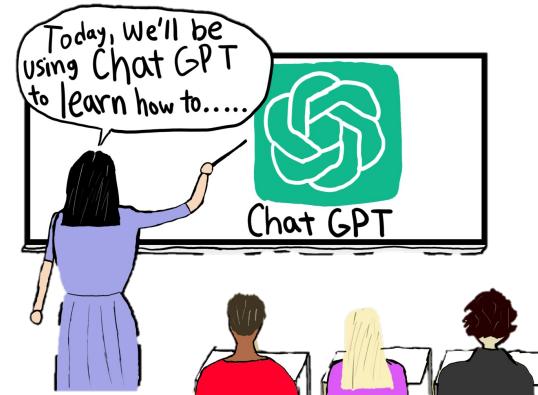
"The molecule is a quinolinemonocarboxylate that is the conjugate base of xanthurenic acid, obtained by deprotonation of the carboxy group. It has a role as an animal metabolite. It is a conjugate base of a xanthurenic acid."

$C1=CC2=C(C(=C1)[O-])NC(=CC2=O)C(=O)O$

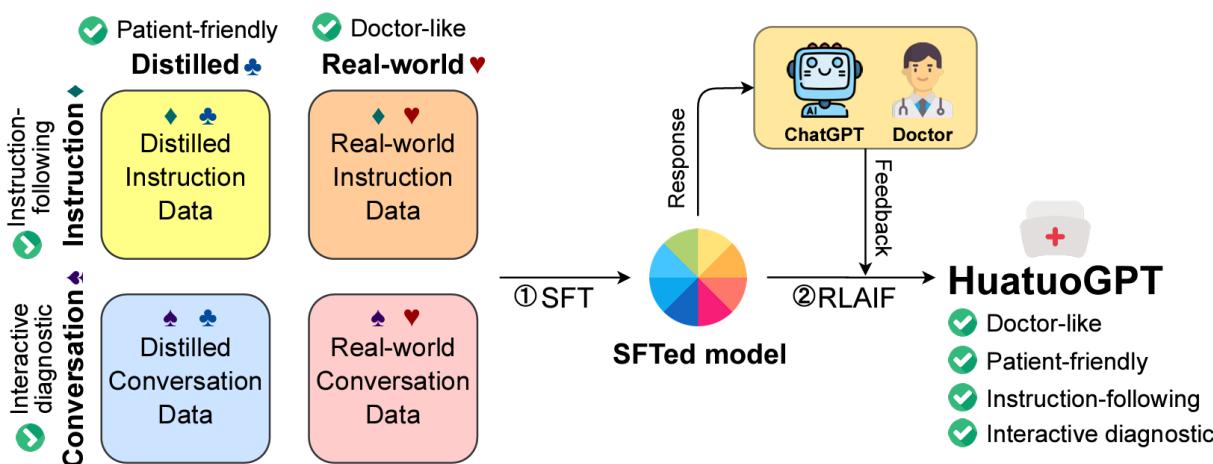
# LLMs in Downstream Domains



❑ Curriculum & Teaching, etc.



❑ Medical consultation, etc.



# LLMs in RecSys

## Top-K Recommendation

A user recently watched movies:



Based on the watch history, please recommend five **candidate movies** that the user might be interested in from the following list:

- 1    2    3    .....    6    .....

## Rating Prediction

Here is the **movie rating history** of a user:



Based on the above rating history of this user, please **rate** a movie named *John Wick: Chapter 4* with a range of 1-10 points.

## Conversational Recommendation

[User]: I recently watched a science fiction movie named *Interstellar*



Please recommend some ... to me.

[User]: .....

[User]: But I don't like ... because ...  
Could you recommend other ... .

## Explanation Generation

A new movie named *The Godfather Part II* is recommended to a user,



who has recently watched movies:



Please **explain** why this new movie is recommended to the user.



ChatGPT



GPT-J

Large Language Models (LLMs)  
for Recommender Systems



LLaMA



Vicuna

...

Based on the watch history, I assume this user is interested in movies of ... genres and ... actor/actress. Here are the top five **candidate movies**:

- 3    1    4    2    8

The movie *John Wick: Chapter 4* has the similar ... to ... movie in the rating history.

Thus, the **rating** is likely to be 9.0.

[LLM]: Sure! Here are some ... recommended to you: ... .

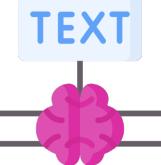
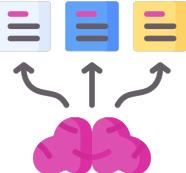
[LLM]: .....

[LLM]: My apologies! Here are ... .

This new movie is recommended to the user **because** the ... features of this new movie are similar to the ... of movies that recently watched by this user. **Thus**, the user may want to watch the recommended new movie.

# Potentials of LLMs in RecSys

As the **parameter size** of LLMs continues to **scale up** with a larger **training corpus** ...

- Language understanding and generation ability
  - ❖ LLMs can comprehend **human intentions** and generate **language responses** that are more human-like in nature.
- Generalization capability
  - ❖ LLMs can apply their learned knowledge to **fit various downstream tasks**, even **without being fine-tuned** on specific tasks.
- Reasoning capability
  - ❖ LLMs can generate the outputs with **step-by-step reasonings** to support complex **decision-making processes**.

# Language Understanding & Generation

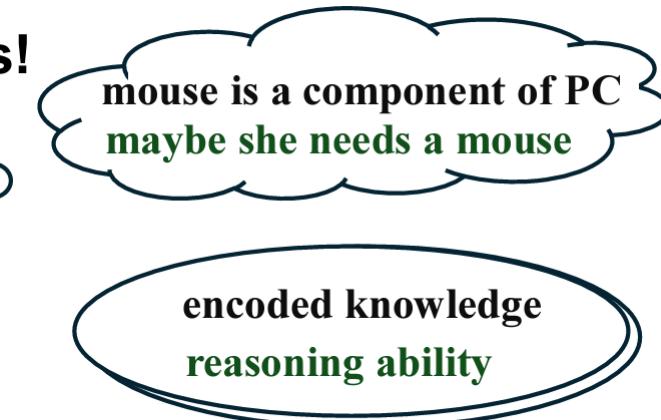
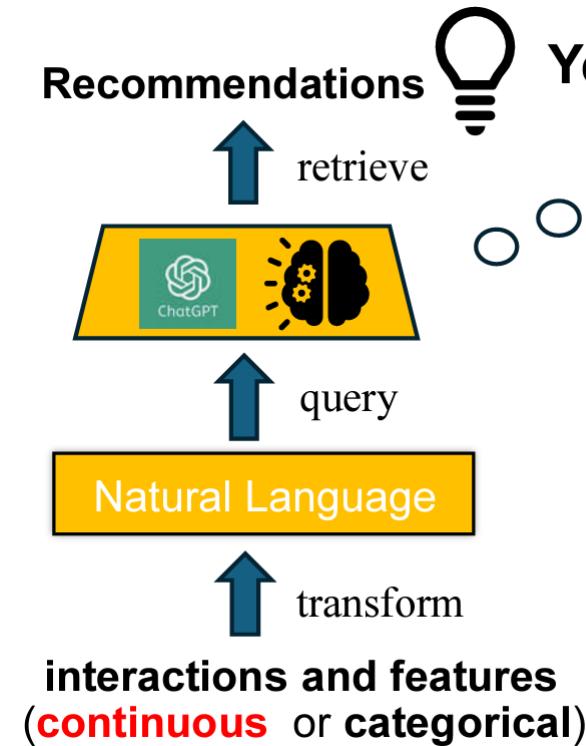
- Sufficiently capture **textual knowledge** about users and items
  - ❖ Rich **textual side information** about users and items in RecSys
  - ❖ Diverse **open-world knowledge** encoded in LLMs

User ID: 0057 Item ID: 0046

**Item Title:** Wet n Wild Mega Last Lip Color 908C Sugar Plum Fairy

**Review:** The color is a perfect mix of dark purple, red and pink. The only downside is the drying aspect of the lipstick, which I counteract by using lip balm before putting it on.

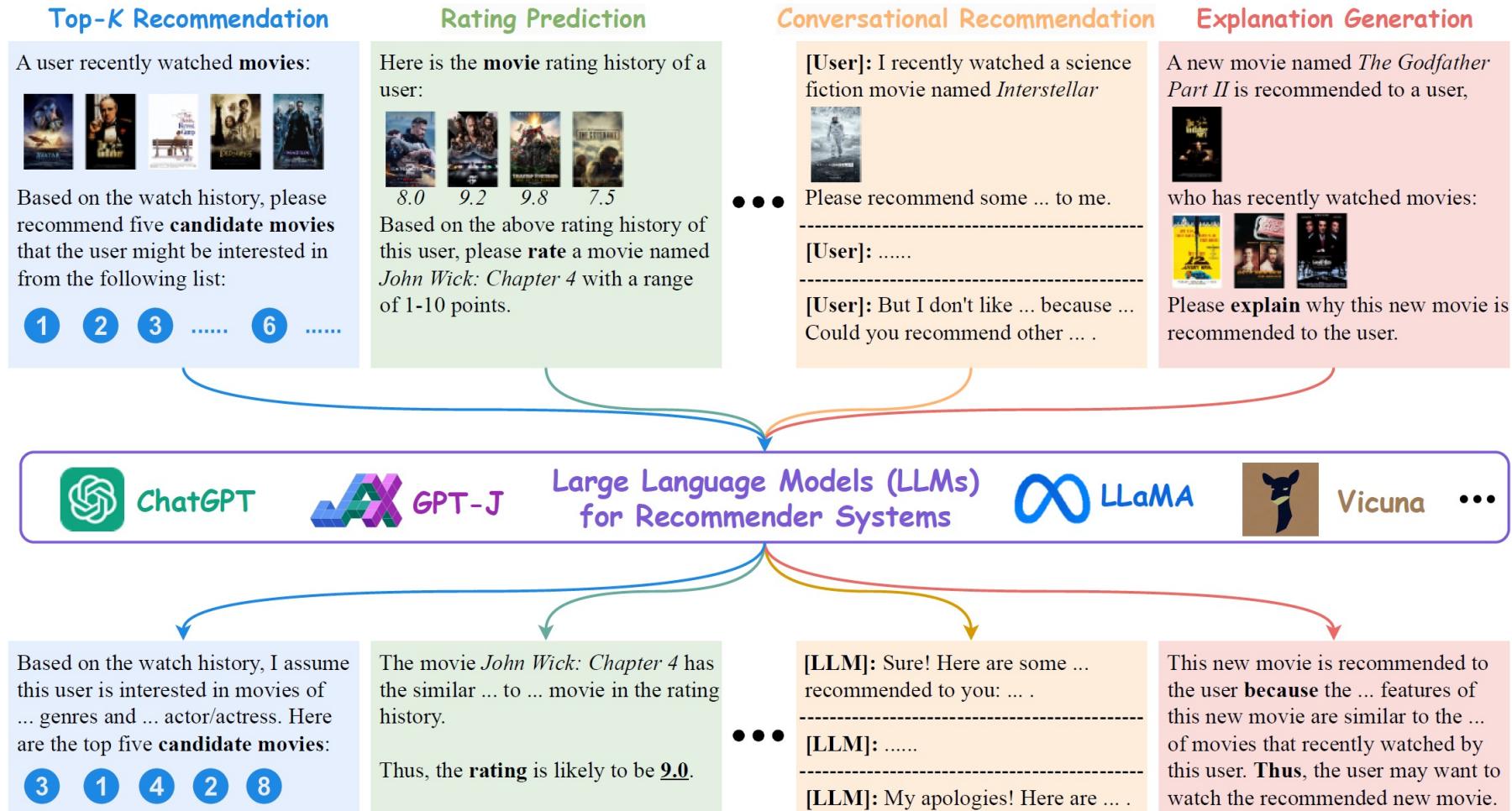
Knowledge of RecSys



e.g.,  
user\_1 has bought item\_2.  
user\_1 is a CS student.  
item\_2 is a computer.  
will user\_1 buy a mouse?

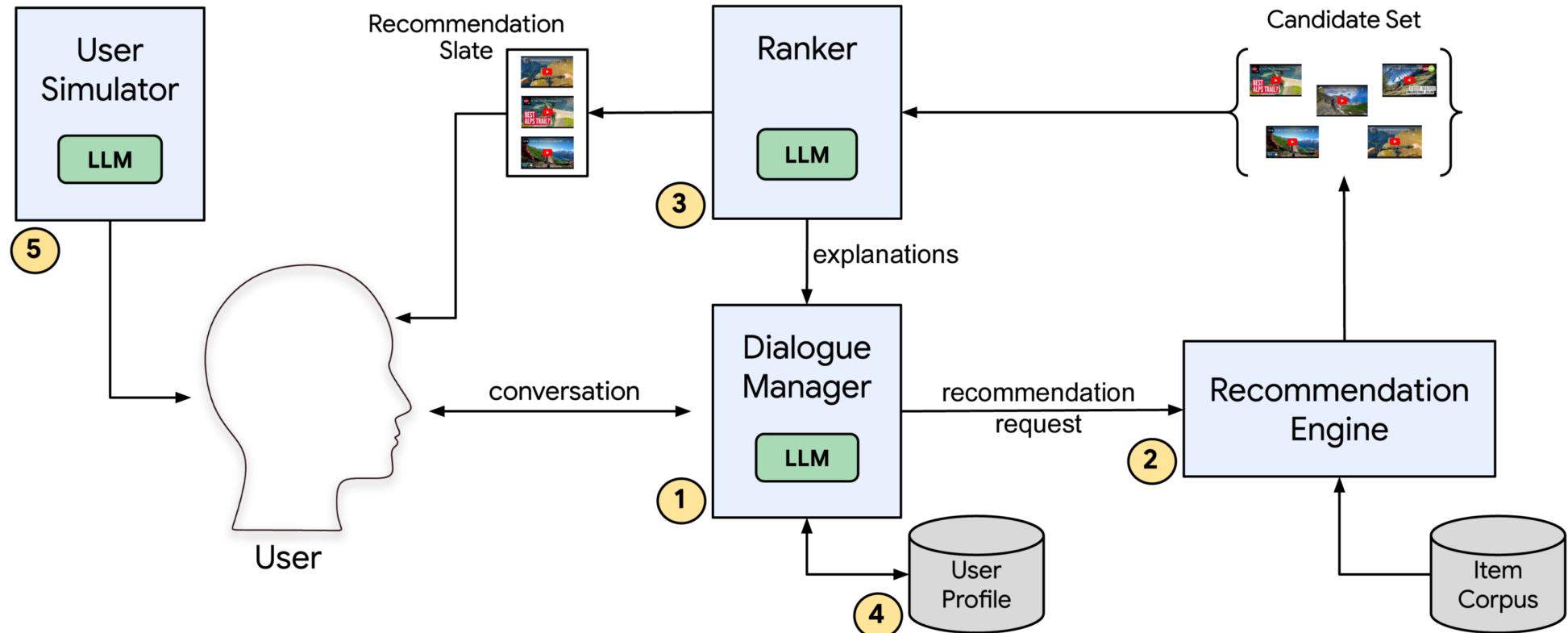
# Generalization

- Adapt to **various recommendation tasks** even without being fine-tuned
  - ❖ LLMs can apply their **learned knowledge** to address recommendation objectives
  - ❖ **Multi-task adaption** by providing appropriate task instructions or a few task demonstrations



# Reasoning

- Support complex **decision-making processes** in RecSys
  - ❖ Retrieve information from **large contexts** and control **multi-step** recommendation tasks
  - ❖ Generate outputs with **step-by-step reasoning** empowered by chain-of-thought prompting



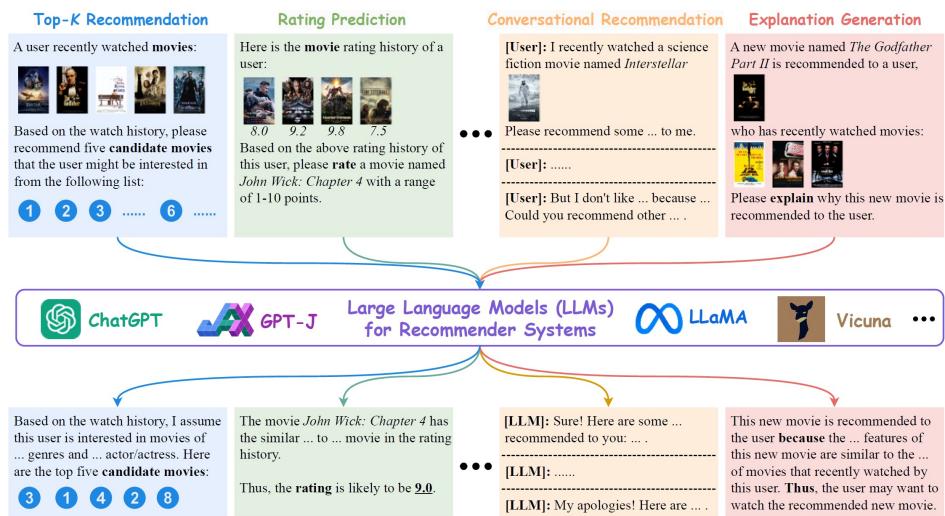
# A Comprehensive Survey Paper

## Recommender Systems in the Era of Large Language Models (LLMs)

Wenqi Fan, Zihuai Zhao, Jiatong Li, Yunqing Liu, Xiaowei Mei, Yiqi Wang,  
Zhen Wen, Fei Wang, Xiangyu Zhao, Jiliang Tang, and Qing Li

<https://arxiv.org/abs/2307.02046>

ICDM'2023  
Tutorial  
Website (Slides)



# Tutorial Outline

Introduction

Preliminaries

Pre-training

Fine-tuning

Prompting

Future Directions

- **Introduction** of RecSys in the era of LLMs (Dr. Wenqi Fan)
- **Preliminaries** of RecSys and LLMs (Yunqing Liu)
- **Pre-training** paradigms for adopting LLMs to RecSys (Jiatong Li)
- **Fine-tuning** paradigms for adopting LLMs to RecSys (Jiatong Li)
- **Prompting** paradigms for adopting LLMs to RecSys (Zihuai Zhao)
- **Future directions** of LLM-empowered RecSys (Dr. Yiqi Wang)



Website QR Code

Preliminaries

Pre-training

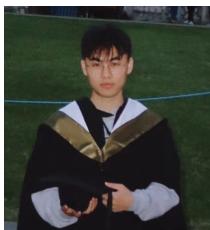
Fine-tuning

Prompting

Future  
Directions

# Overview

**Presenter:**  
Yunqing Liu



## RecSys

- Collaborative Filtering (CF)
- Content-based Recommendation
- Deep Recommender Systems

## LLMs

- Encoder-Only
- Decoder-Only
- Encoder-Decoder



Website QR Code

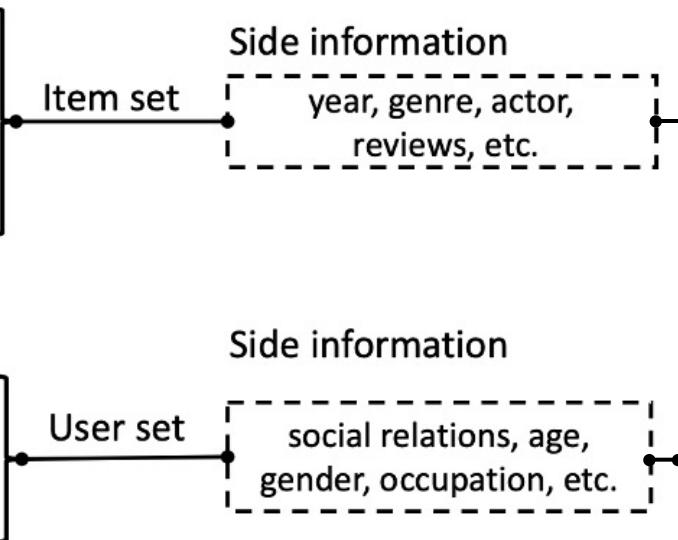
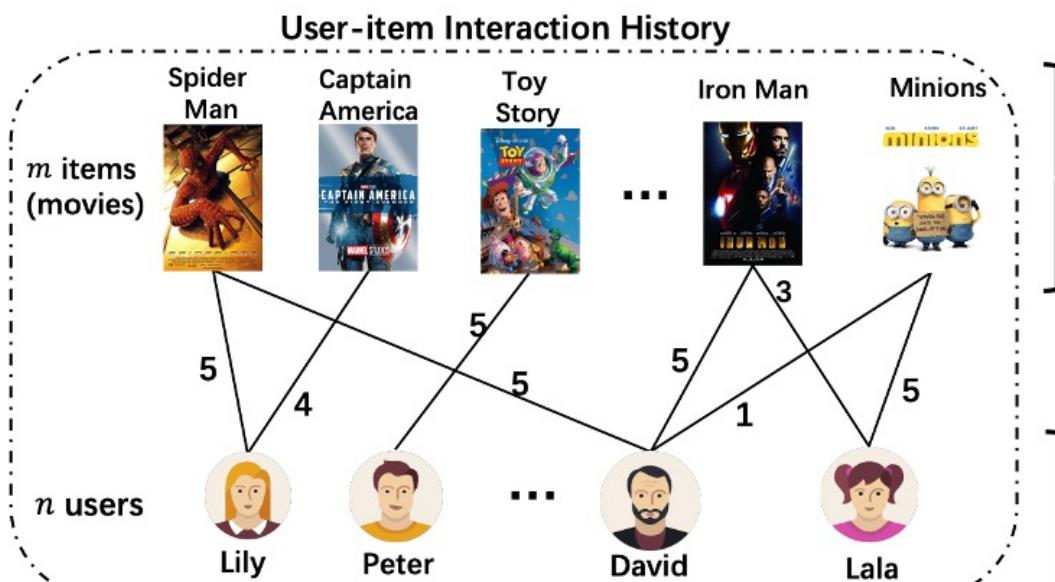
# Recommender Systems



Historical user-item interactions or additional side information (e.g., social relations, item's knowledge, etc.)



Predict how likely a user would interact with a target item (e.g., click, view, or purchase)

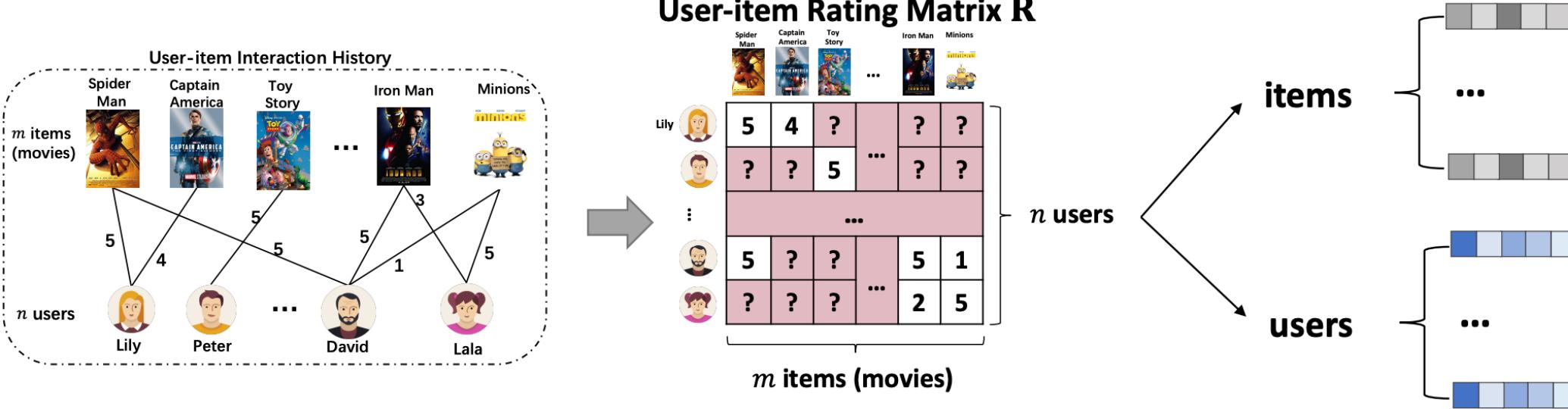


**Title:** Spider Man (2002)  
**Genre:** Action · Adventure · Sci-Fi  
**Actor:** Tobey Maguire, .....  
**Reviews:** 1. Considered as one of the most successful superhero movies ever made .....

**Name:** Peter  
**Social Relations:** David (Friend),...  
**Age:** 18  
**Gender:** Male  
**Occupation:** Student  
.....

# Collaborative Filtering (CF)

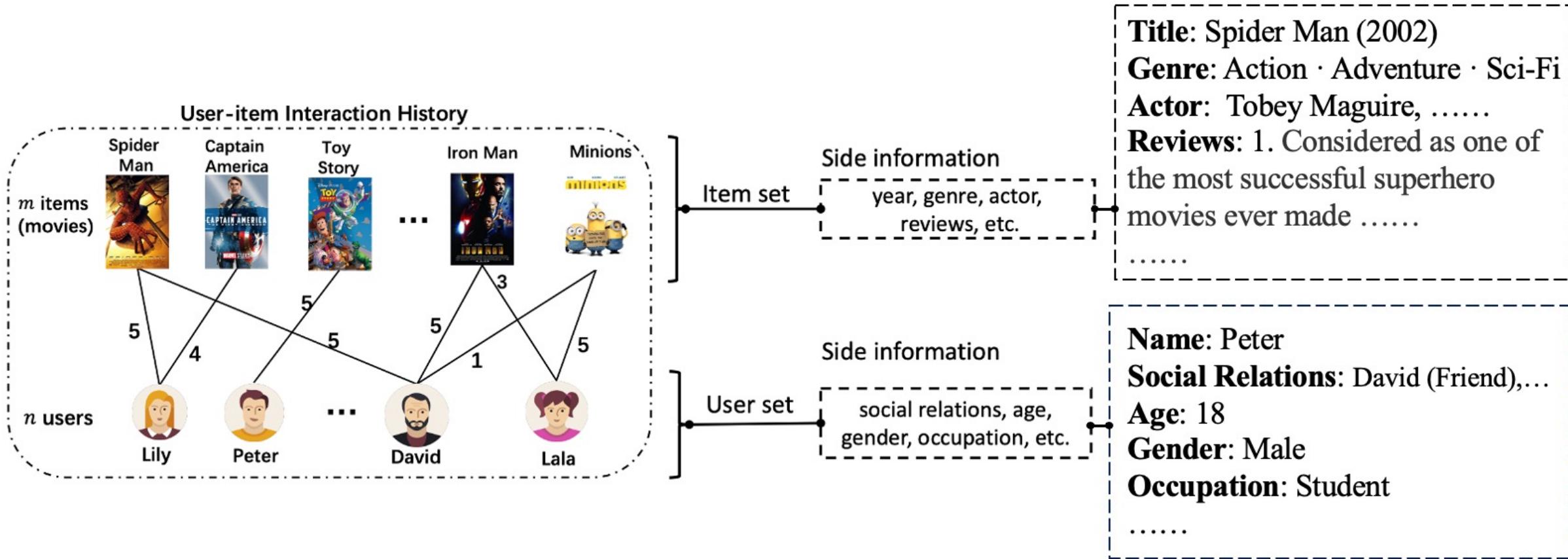
- The most well-known technique for recommendation
  - ❖ Similar users (with respect to their historical interactions) have similar preferences.
  - ❖ Modelling user's preferences on items based on their past interactions (e.g., ratings and clicks).
  
- Learning representations of users and items is the key to CF



**Task: predicting missing movie ratings in Netflix.**

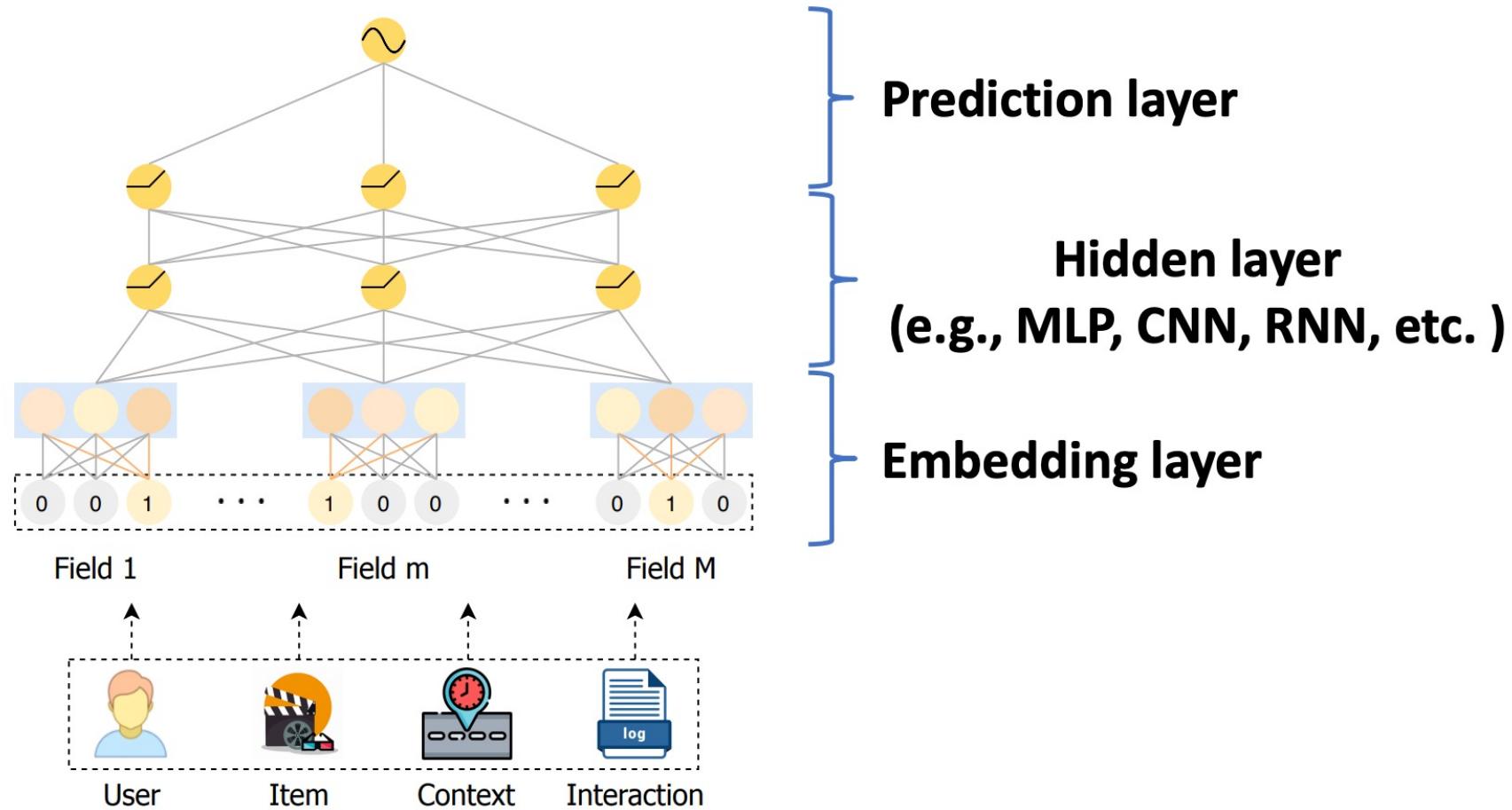
# Content-based Recommendation

- Taking advantage of **additional knowledge** about users or items
- **Enhancing** user and item representations for improving recommendation performance



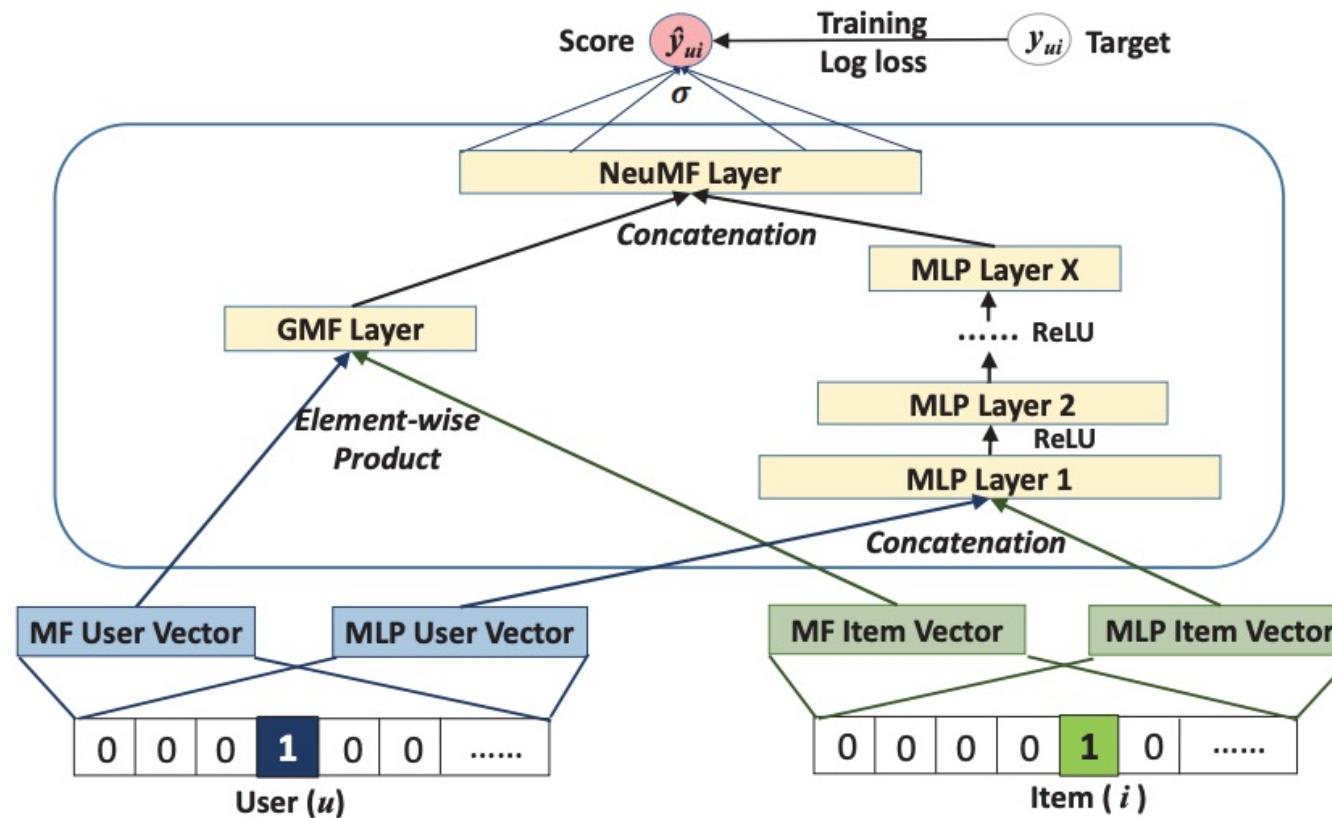
# Deep Recommender Systems

- Deep learning techniques have been effectively applied to develop recommender systems
- Remarkable representation learning capabilities



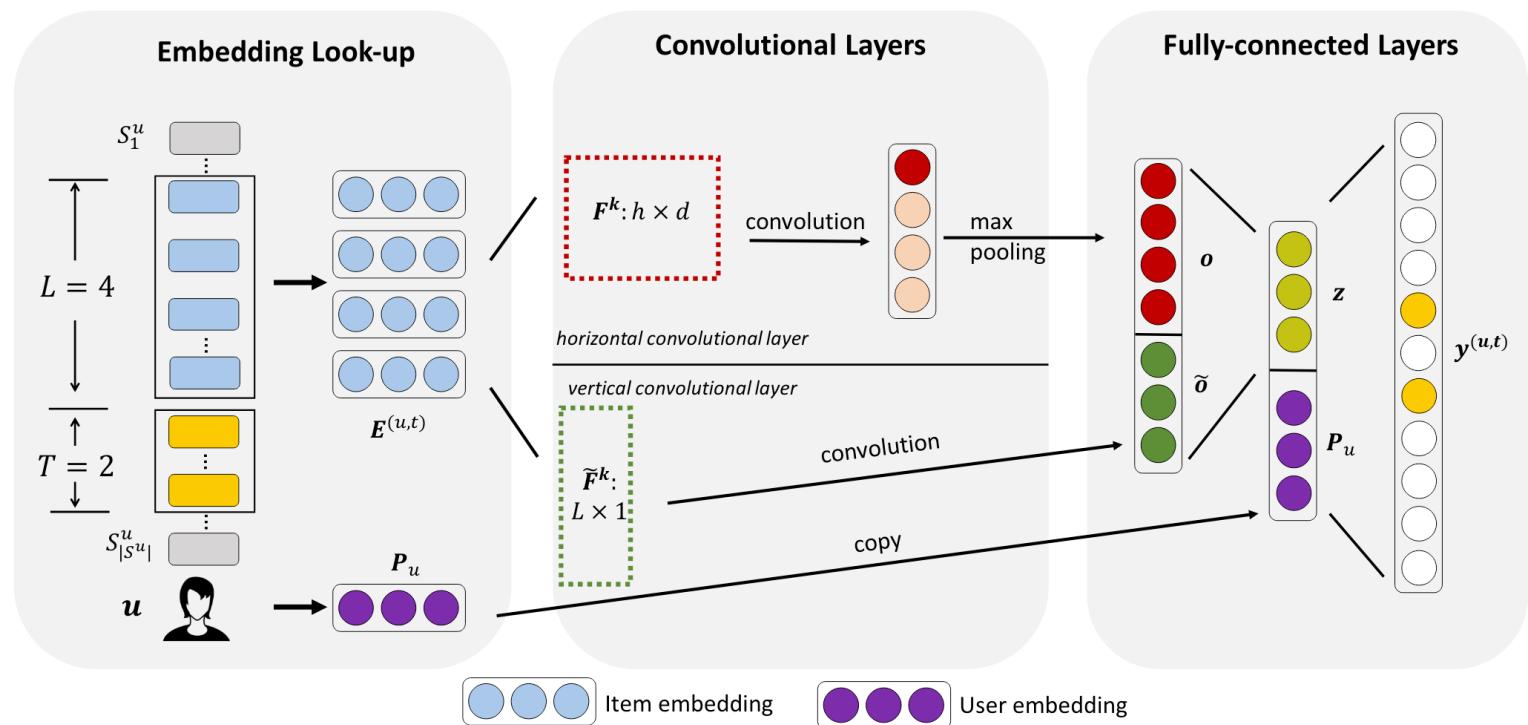
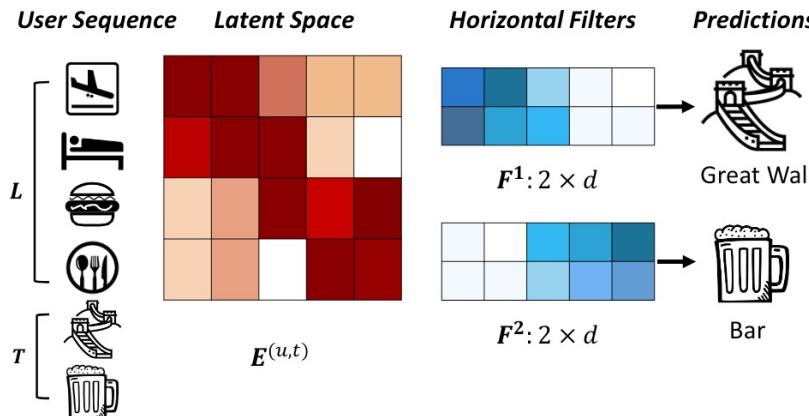
□ NeuMF unifies the strengths of MF and MLP in modelling user-item interactions

- ❖ MF uses an inner product as the interaction function
- ❖ MLP is more sufficient to capture the complex structure of user interaction data



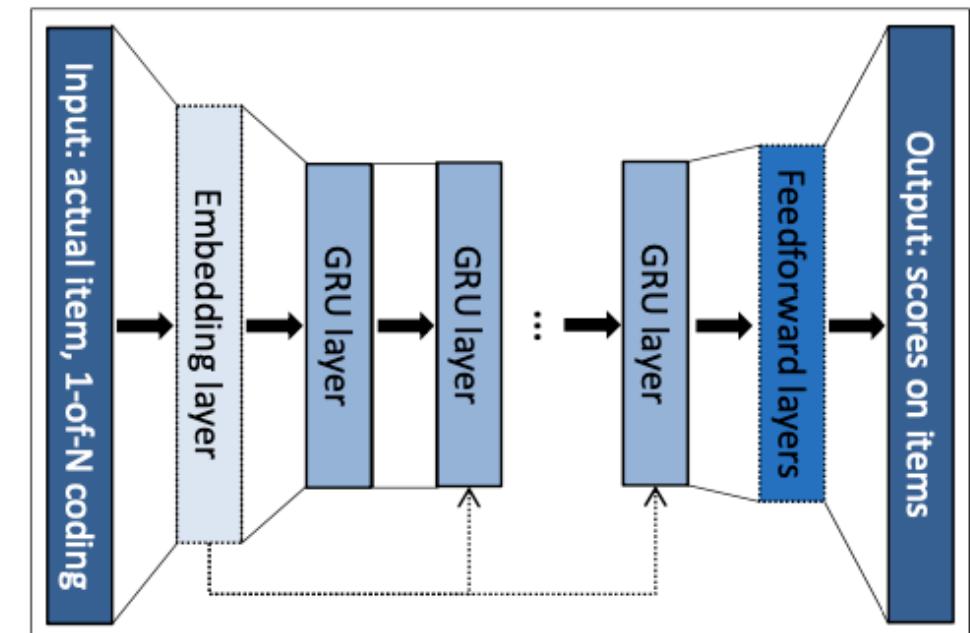
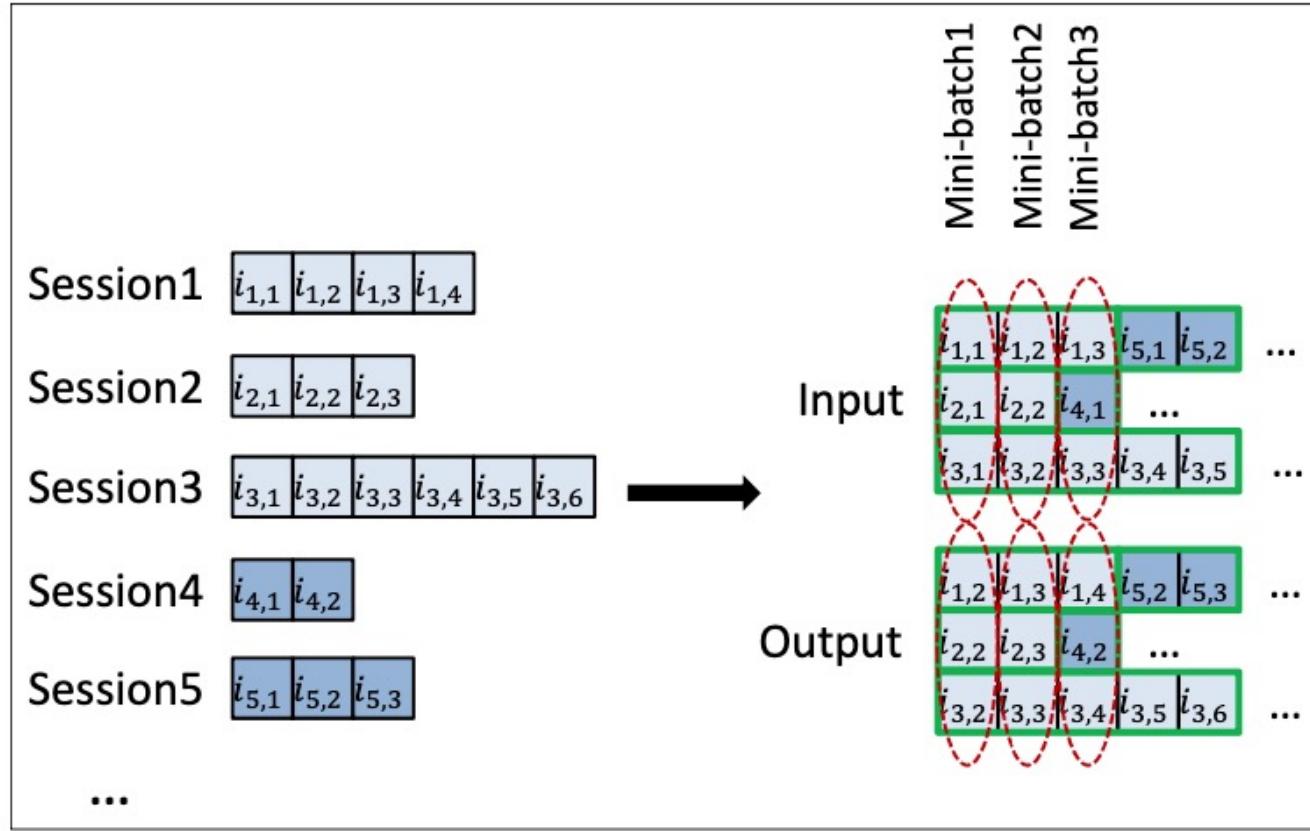
# Caser

- **Top-N sequential recommendation** models each user as a **sequence of items** interacted in the past and aims to **predict top-N ranked items**
- Convolutional Sequence Embedding Recommendation Model



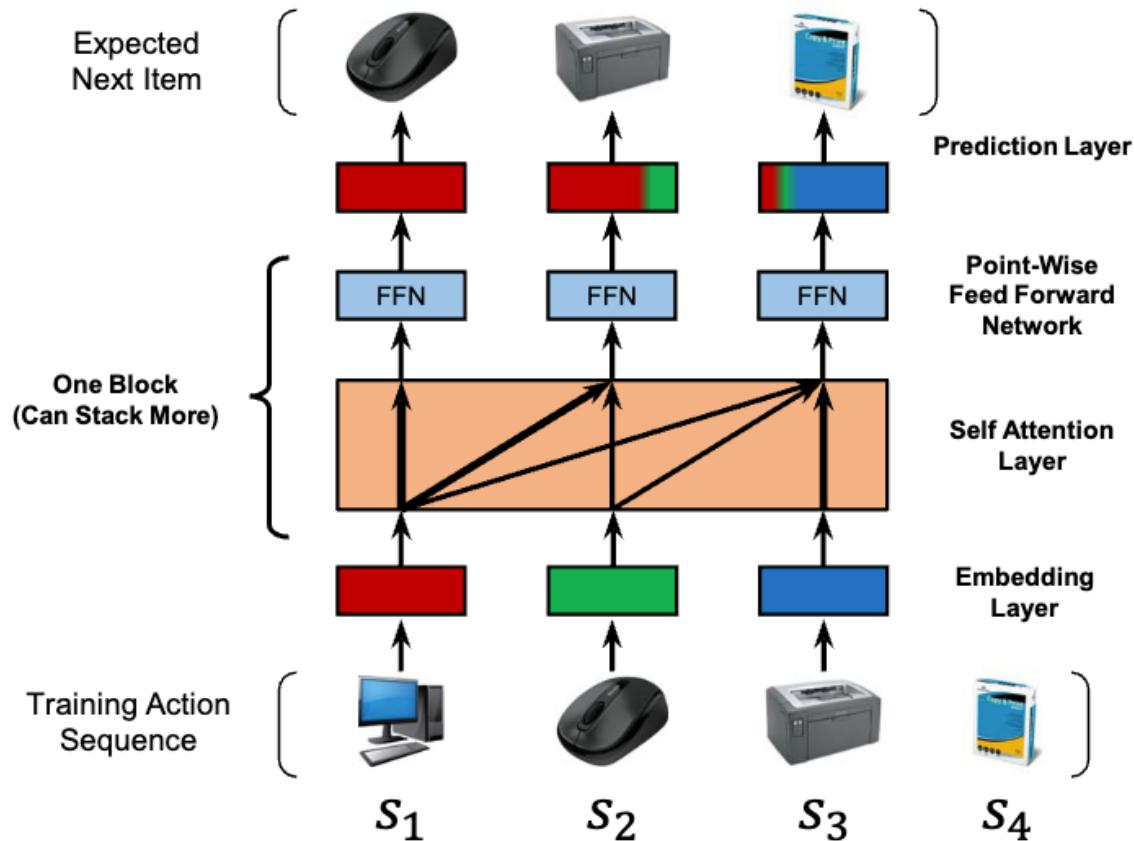
# GRU4Rec

- **Session-based** Recommendations with Recurrent Neural Networks (**RNN**)
- Introducing **session-parallel mini-batches**, mini-batch based output sampling and ranking loss function



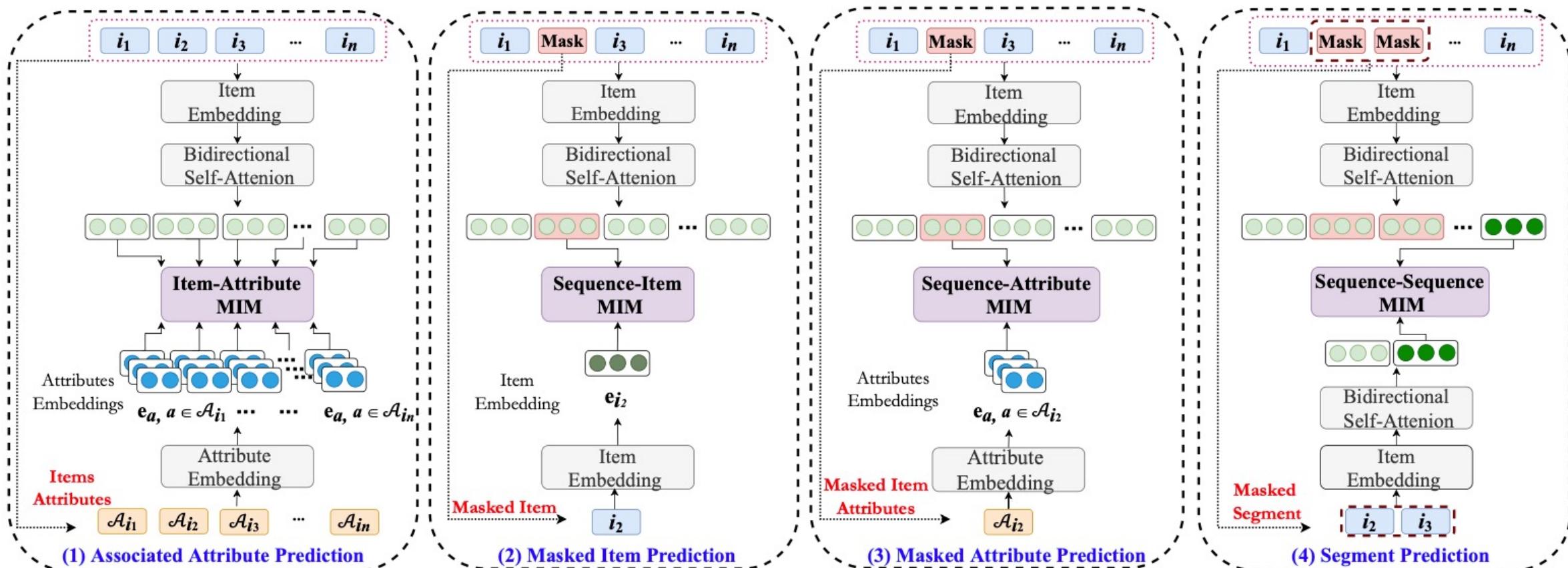
# SASRec

- Self-Attentive Sequential Recommendation
- Using an **attention** mechanism to capture **long-term semantics** and makes its **predictions** based on relatively **few actions**



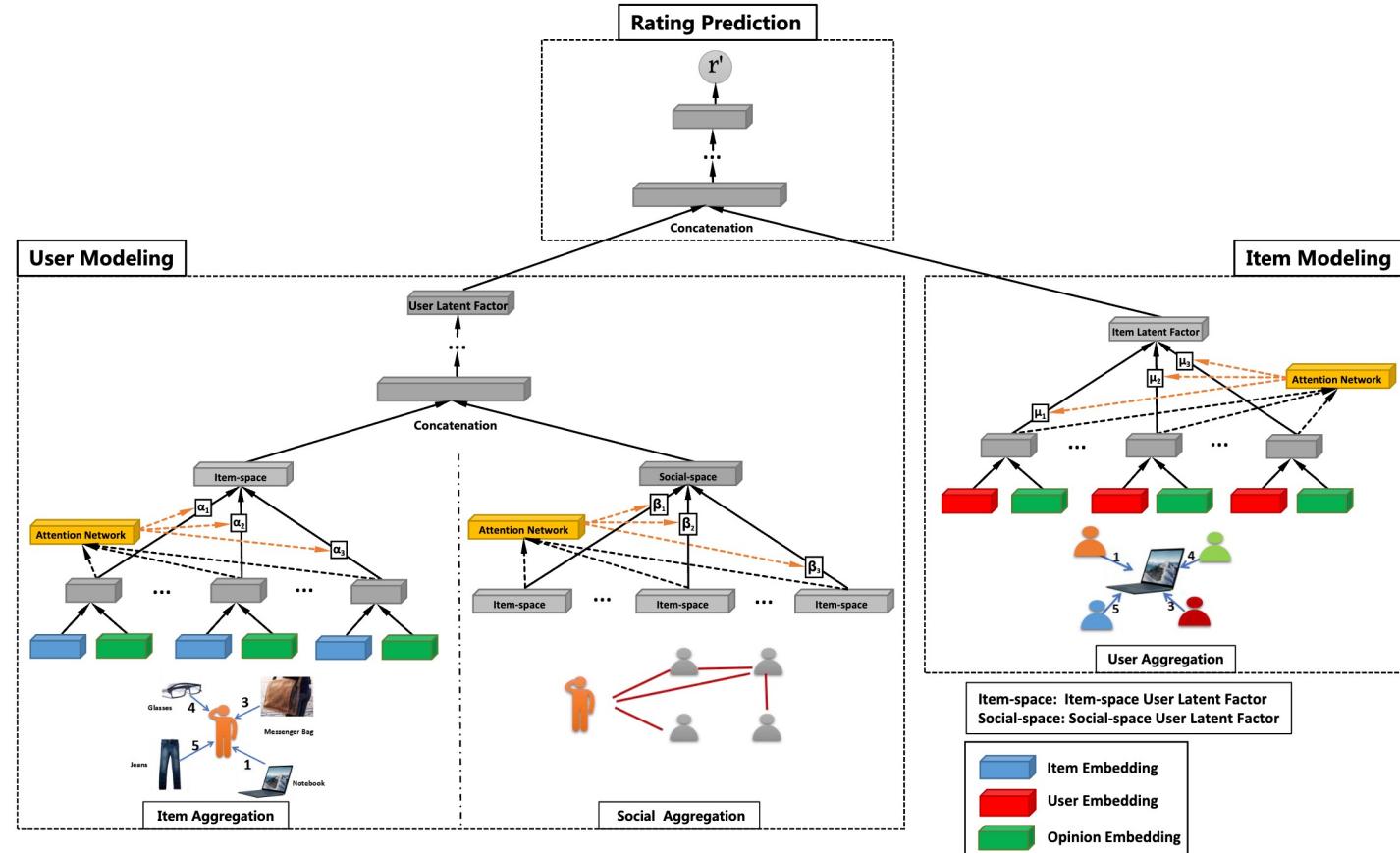
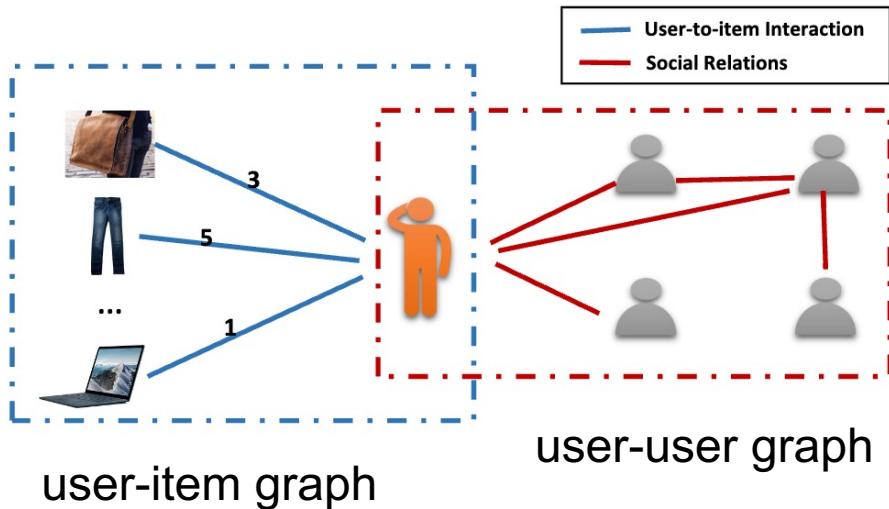
# S3-Rec

- Utilizing the intrinsic data correlation to derive **self-supervision** signals
- Enhancing the data representations via **pre-training** methods



# GraphRec

- Data in social recommender systems can be represented as **user-user** social graph and **user-item** graph



Preliminaries

Pre-training

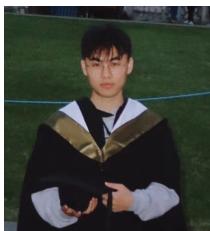
Fine-tuning

Prompting

Future  
Directions

# Overview

**Presenter:**  
Yunqing Liu

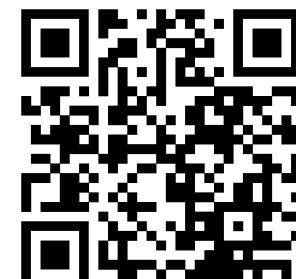


## RecSys

- Collaborative Filtering (CF)
- Content-based Recommendation
- Deep Recommender Systems

## LLMs

- Encoder-Only
- Decoder-Only
- Encoder-Decoder



Website QR Code

# LLMs to RecSys

- LLMs can be used for a variety of tasks, such as **Image Generation**

DALL·E mini

Generate images from text

What do you want to see?

an avocado armchair flying into space

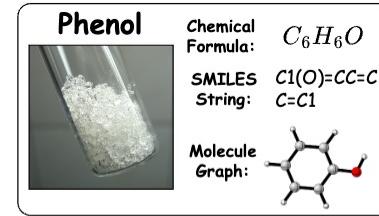
an avocado armchair flying into space



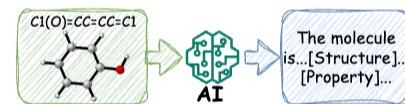
Text to Image

# LLMs to RecSys

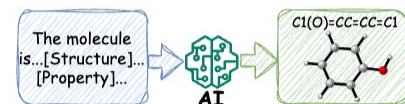
- LLMs can be used for a variety of tasks, such as **Molecule Generation**



(a) Molecule Representations.



(b) Molecule Captioning.



(c) Text-based Molecule Generation.

**ChatGPT**

(a) Molecule Captioning

Please show me a description of this molecule:  
"C1=CC=C(C=C1)OC2=CC=CC=C2"

The molecule is an aromatic ether in which the oxygen is attached to two phenyl substituents. It has been found in muscat grapes and vanilla. It has a role as a plant metabolite.

(b) Text-based Molecule Generation

Help me generate a molecule based on the given description:

The molecule is a quinolinemonocarboxylate that is the conjugate base of xanthurenic acid, obtained by deprotonation of the carboxy group. It has a role as an animal metabolite. It is a conjugate base of a xanthurenic acid.

C1=CC2=C(C(=C1)[O-])NC(=CC2=O)C(=O)O

(d) Empowering ChatGPT with molecule captioning and text-based molecule generation abilities.

Text to Molecule

# LLMs to RecSys

- LLMs can be used for a variety of tasks, such as **Recommendation**

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®, 5.0 ; 4.Nail Art Jumbo Stamp Stamping Manicure Image Plate 6 Happy Holidays by Cheeky®, 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 and 5 being highest,The output should be like: (x stars, xx%), do not explain the reason.)
Sequential Recommendation	
zero-shot	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.
few-shot	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.

## Text to Recommendation

# What are Language Models?

## ❑ Narrow Sense

- ❖ A **probabilistic model** that assigns a probability to every **finite sequence** (grammatical or not)

Sentence: “the cat sat on the mat”

$$\begin{aligned}
 P(\text{the cat sat on the mat}) = & P(\text{the}) * P(\text{cat}|\text{the}) * P(\text{sat}|\text{the cat}) \\
 & * P(\text{on}|\text{the cat sat}) * P(\text{the}|\text{the cat sat on}) \\
 & * P(\text{mat}|\text{the cat sat on the})
 \end{aligned}$$

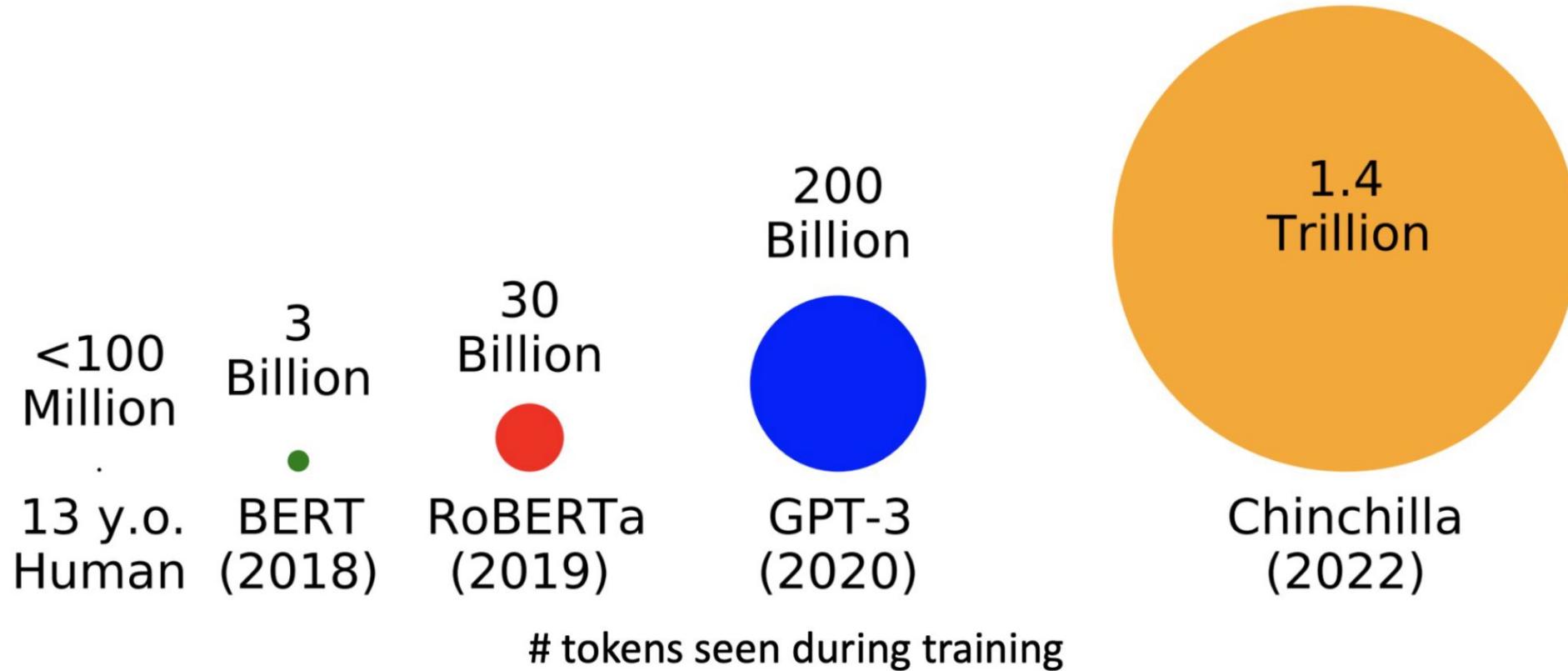
Implicit order

## ❑ Broad Sense

- ❖ Encoder-only models (BERT, RoBERTa, ELECTRA)
- ❖ Decoder-only models (GPT-X, OPT, LLaMa, PaLM)
- ❖ Encoder-decoder models (T5, BART)

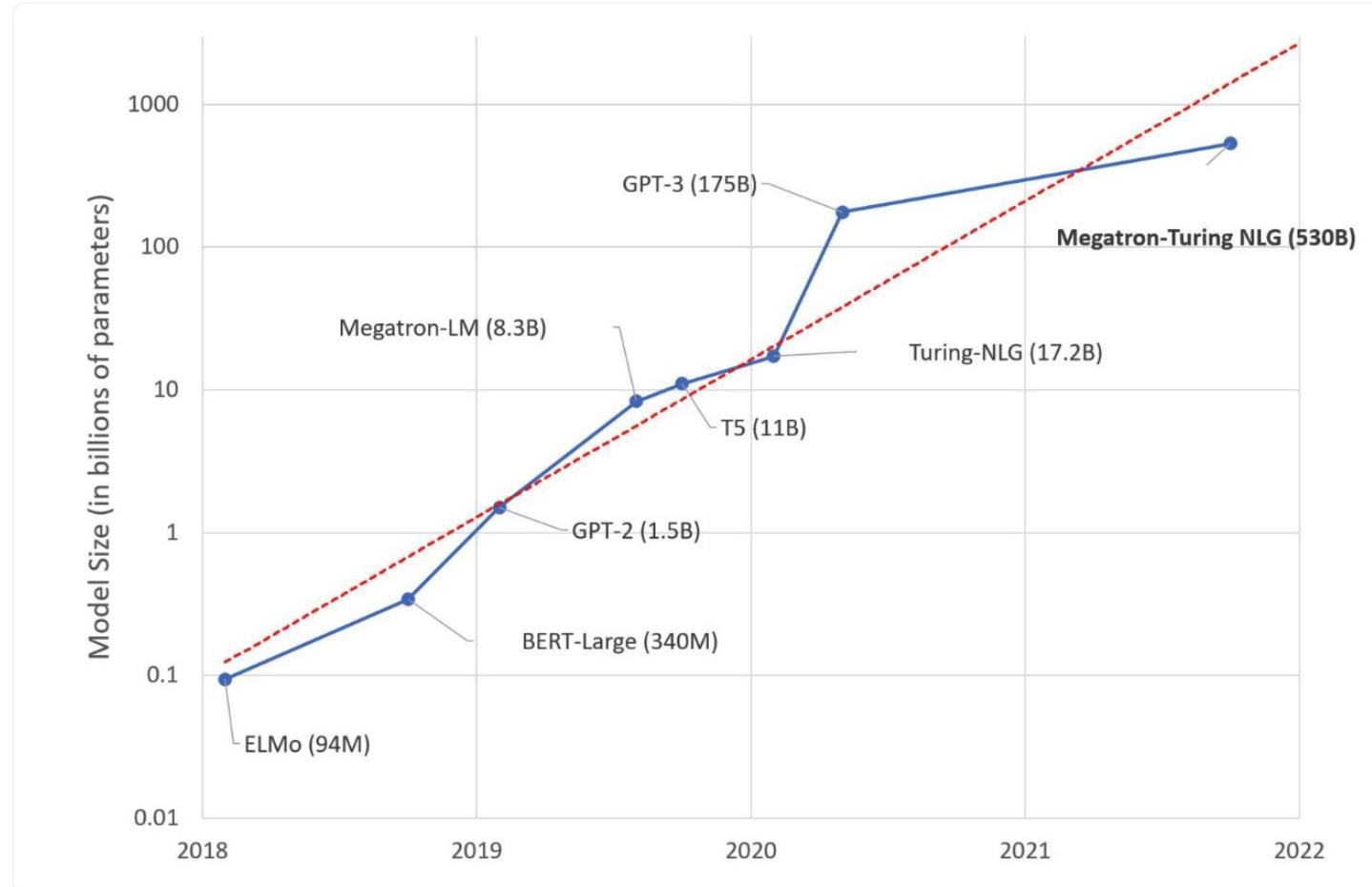
# Large Language Models

- Trained on more and more data – **Hundreds of Billions of Tokens**



# Large Language Models

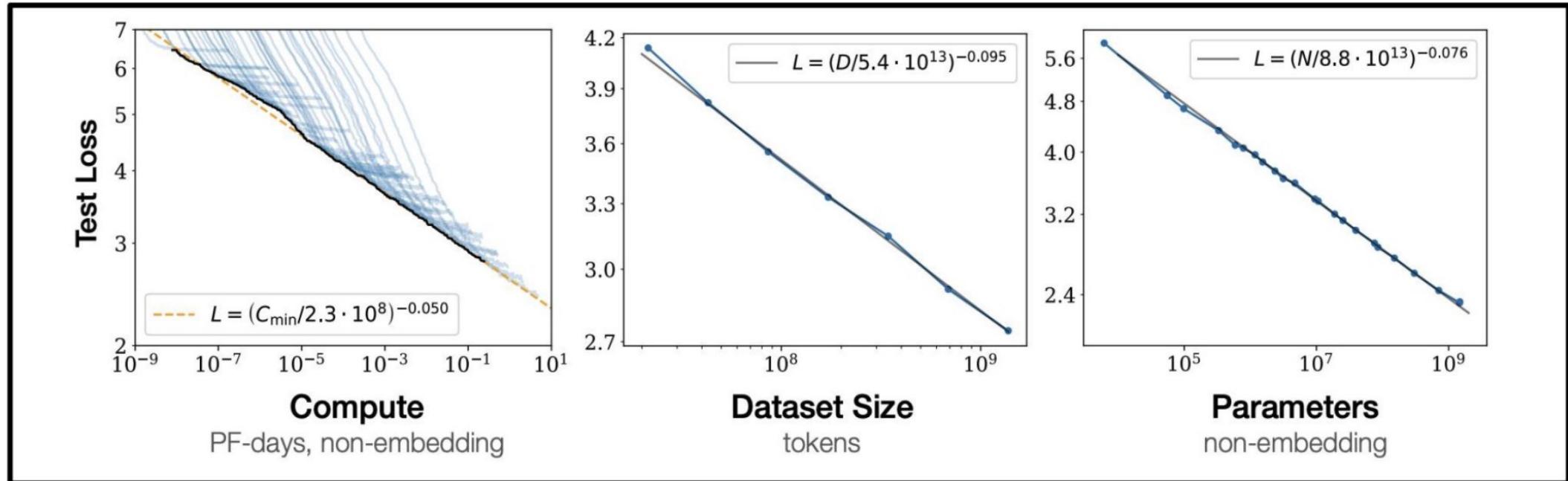
## ☐ Larger and larger models – **Billions of Parameters**



# Why Large Language Models

## □ Scaling Law for Neural Language Models

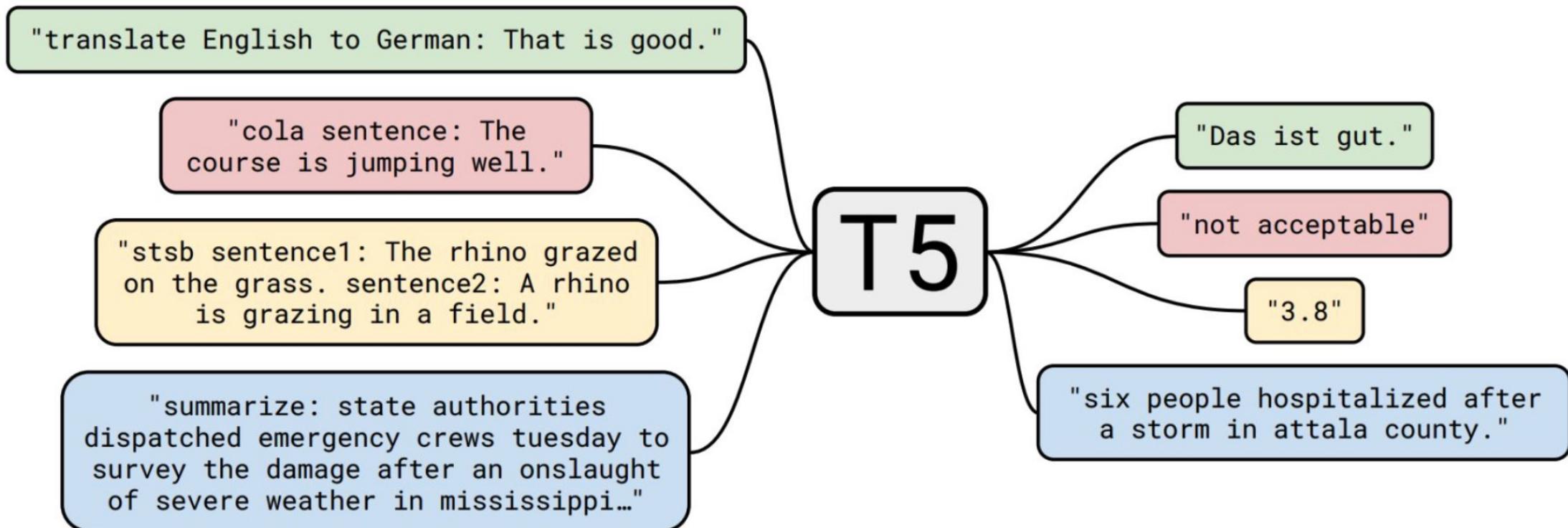
- ❖ Performance depends strongly on scale! We keep getting better performance as we scale the model, data, and compute up!



# Why Large Language Models?

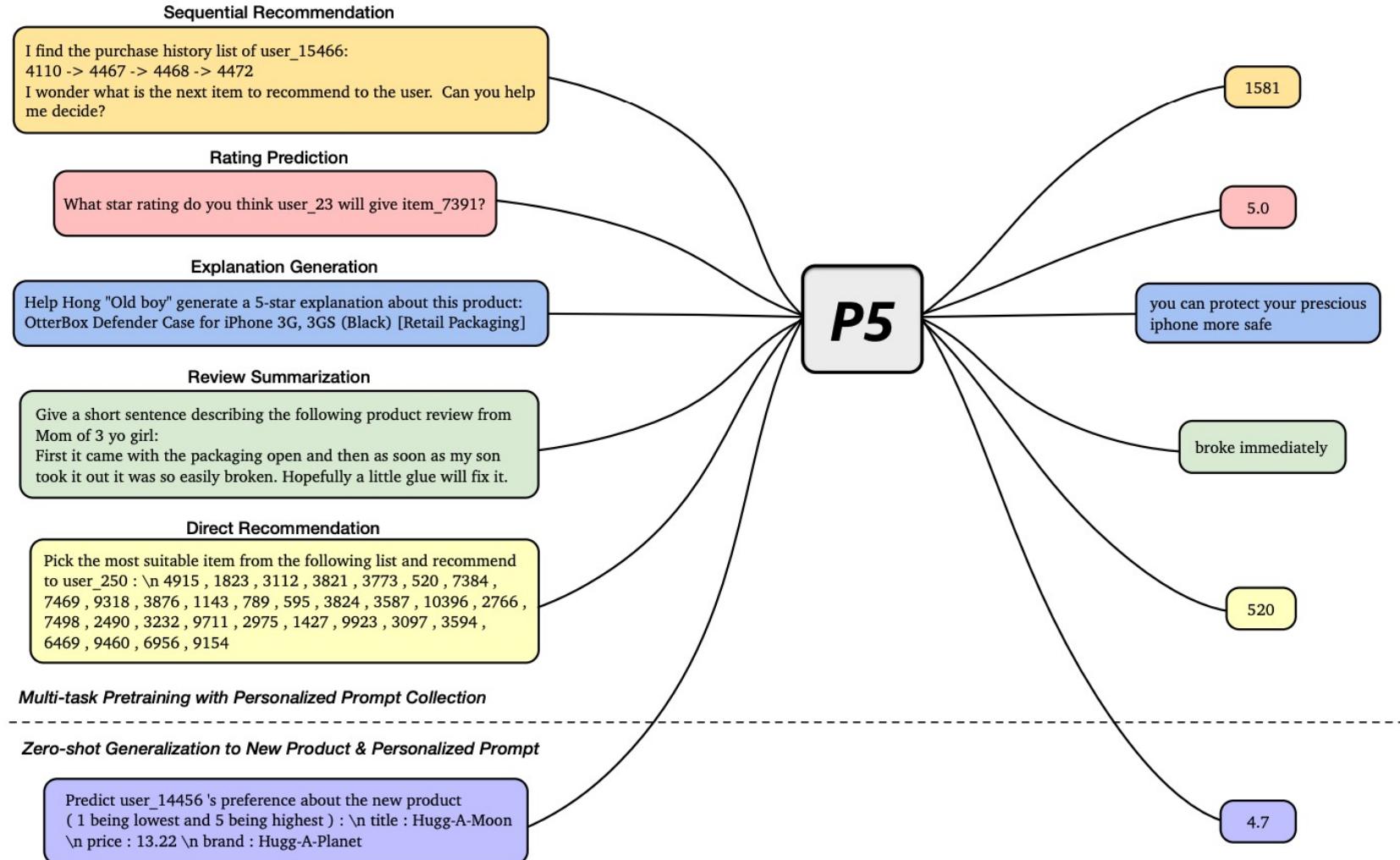
## □ Generalization

- ❖ We can now use one single model to solve many NLP tasks.



# Why Large Language Models?

## □ The Strong Zero-shot/Few-shot Ability



# Large Language Models

## □ Encoder-Only Models

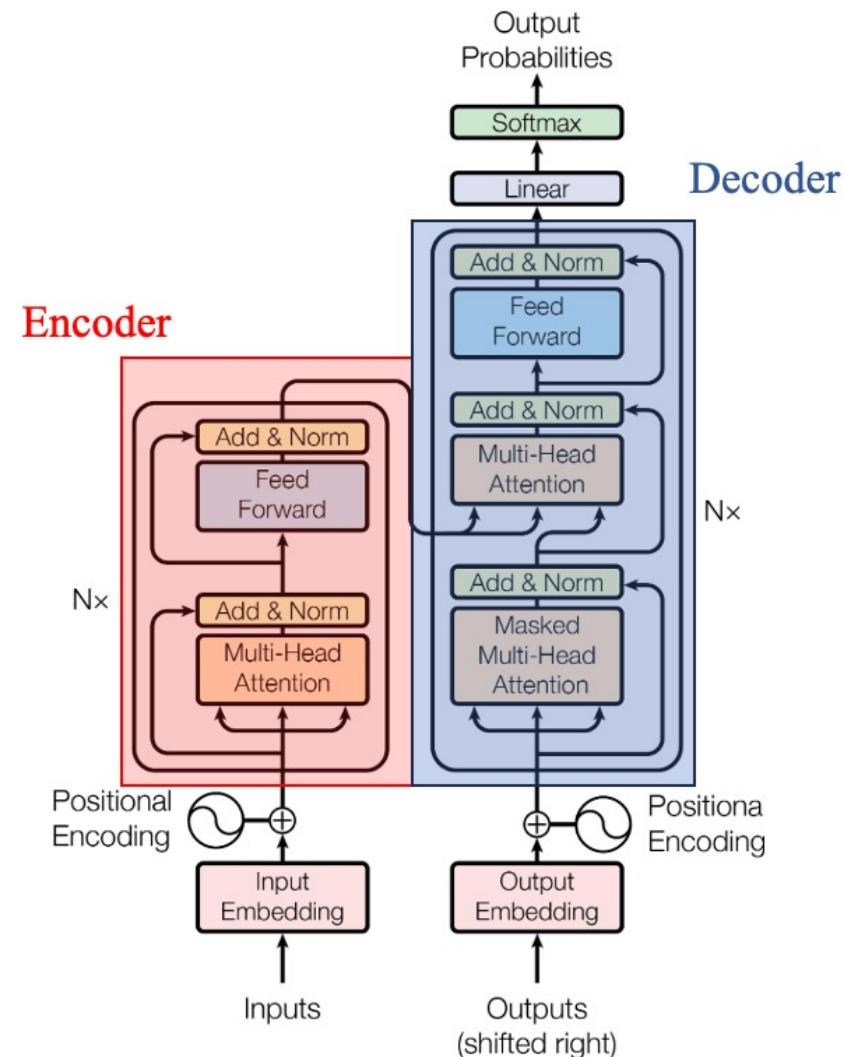
- ❖ BERT, RoBERTa, ELECTRA

## □ Decoder-Only Models

- ❖ GPT-X, OPT, LLaMa, PaLM

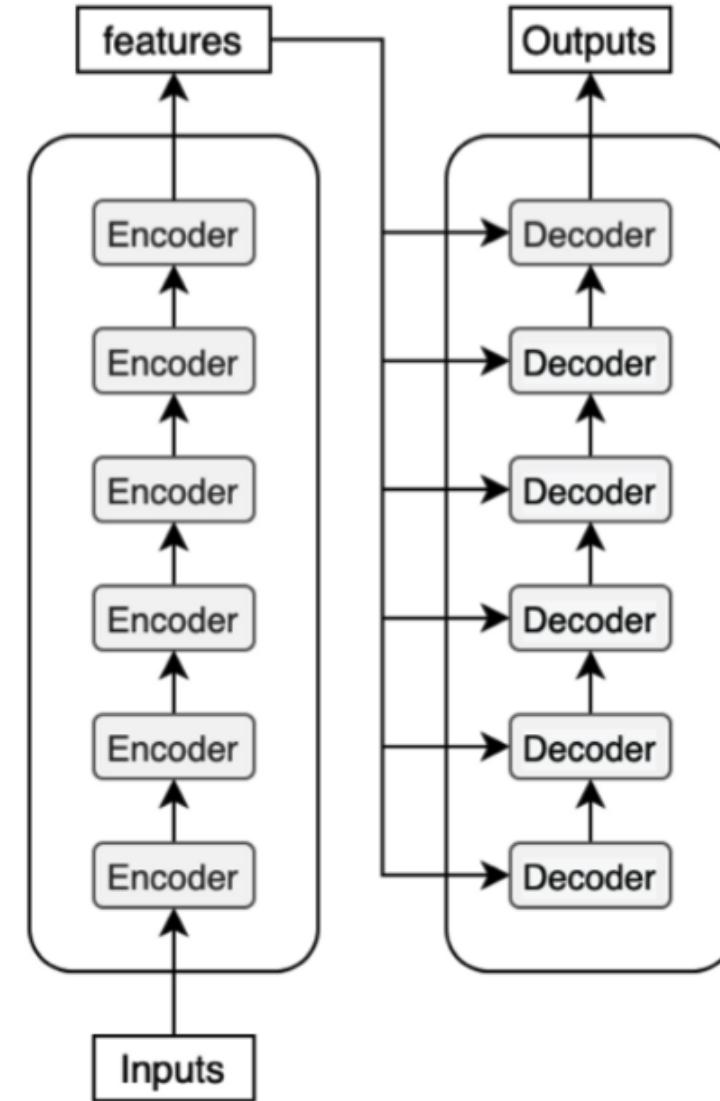
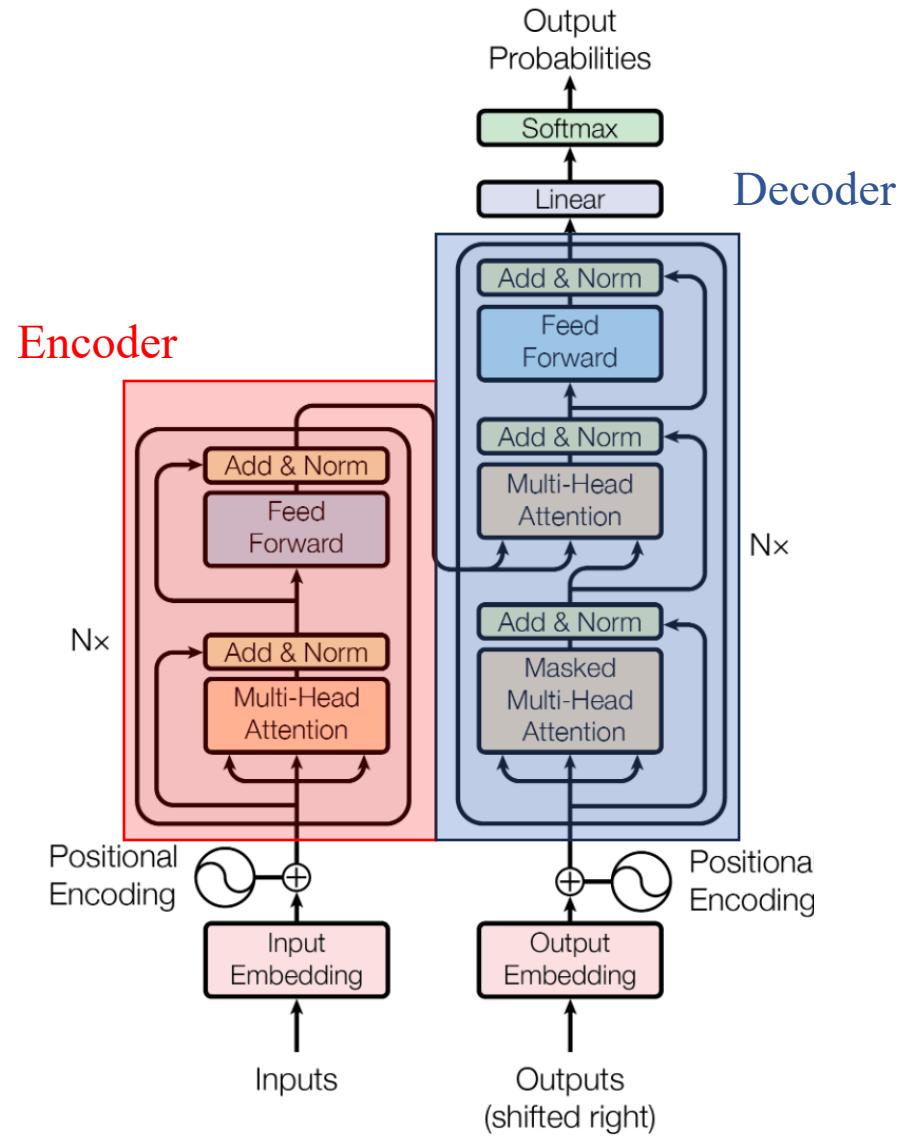
## □ Encoder-Decoder Models

- ❖ T5, BART

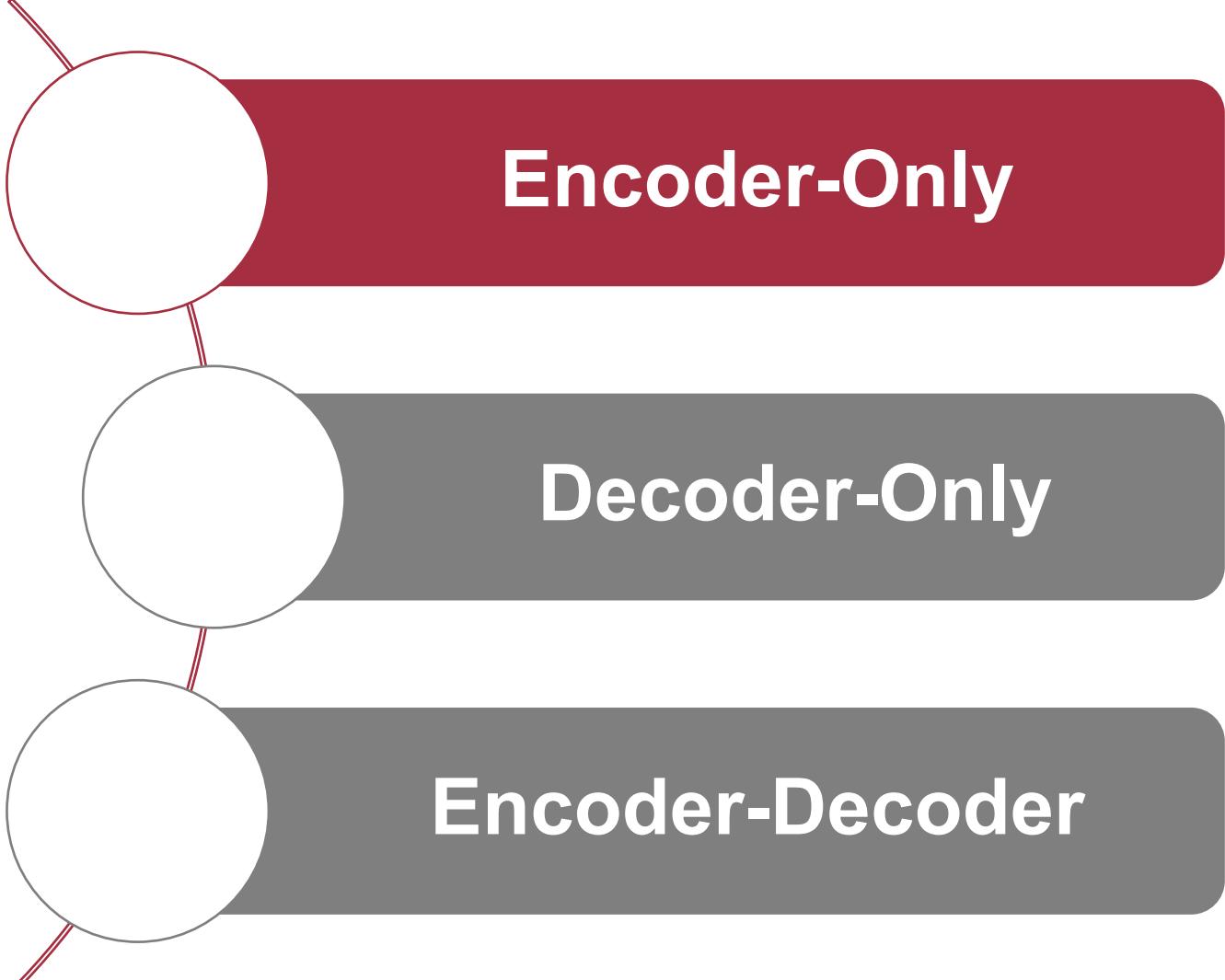


The Transformer – model architecture

# Transformer



# Large Language Models



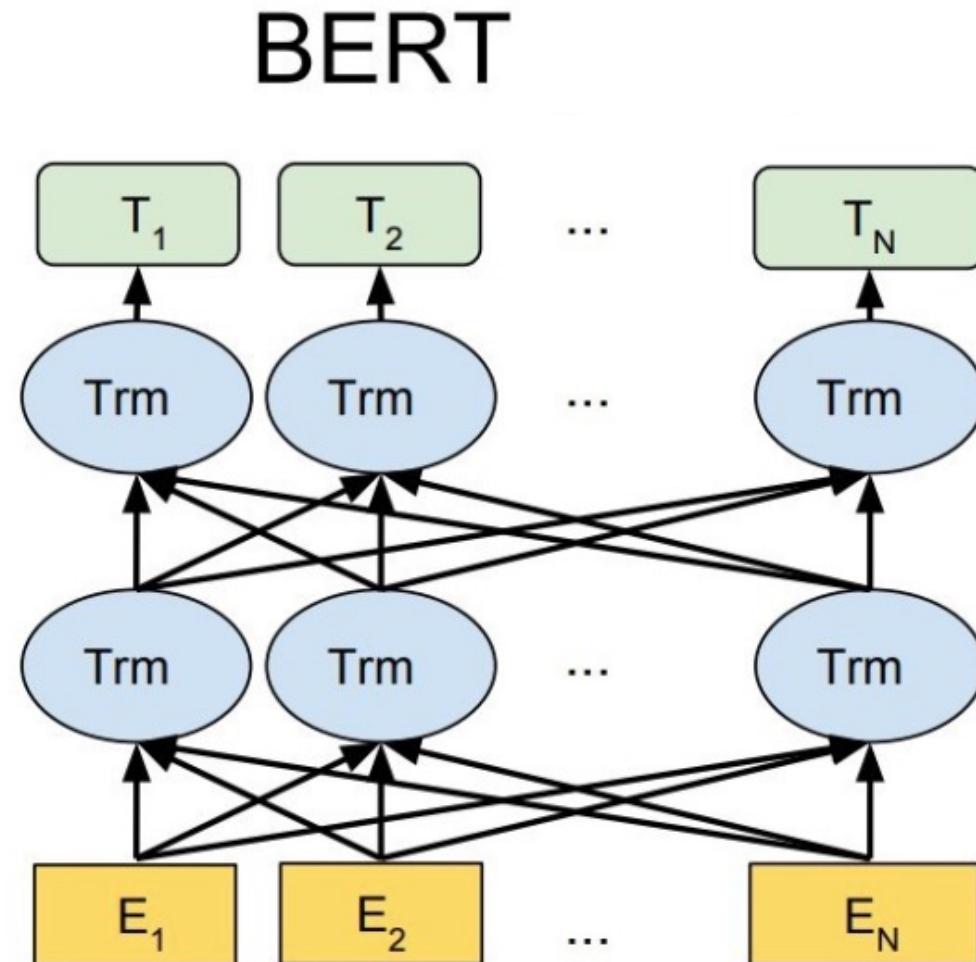
Encoder-Only

Decoder-Only

Encoder-Decoder

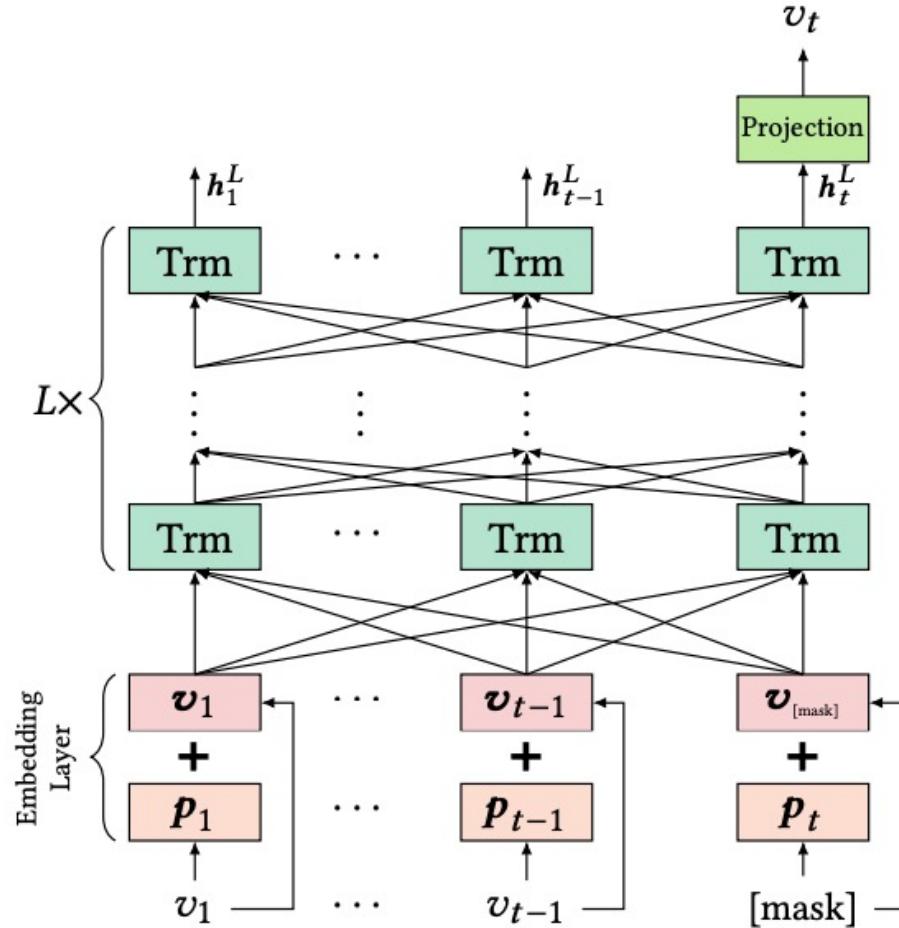
# BERT

- BERT uses a bidirectional Transformer

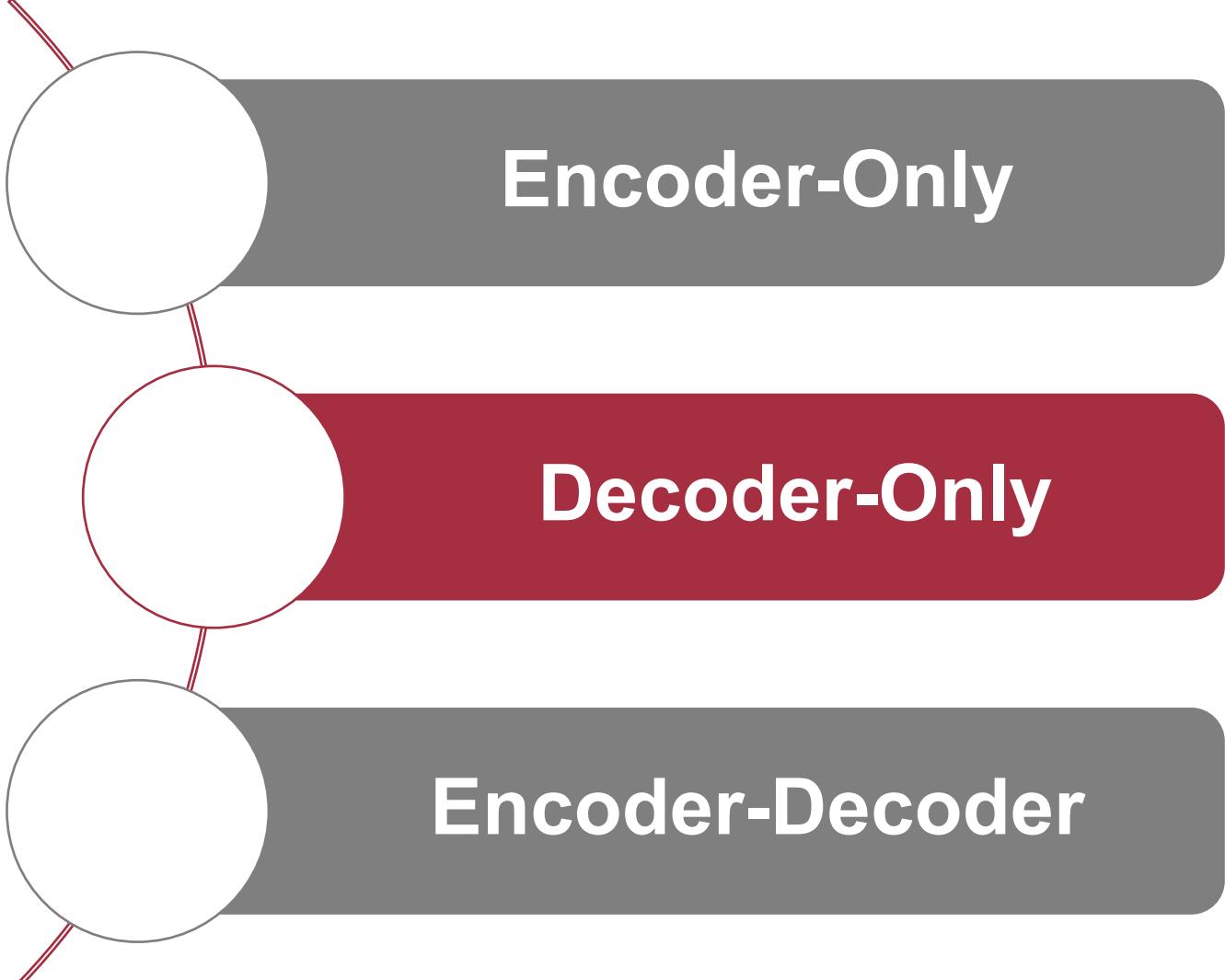


# BERT4Rec

- Adopt **Bidirectional Encoder Representations** from Transformers to model the sequential nature of user behaviors



# Large Language Models



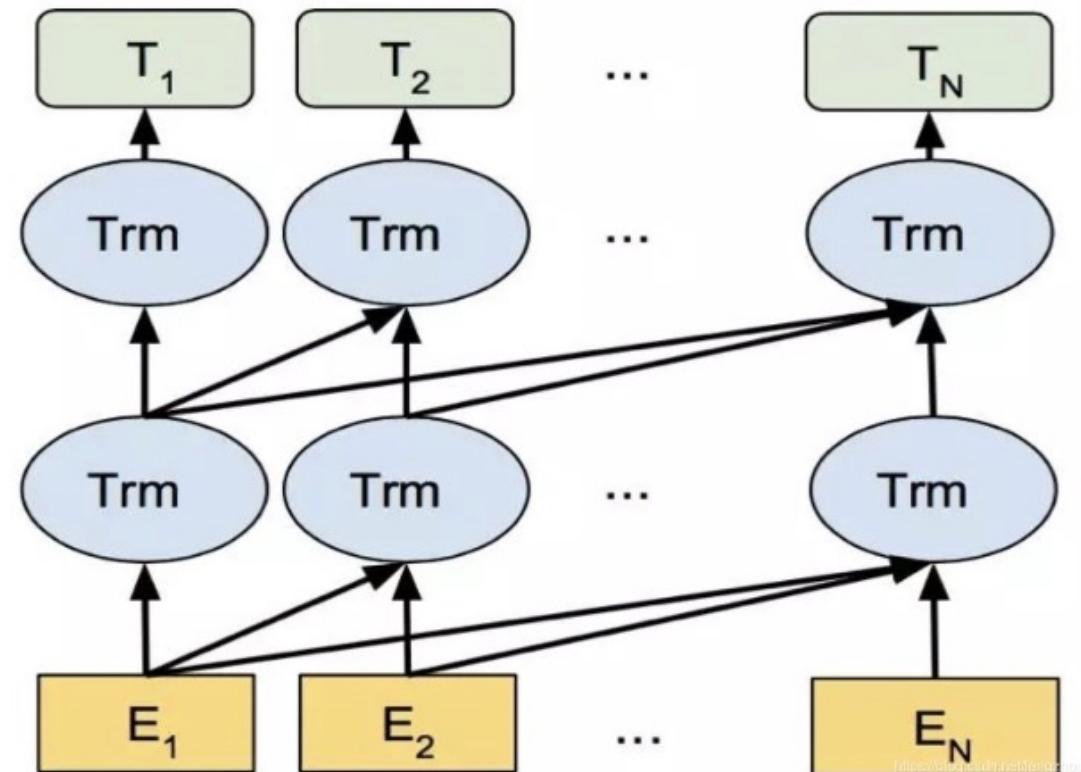
Encoder-Only

Decoder-Only

Encoder-Decoder

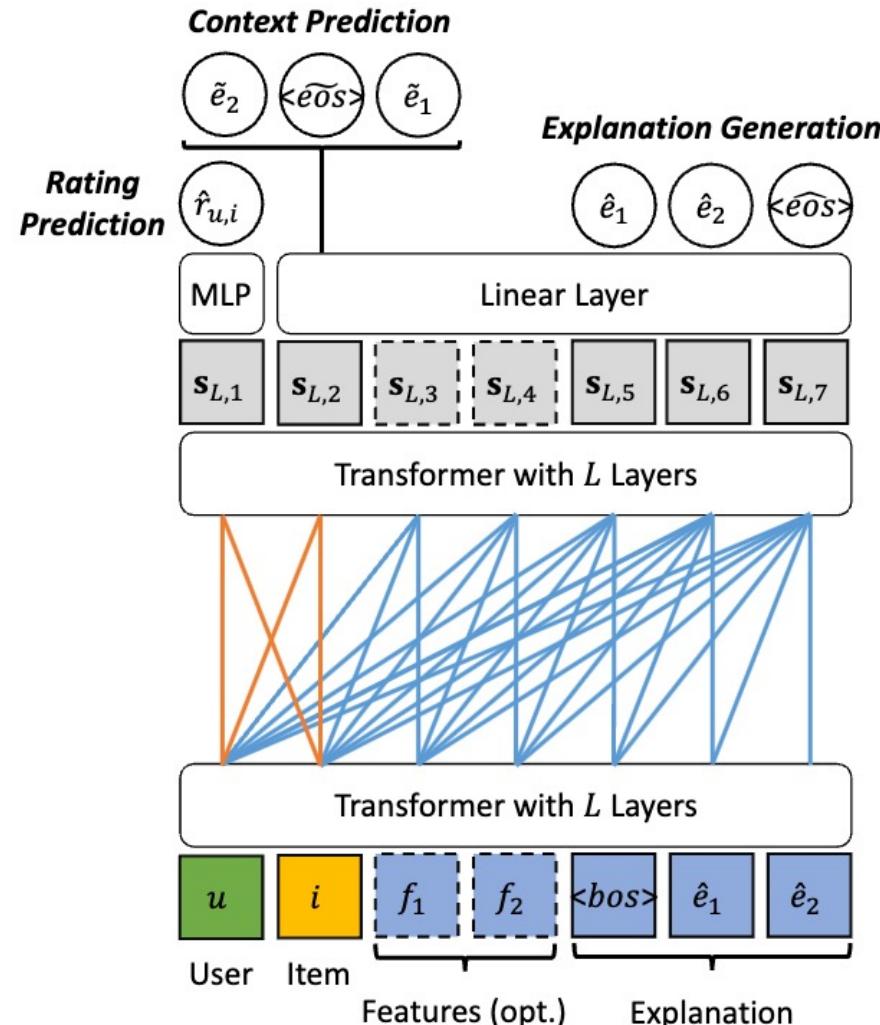
- OpenAI GPT uses a left-to-right Transformer

## OpenAI GPT

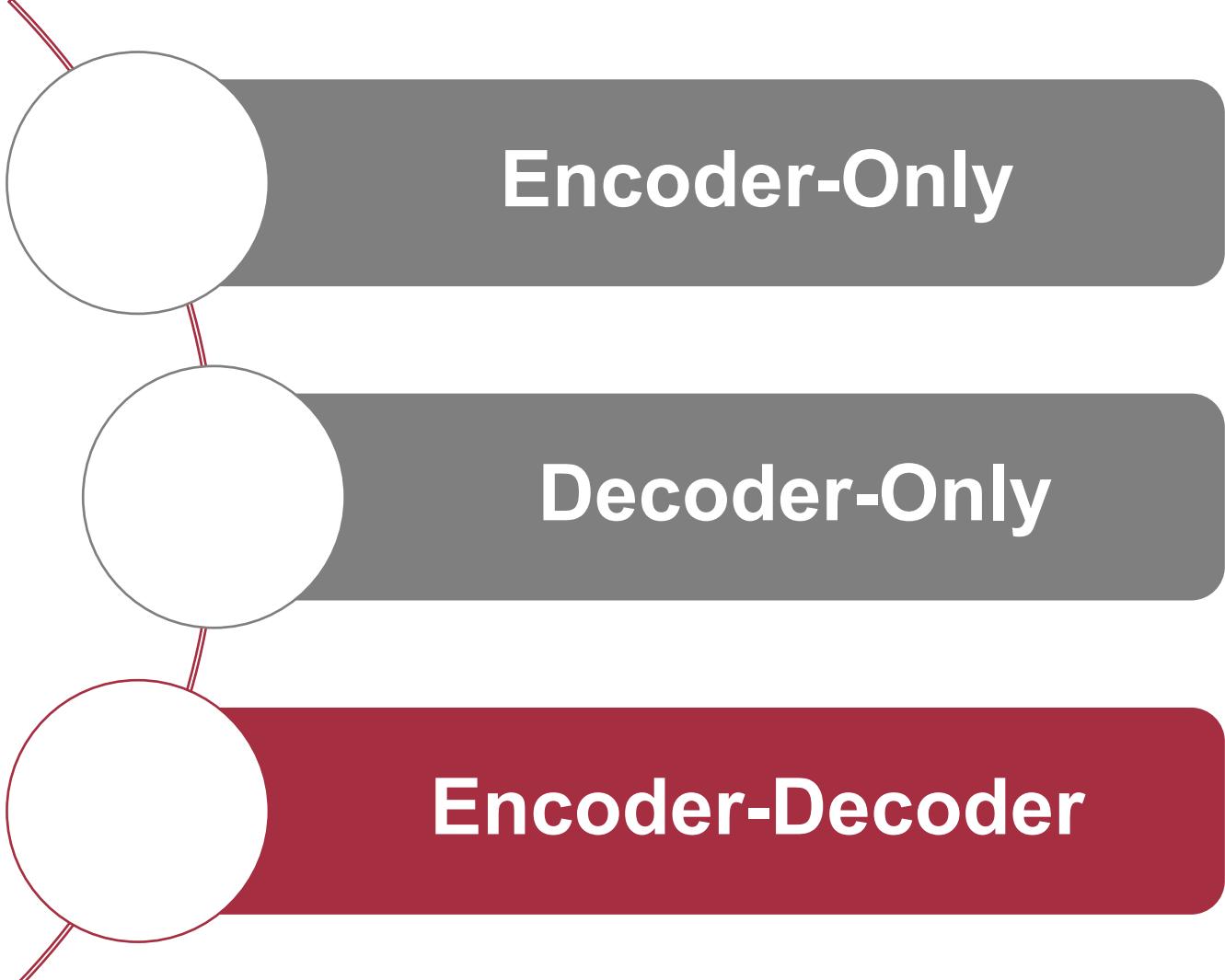


[https://dqlab.id/pythonBasis...](https://dqlab.id/pythonBasics/10/)

- Utilizing the IDs to predict the words in the target explanation



# Large Language Models

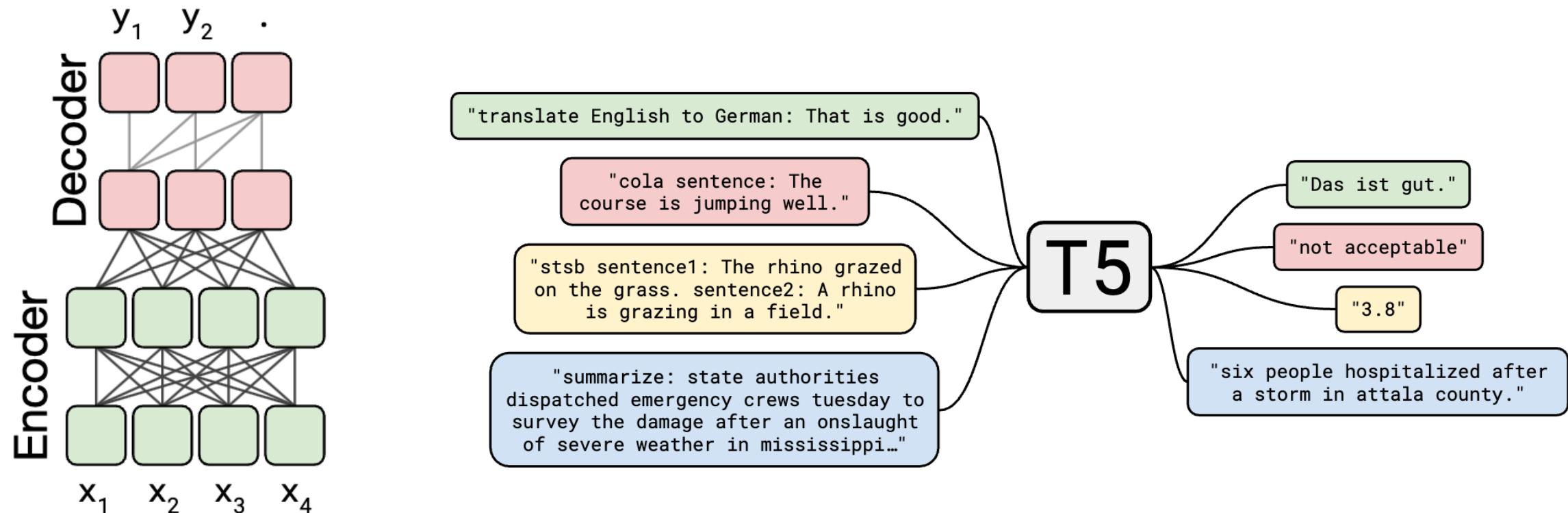


Encoder-Only

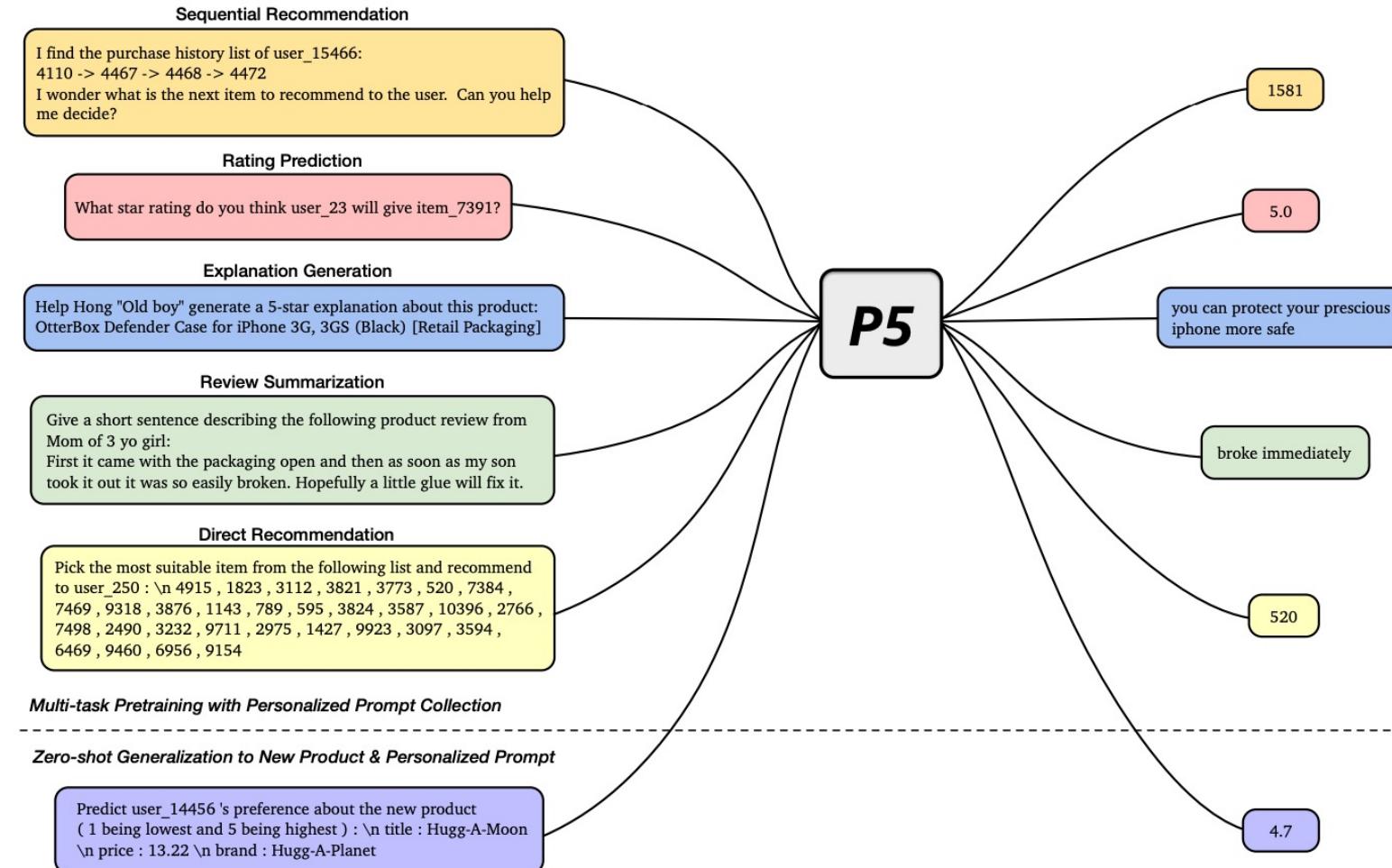
Decoder-Only

Encoder-Decoder

- T5 handles any text-to-text task by converting every natural language processing problem into a text generation problem.



□ Text-to-text paradigm - “Pretrain, Personalized Prompt, and Predict Paradigm” (P5) for recommendation



Preliminaries

Pre-training

Fine-tuning

Prompting

Future  
Directions

# Overview

**Presenter:**  
Jiatong Li

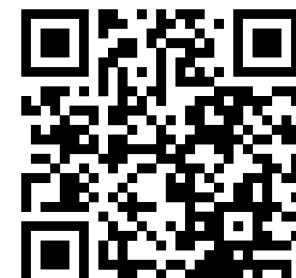


## User & Item Representation

- ID-based LLM RecSys
- Text-based LLM RecSys

## Pre-training

- Pre-training in NLP
- Pre-training LLM-based RecSys



Website QR Code

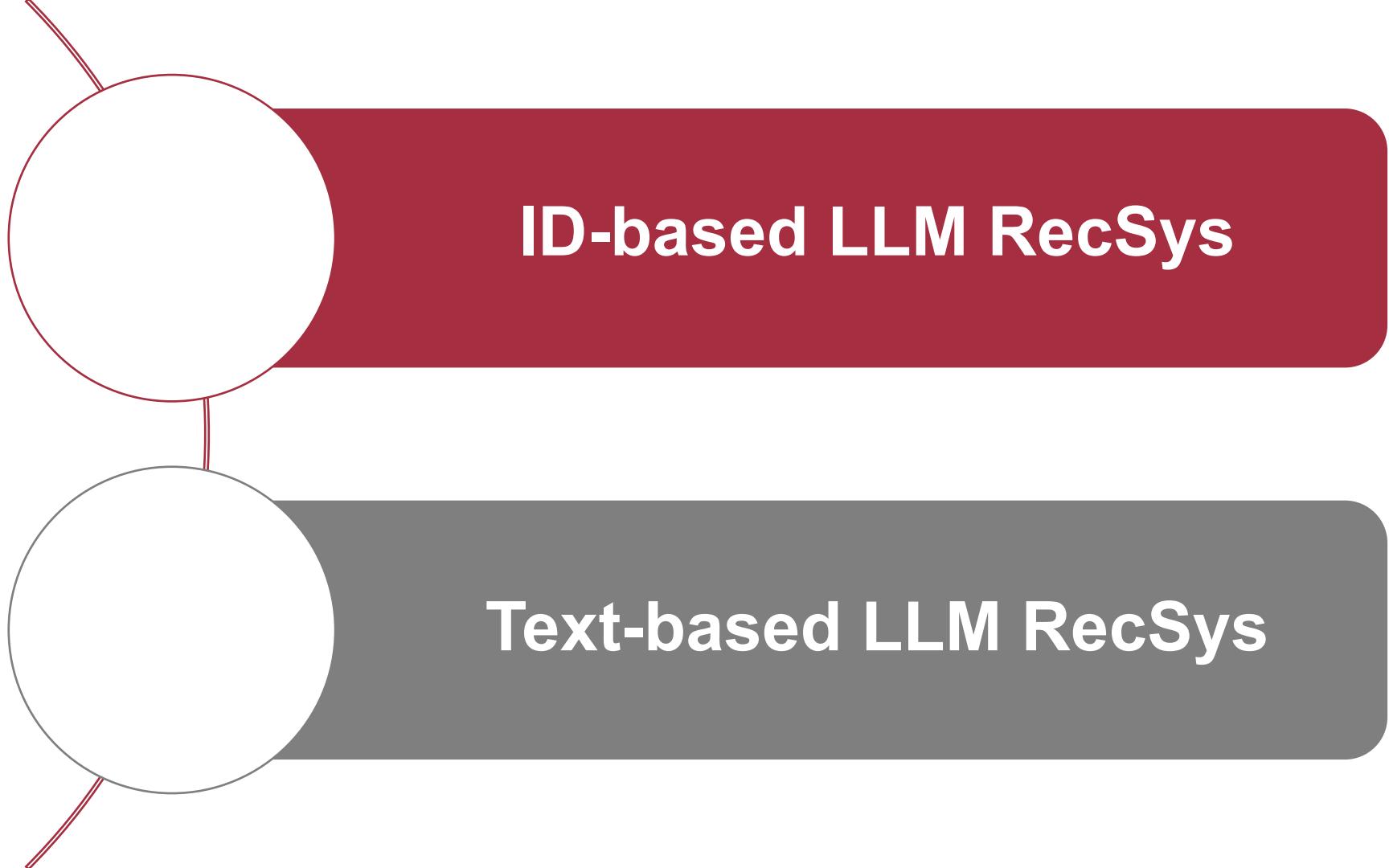
# User & Item Representation

- Users and Items can be represented in various ways



Poster	Movie Name	Numeric ID
	In Broad Daylight	1697292155
	The Marvels	1699436461
	TAYLOR SWIFT   THE ERAS TOUR	1695730583
	The Dark Knight Rises	1699611567
	Oppenheimer	1687513232

# User & Item Representation in LLMs

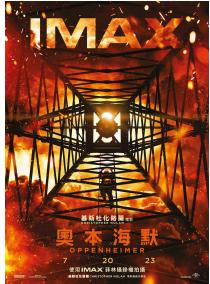


**ID-based LLM RecSys**

**Text-based LLM RecSys**

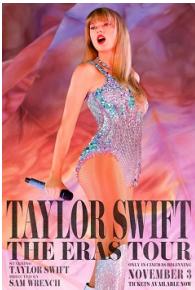
# ID-based LLM RecSys

- Various ways of assigning IDs



**Randomly**

AXGGWD027



XJSGDG0881



BXGW2UD803

**Based on Popularity**

01

02

03

**Based on Time**

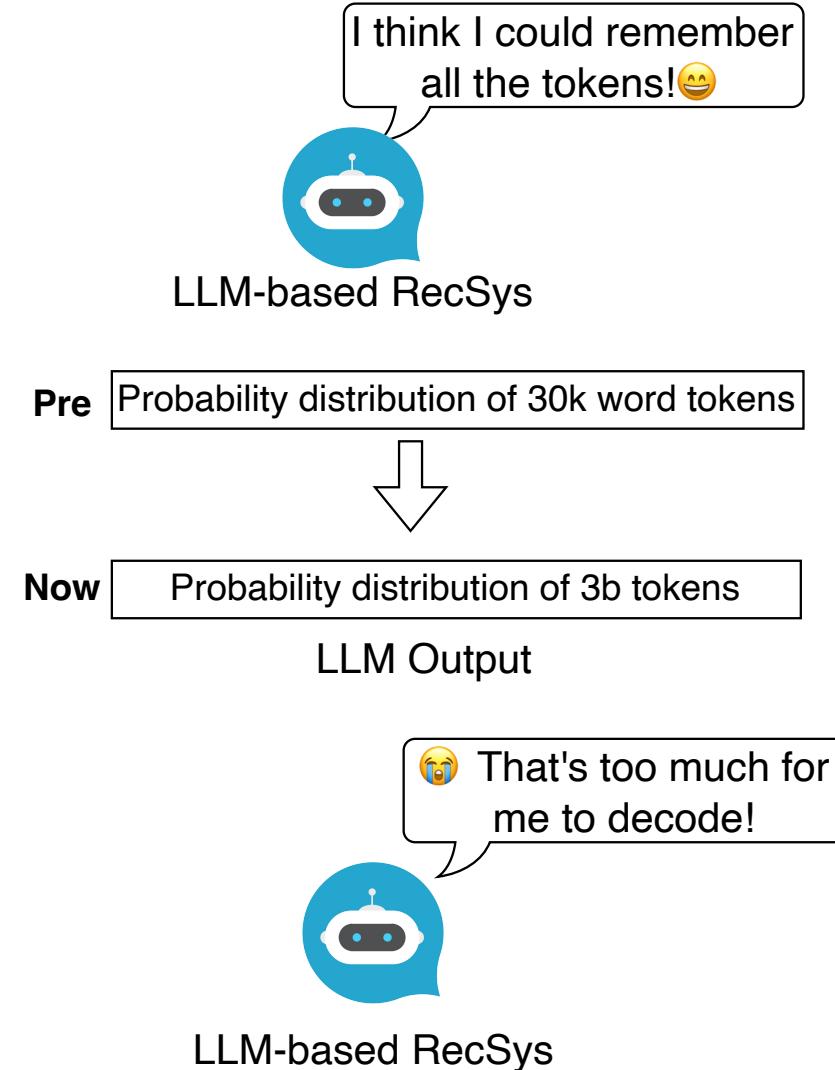
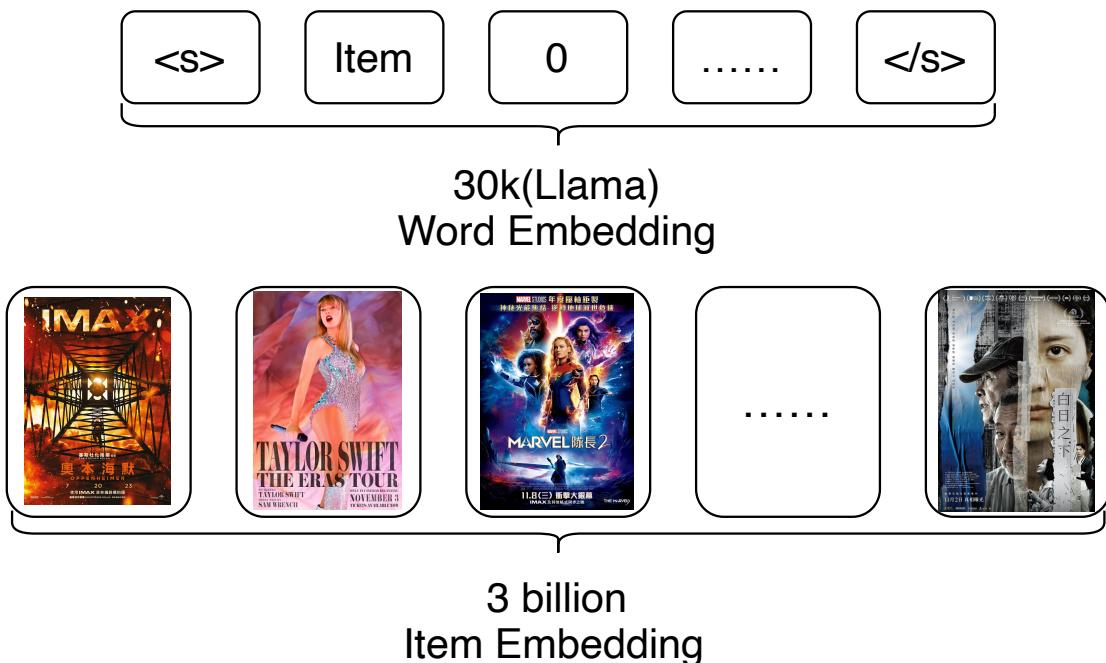
1687513232

1695730583

1699436461

# ID-based LLM RecSys

- IDs are originally for unique identification
- However, the embedding of LLMs can not hold millions of items and users



# ID-based LLM RecSys

- Normally, we can represent users and items with a span of tokens.
- The format is like “[Prefix]\_[ID]”. Examples:
  - ❖ User\_0123 : [“User”, “\_”, “0”, “1”, “2”, “3”]
  - ❖ Item\_5471 : [“Item”, “\_”, “5”, “4”, “7”, “1”]
  - ❖ However, for Item\_1003, it could be [“Item”, “\_”, “100”, “3”], which might be confusing for LLMs!



# ID-based LLM RecSys

- Indexing methods might affect the performance of RecSys



01

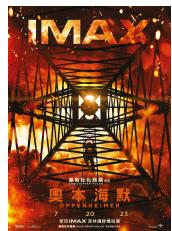
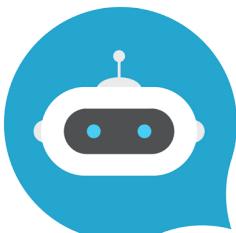


02



03

Is Item "04" still a movie?

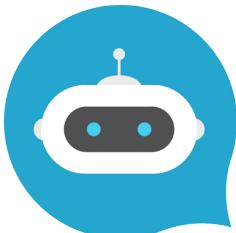


01



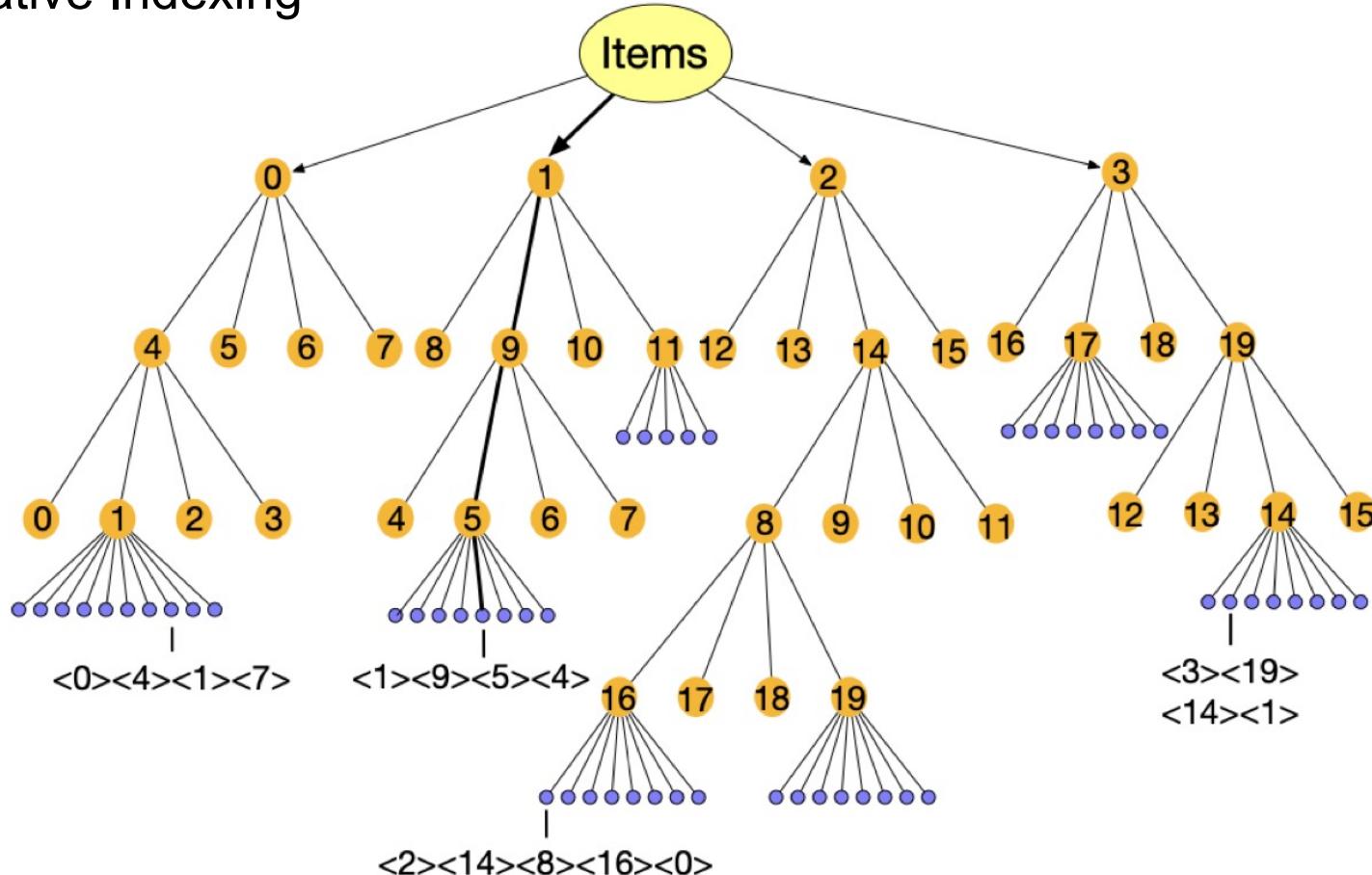
02

Do Item "01" and Item "02" share similar characteristics?



# ID-based LLM RecSys

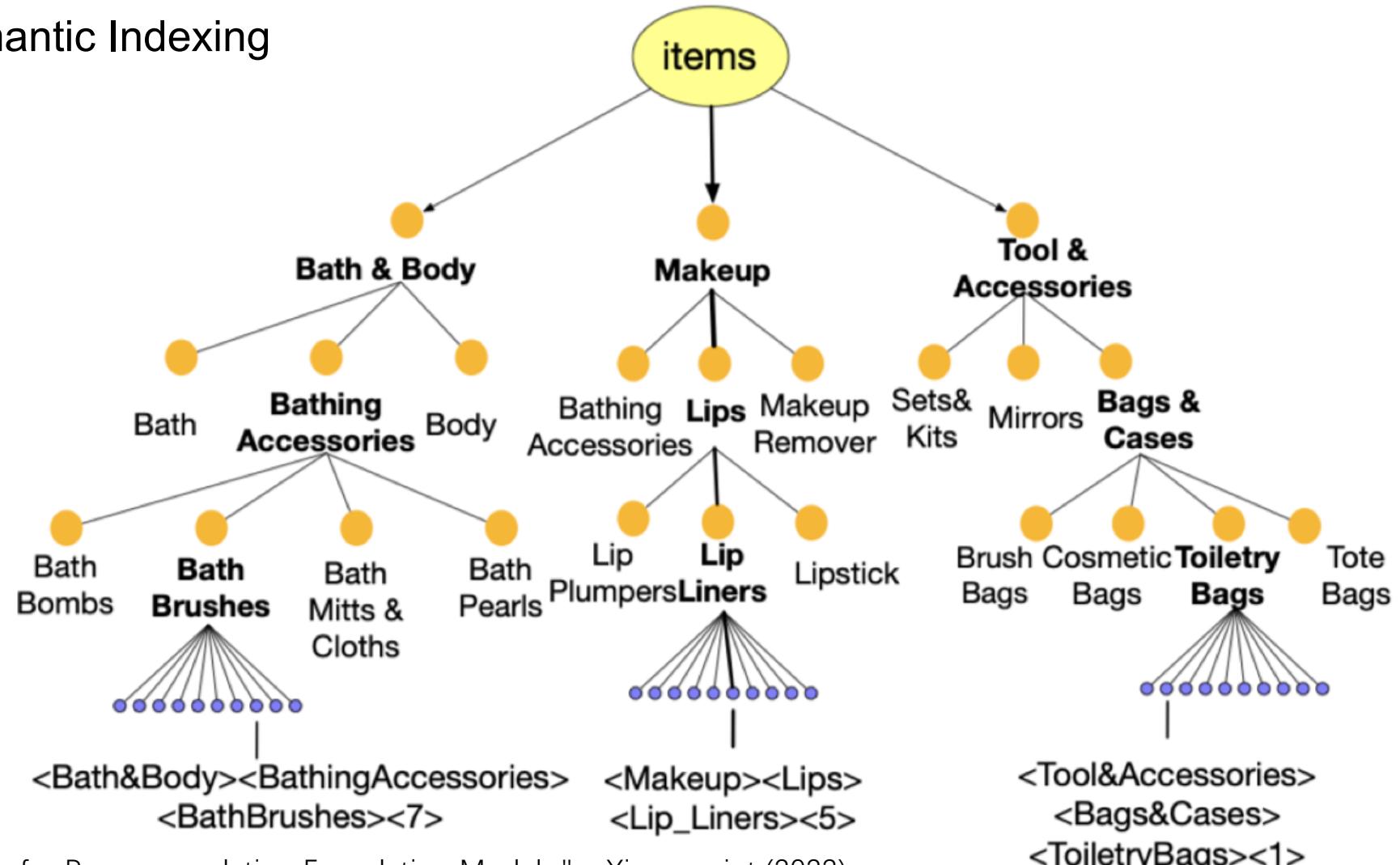
- Introducing more Information to the ID representation
  - ❖ Collaborative Indexing



# ID-based LLM RecSys

- Introducing more Information to the ID representation

- ❖ Semantic Indexing



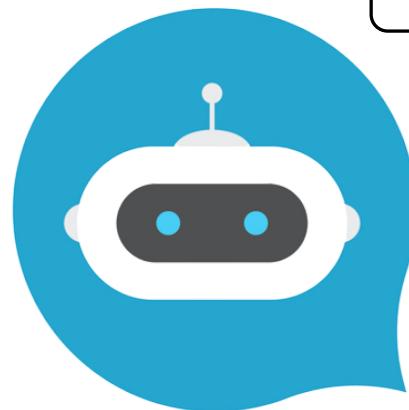
# ID-based LLM RecSys

- Modelling user interaction history with Markov chain



Question

User\_4782 has bought  
3472, 7653, 0192, 4271.  
What will he buy next?



RecSys

User\_4782 will buy 7251.

# ID-based LLM RecSys

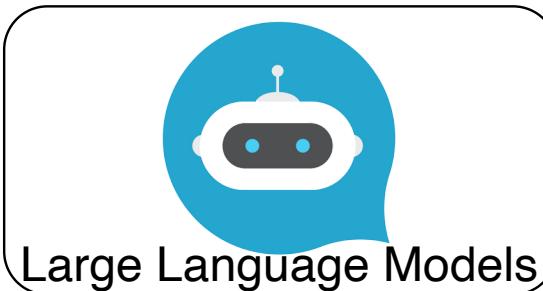
- Modelling user interaction history with Markov chain

User\_0513 bought  
3472, 1784, 2563,  
8892, 4271, 0988

User\_4781 bought  
7562, 8892, 4057,  
0192, 7251

.....

User\_2788 bought  
4271, 7251, 9082,  
0192, 5672



Pre-training & Fine-tuning

$$v_{i+1} = \arg \max_v P(v_{i+1} | v_1, v_2, \dots, v_i)$$



User\_4782 will buy 7251.

Modelling the probability of the next item

# ID-based LLM RecSys

## □ The N-gram probability in NLP

### ❖ Unigram

$$P("3472") = \frac{1}{16}$$

$$P("2563") = \frac{1}{16}$$

$$P("4271") = \frac{2}{16}$$

$$P("7562") = \frac{1}{16}$$

$$P("0192") = \frac{2}{16}$$

$$P("9082") = \frac{1}{16}$$

$$P("1784") = \frac{1}{16}$$

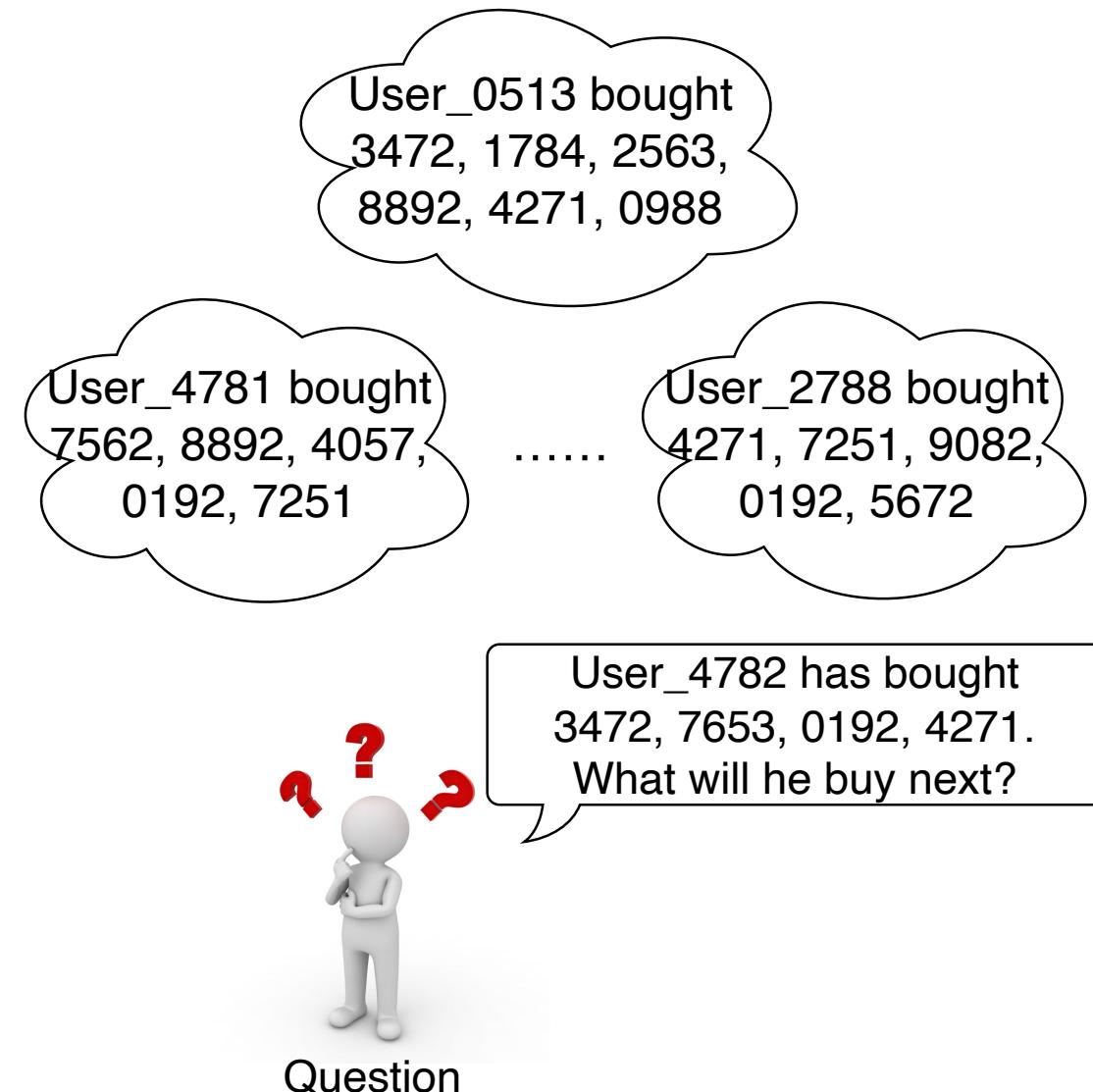
$$P("8892") = \frac{2}{16}$$

$$P("0988") = \frac{1}{16}$$

$$P("4057") = \frac{1}{16}$$

$$P("7251") = \frac{2}{16}$$

$$P("5672") = \frac{1}{16}$$



# ID-based LLM RecSys

## □ The N-gram probability in NLP

### ❖ Bigram

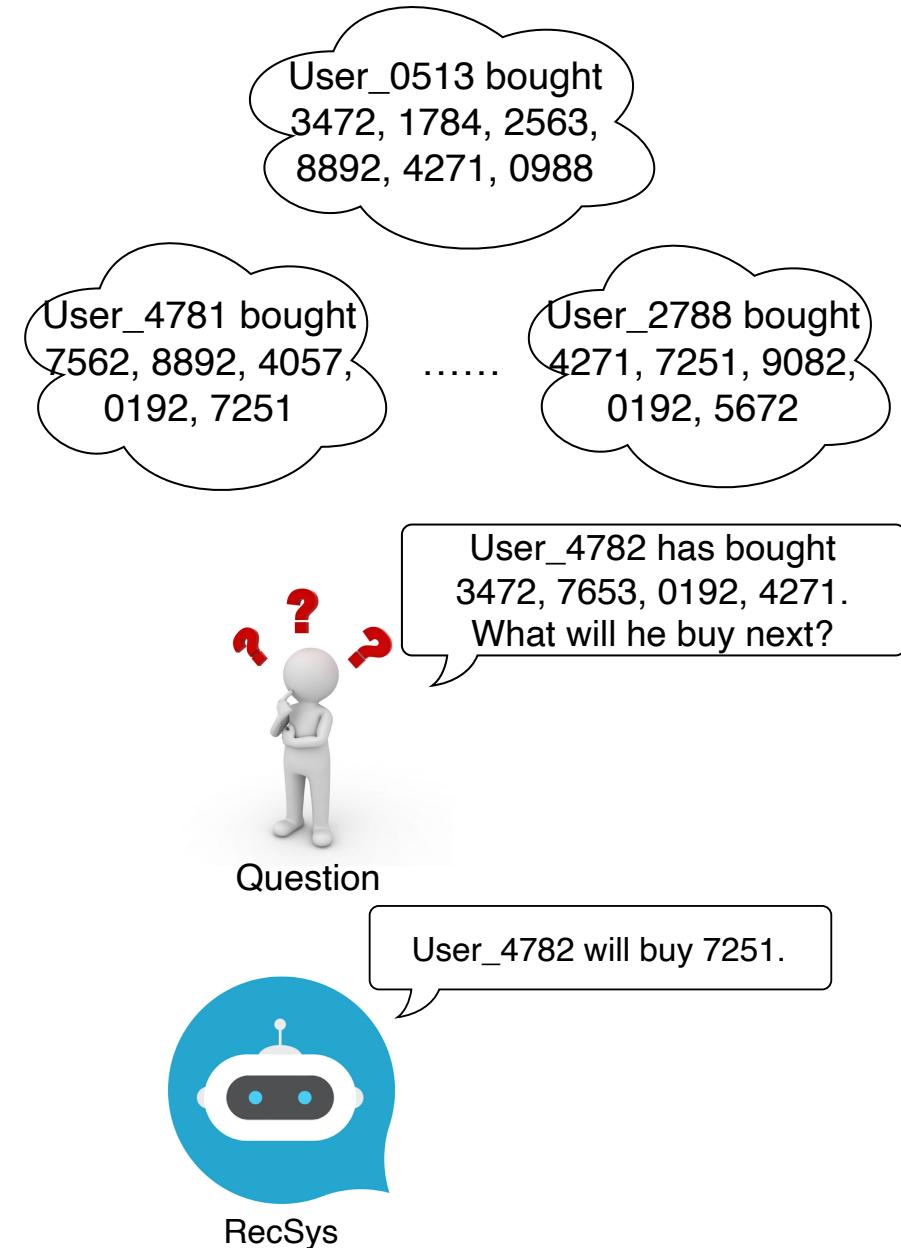
$$P("0988" | "4271") = \frac{1}{2}$$

$$P("7251" | "4271") = \frac{1}{2}$$

### ❖ Which one to choose?

$$P("0988") = \frac{1}{16}$$

$$P("7251") = \frac{2}{16}$$



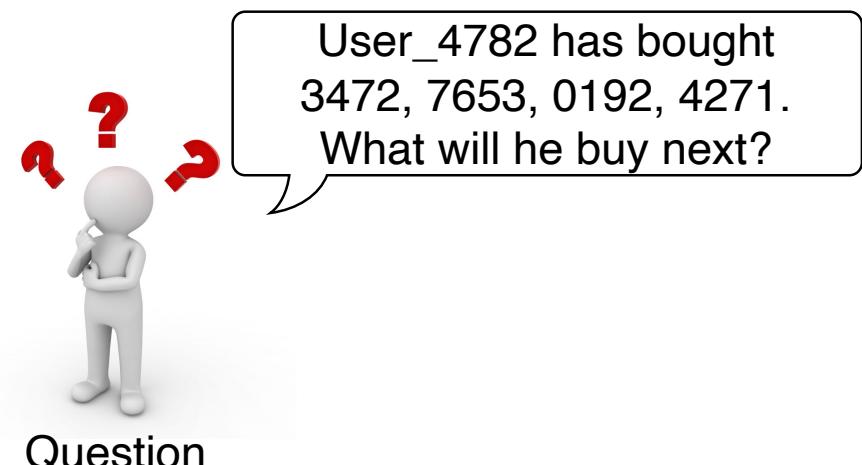
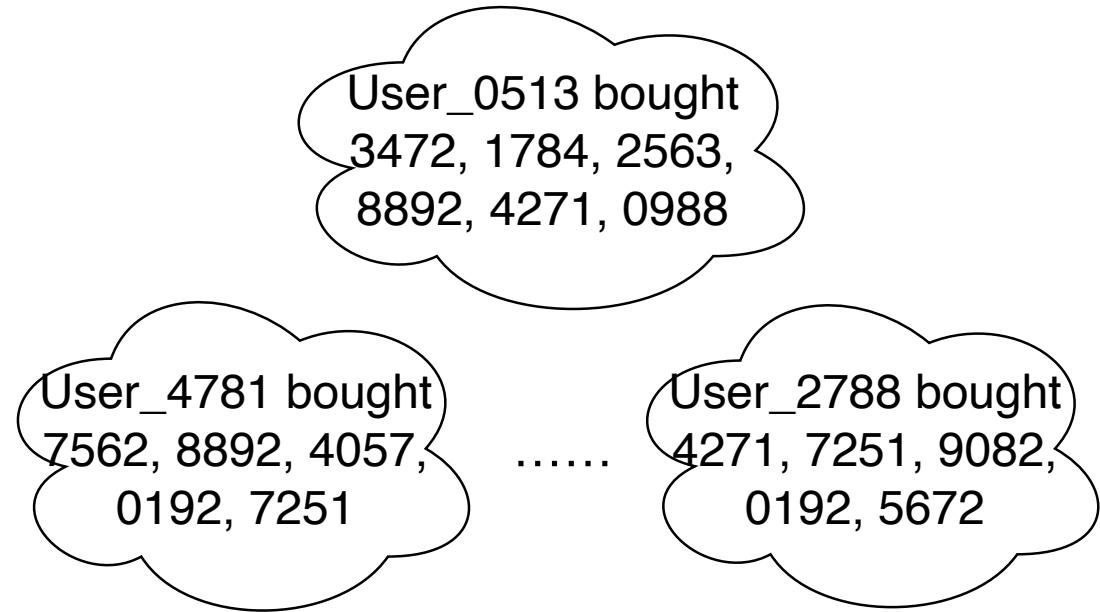
# ID-based LLM RecSys

## □ The N-gram probability in NLP

- ❖ The co-occurrence of item IDs
- ❖ User\_0513 bought 3472, ..., 4271, 0988
- ❖ User\_4782 bought 3472, ..., 4271, ?



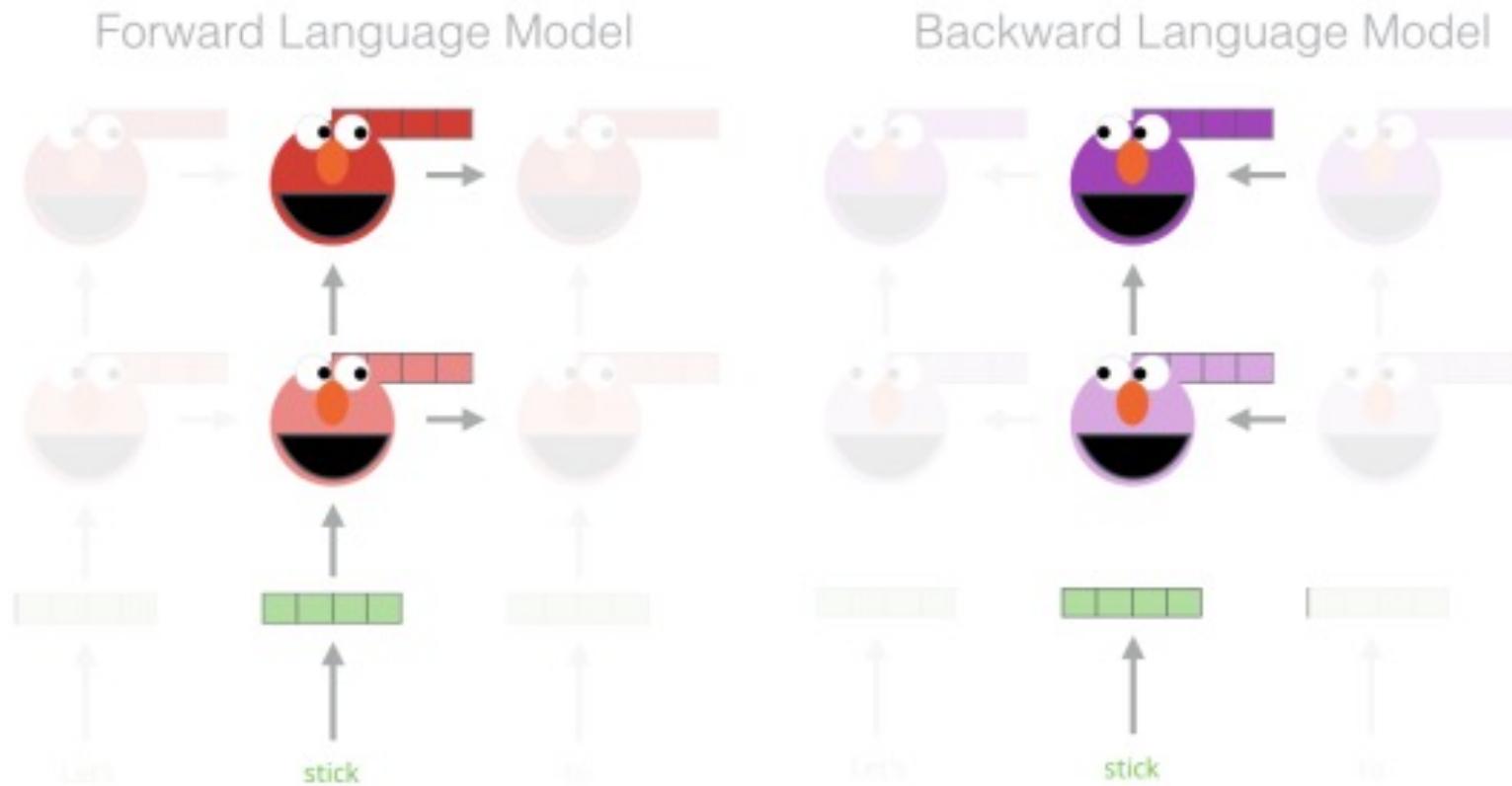
- ❖ Is “0988” a better answer than “7251”?



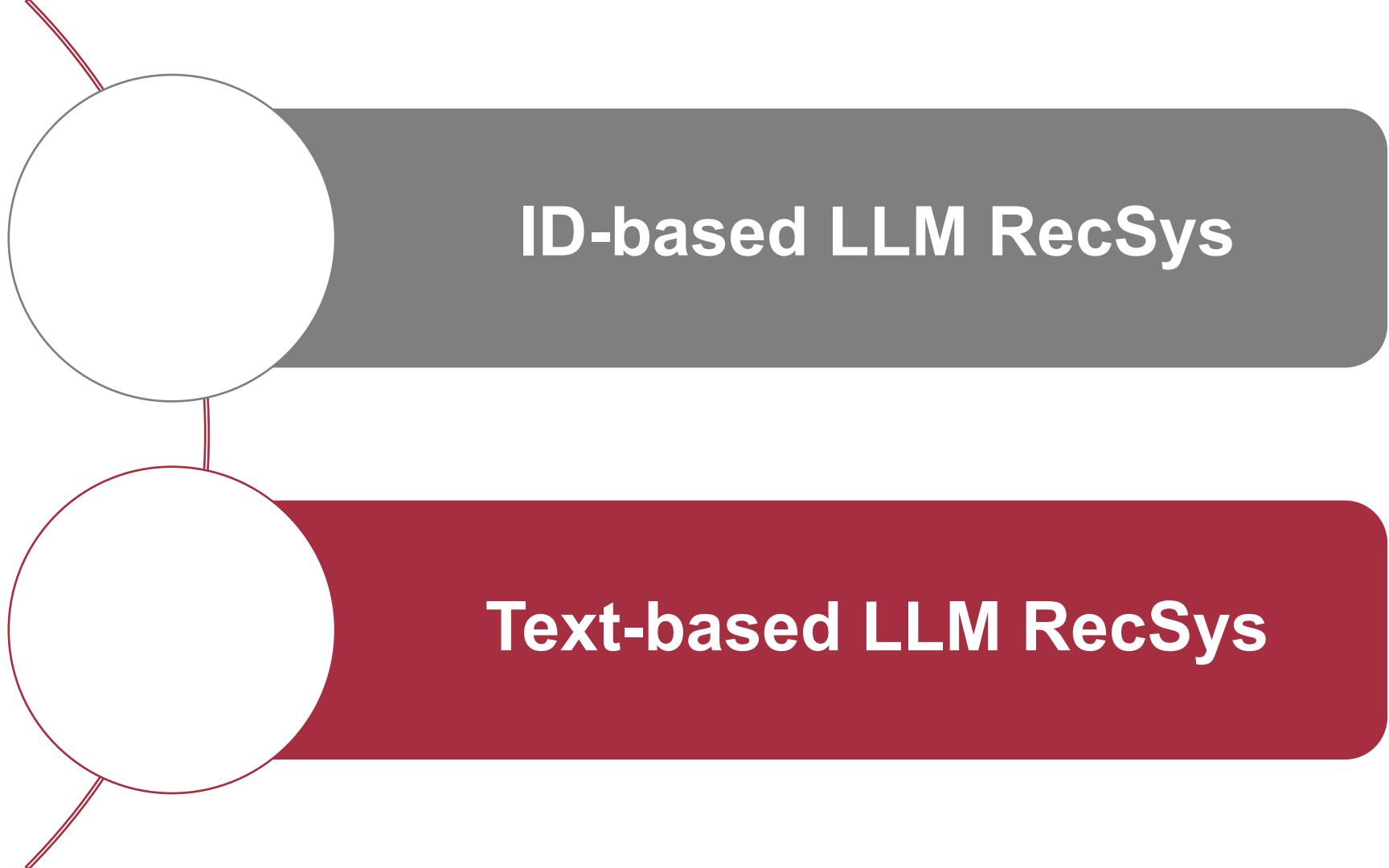
# ID-based LLM RecSys

## □ Contextual representations of words in LLMs

- ❖ User\_0513 bought 3472, 1784, 2563, 8892, 4271, 0988
- ❖ User\_4782 bought 3472, 7653, 0192, 4271, ?
- ❖ The item representations can vary for different contexts



# User & Item Representation in LLMs



**ID-based LLM RecSys**

**Text-based LLM RecSys**

# Text-based LLM RecSys

## □ GPT4Rec

- ❖ Item title contains rich semantic information
- ❖ It's a natural way to use text to describe items

Previously, the customer has bought

Ben Nye Banana Luxury Face Powder 3.0 oz Makeup Kim Kardashian NEW!!!.  
Rosallini Women Stainless Steel Extension Eyelash Applicator Tool Fish Tail Clip.  
Beauty Flawless Makeup Blender Sponge Puff (size 1). Fruit Of The Earth 100%  
Aloe Vera 24oz Gel Pump.

In the future, the customer wants to buy

Fine-tuned GPT-2



Ben Nye Luxury Powders - Banana 1.5oz.  
Beautyblender Solid Blendercleanser 1 oz.  
Professional 15 Color Concealer Camouflage Makeup Palette.  
Pro Beauty Makeup Sponge Blender Flawless Smooth Shaped Water Droplets Puff (Random Color).  
L'Oreal Paris True Match Super Blendable Makeup, Natural Buff, 1.0 Ounces.



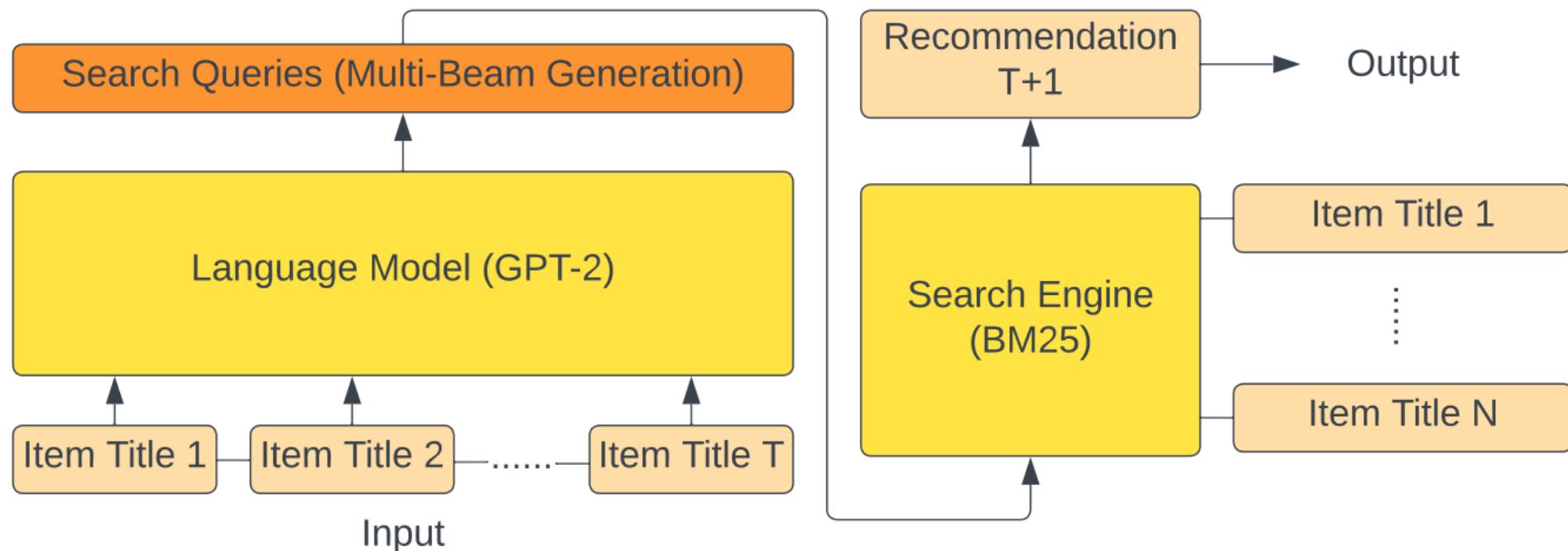
GPT4Rec



# Text-based LLM RecSys

## □ GPT4Rec

- ❖ In the era of LLMs, **Retrieval-Augmented Generation (RAG)** could be a way to improve the capability of LLMs
- ❖ RAG also enhances the explainability of LLM-based RecSys



# Text-based LLM RecSys

## □ TF-DCon

- ❖ Content-level condensation for recommendation
- ❖ Condense Item title and description to refine item representation

Enhance item titles based on given contents in the following format:

[title] {title}, [abstract] {abstract}, [category] {category}

You should rephrase the title to be clear, complete, objective, and neutral. Only provide the new title in the following format:

[newtitle] {newtitle}



[title] {Health Weightloss Watch},

[abstract] {Man Shares Time-Lapse Video of Six-Month Weight-Loss Journey We're big fans of weight-loss stories, but we usually only get to see the before and after photos. Very rarely do we get to see someone's physique transform right before our very eyes.},

[category] {Health}



[newtitle] {A Six-Month Weight-Loss Journey Captured in Time-Lapse Video},

## Content-Level Condensation

Preliminaries

Pre-training

Fine-tuning

Prompting

Future  
Directions

# Overview

**Presenter:**  
Jiatong Li

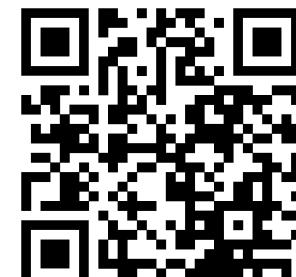


## User & Item Representation

- ID-based LLM RecSys
- Text-based LLM RecSys

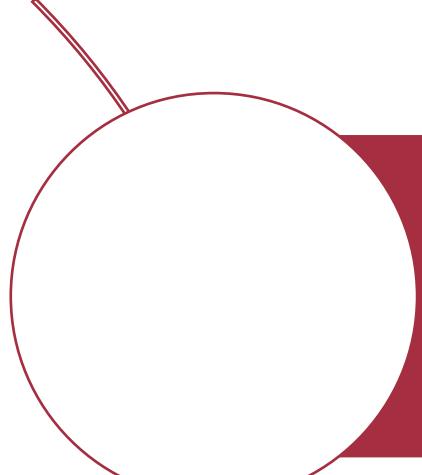
## Pre-training

- Pre-training in NLP
- Pre-training LLM-based RecSys

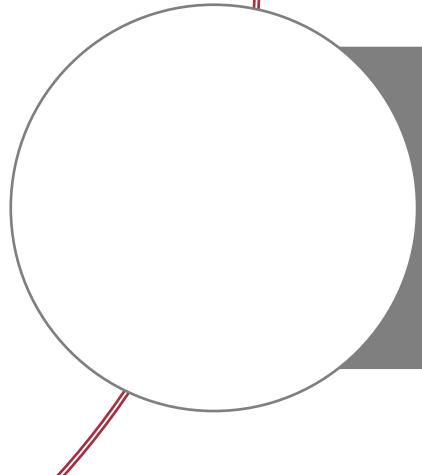


Website QR Code

# Pre-training



Pre-training in NLP

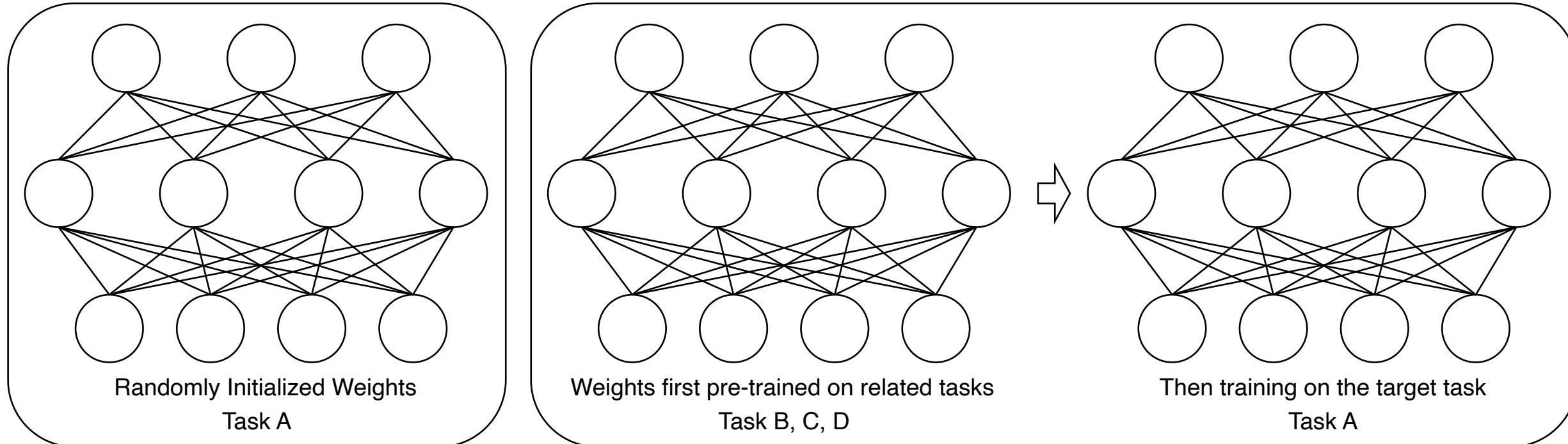


Pre-training LLM-based RecSys

# Pre-training in NLP

## □ What is pre-training?

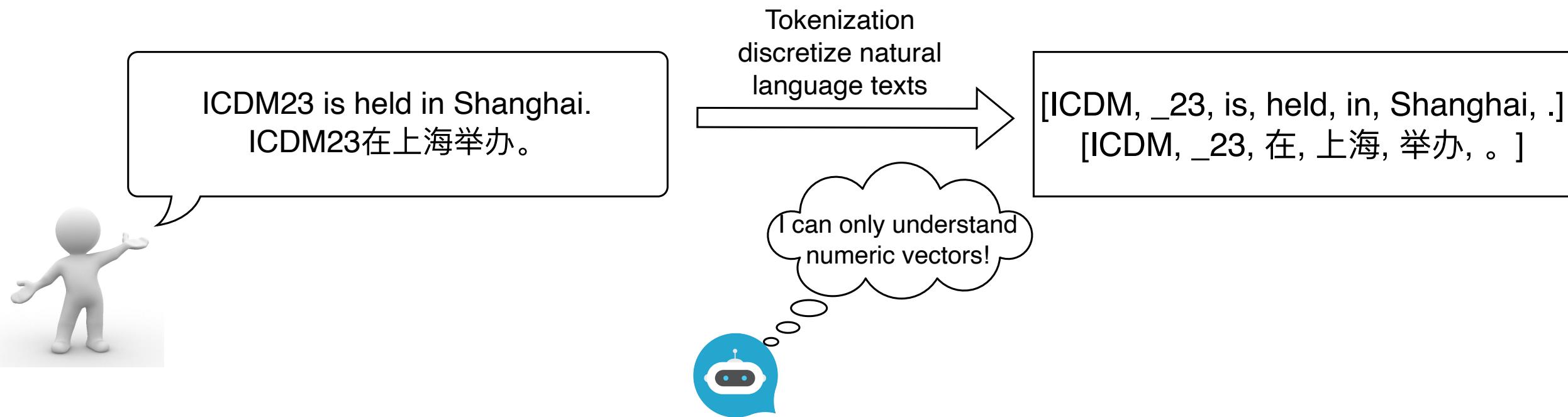
- ❖ Core Idea: knowledge transfer
- ❖ Technically:



# Pre-training in NLP

## □ Why pre-training?

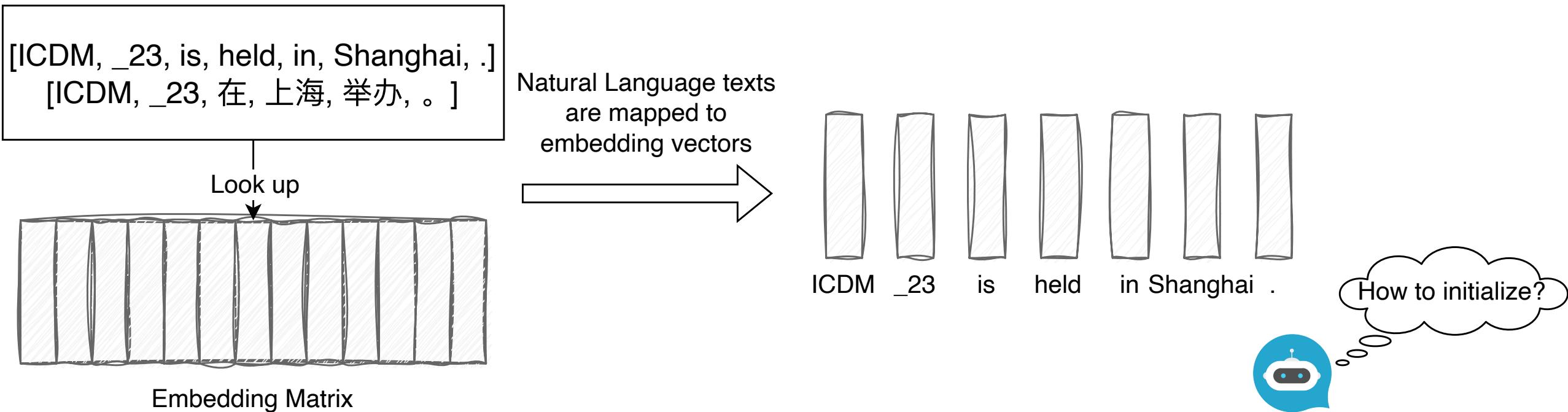
- ❖ Recall: Tokenization



# Pre-training in NLP

## □ Why pre-training?

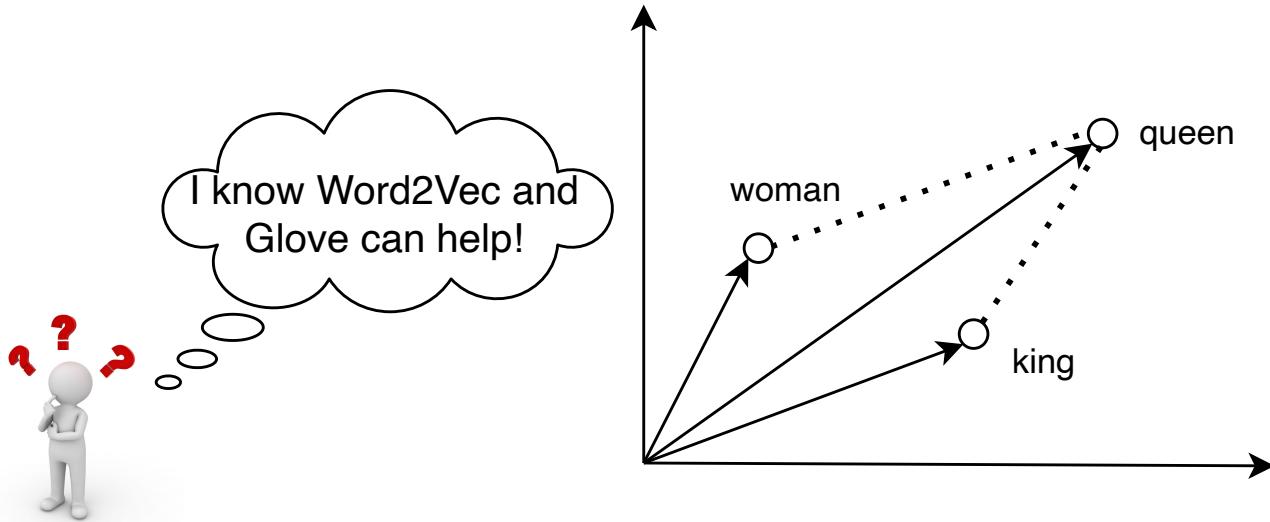
- ❖ Recall: Tokenization



# Pre-training in NLP

## ❑ Word embeddings?

- ❖ king: [-0.5, -0.9, 1.4, ...]
- ❖ queen: [-0.6, -0.8, -0.2, ...]
- ❖ woman: [-0.1, -0.1, -1.6, ...]



## ❑ Static word embeddings (word2vec, Glove) are pre-trained on text corpus from co-occurrence statistics

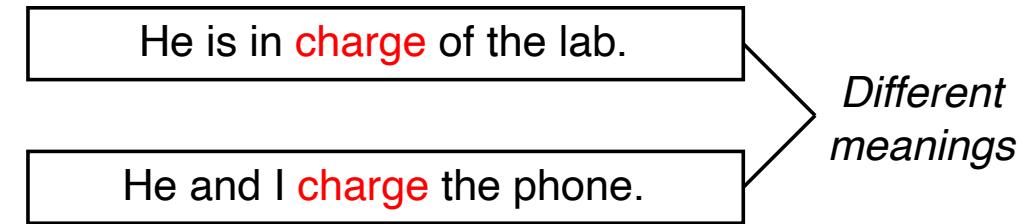
- ❖ He is the king of the country



- ❖ She is the queen of the country

# Pre-training in NLP

## □ Problem of static word embedding – Context-Free



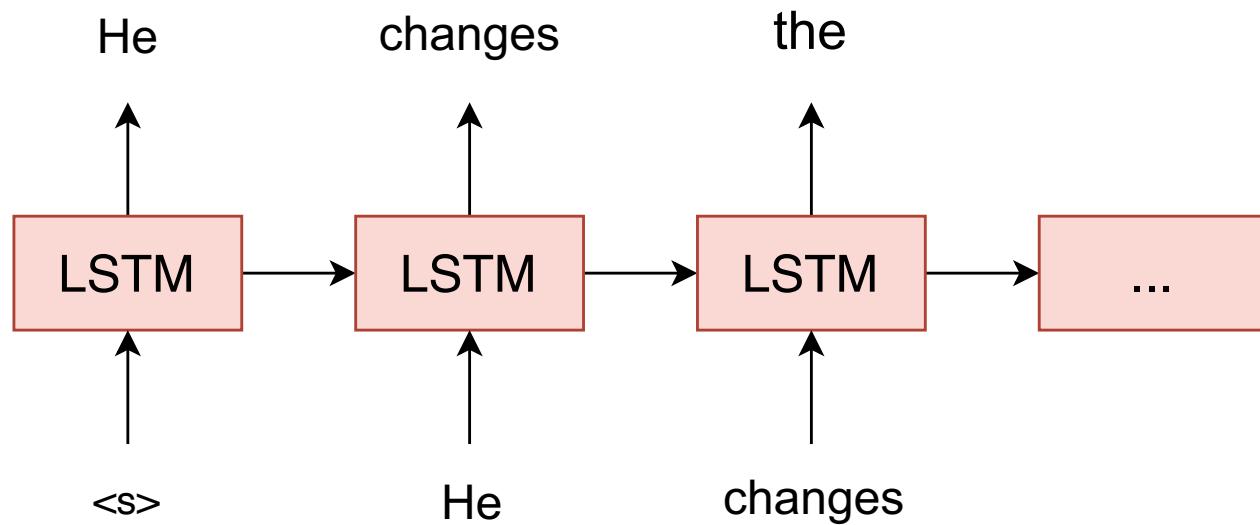
## □ How to solve it? – Contextual representations

- ❖ He is in **charge** of the lab
  - charge: [0.2, 0.8, 1.4, ...]
  
- ❖ He and I **charge** the phone
  - charge: [-0.3, -0.4, 0.7, ...]

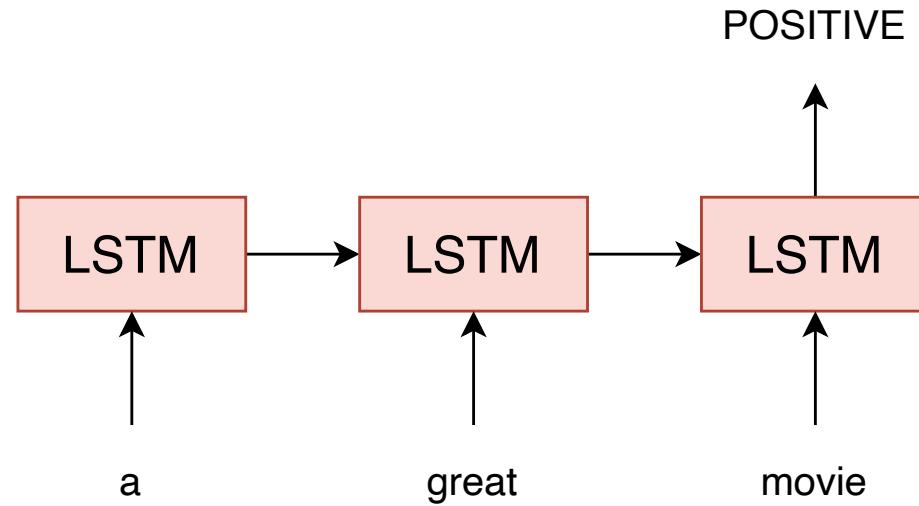
# Pre-training in NLP

## □ Semi-Supervised Sequence Learning

**Training LSTM as Language Model**



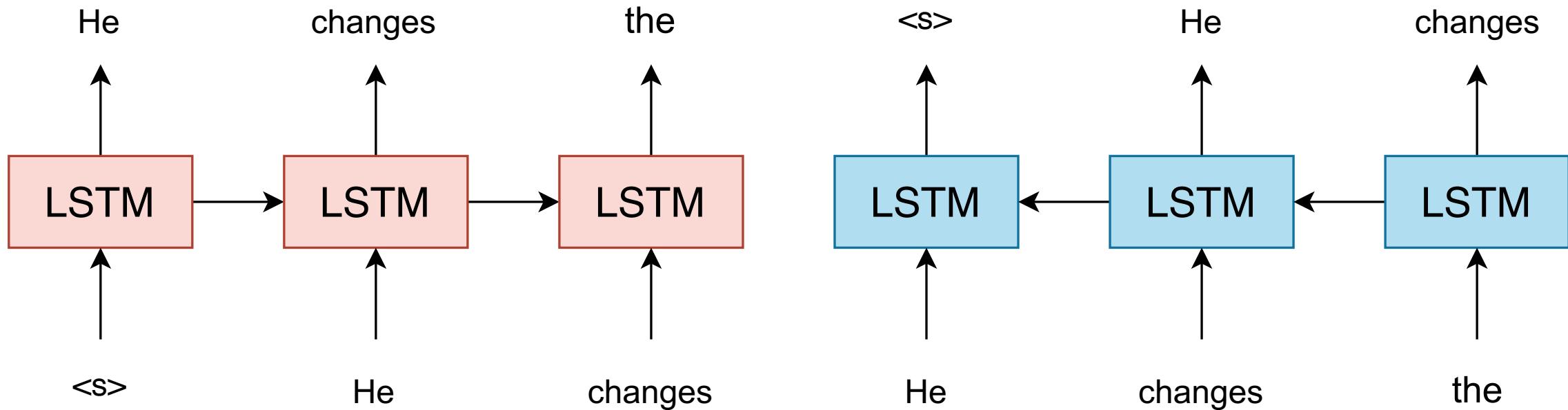
**Fine-tuning on Sentiment Classification**



# Pre-training in NLP

## □ ELMo: Deep Contextual Word Embeddings

### Training Separate Left-to-Right and Right-to-Left Language Models



# Pre-training in NLP

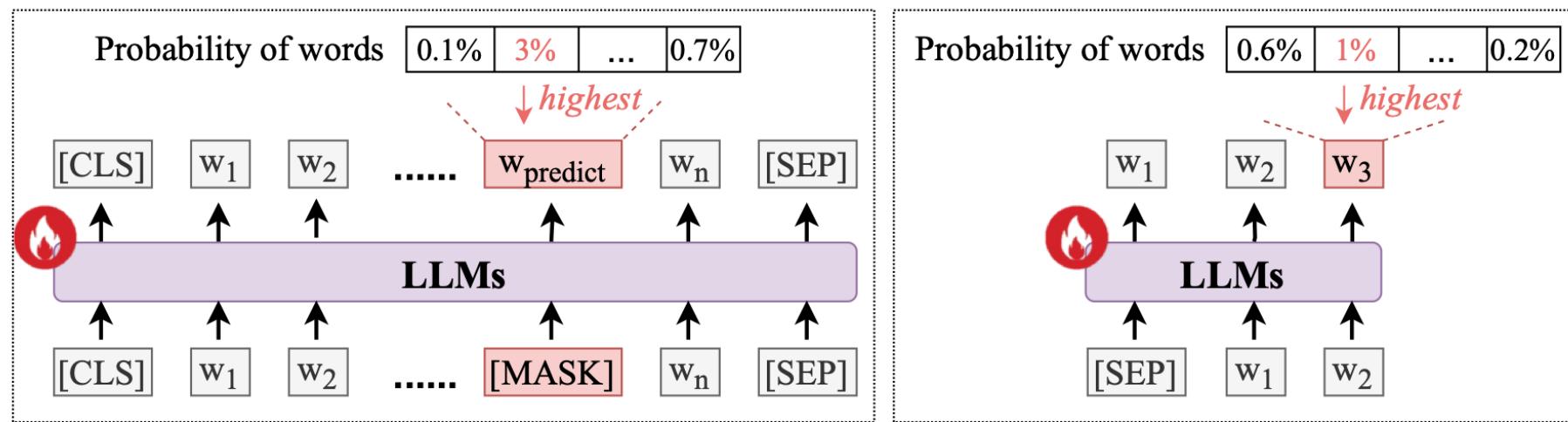
## □ Most Favored Pre-training Tasks in NLP

- ❖ Design specific pre-training tasks that could introduce knowledge
  - Masked Language Modelling (For Encoder-Decoder and Encoder-only Structures)
  - Next Token Prediction (For Decoder-only Structures)

### Pre-training



Large corpus  
unlabeled data

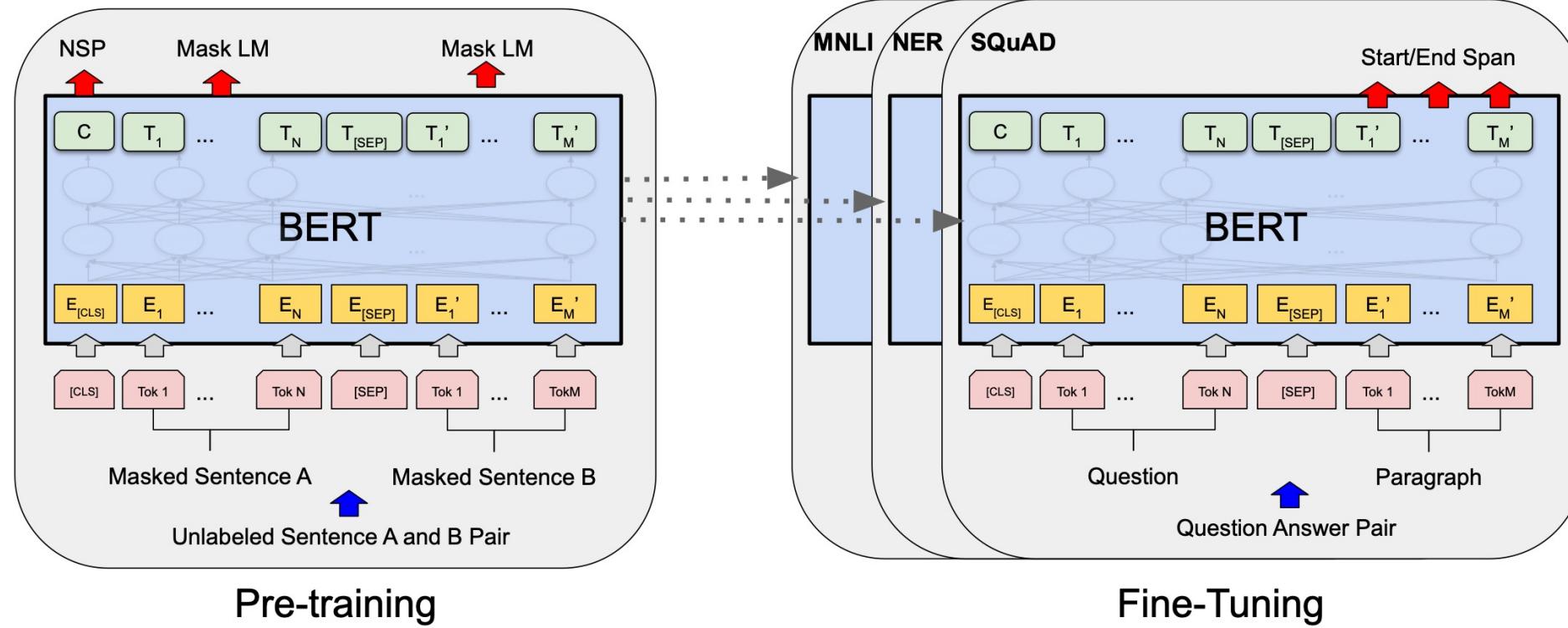


(Masked Language Modeling)

(Next Token Prediction)

# Pre-training in NLP

## □ BERT: Bidirectional Encoder Representations from Transformers

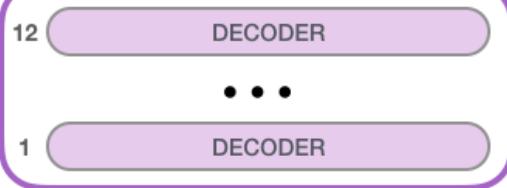


# Pre-training in NLP

## □ GPT



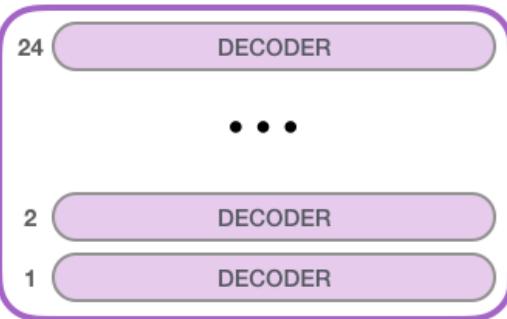
GPT-2  
SMALL



Model Dimensionality: 768



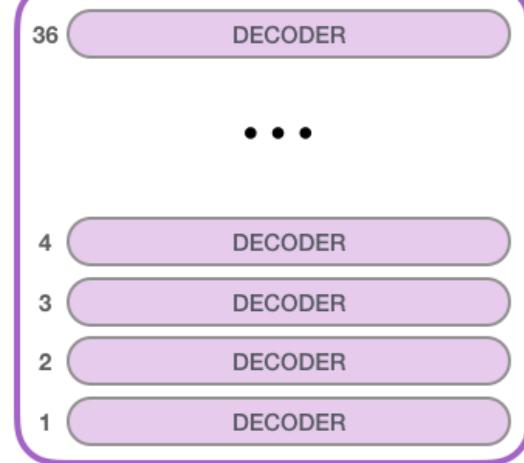
GPT-2  
MEDIUM



Model Dimensionality: 1024



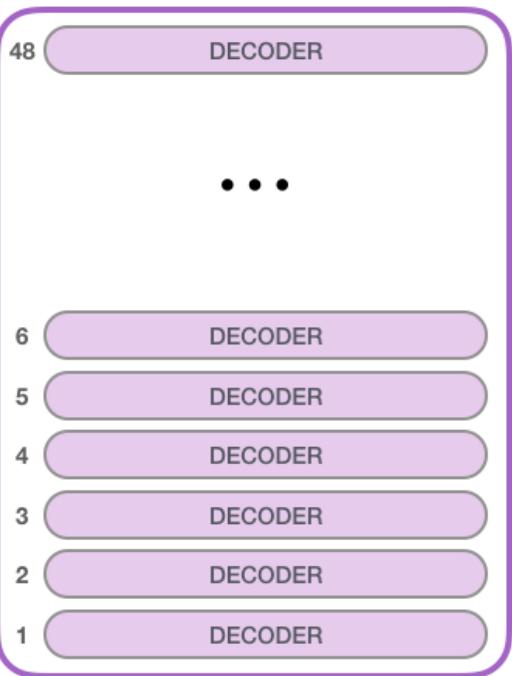
GPT-2  
LARGE



Model Dimensionality: 1280



GPT-2  
EXTRA  
LARGE



Model Dimensionality: 1600

# Pre-training



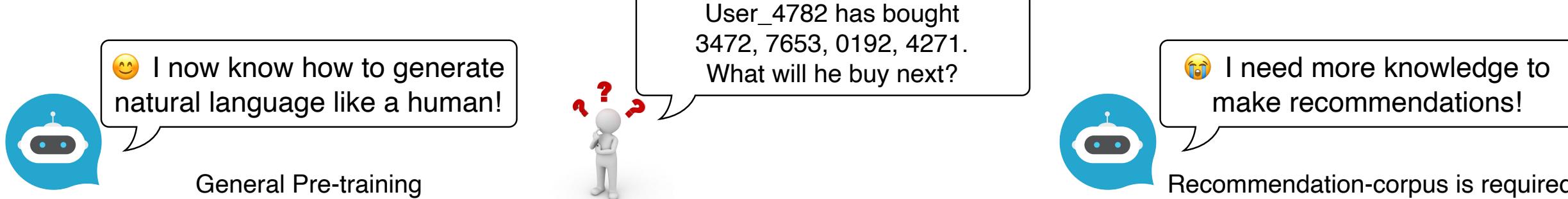
Pre-training in NLP

Pre-training LLM-based RecSys

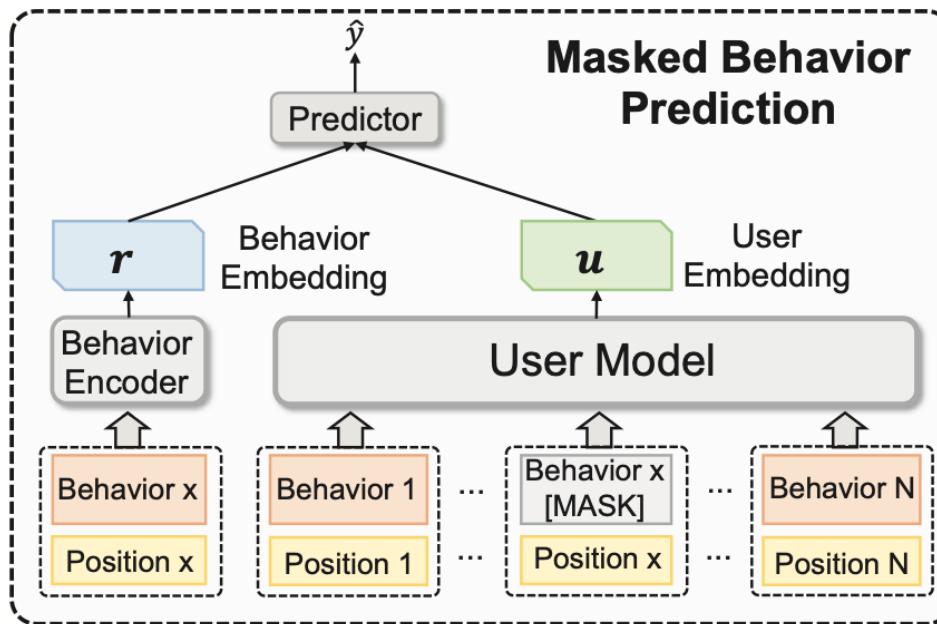
# Pre-training LLM-based RecSys

## □ What is Pre-training in LLM-based RecSys and Why is it Necessary?

- ❖ General pre-training vs. domain-specific pre-training
- ❖ Domain knowledge is essential for relieving the knowledge gap



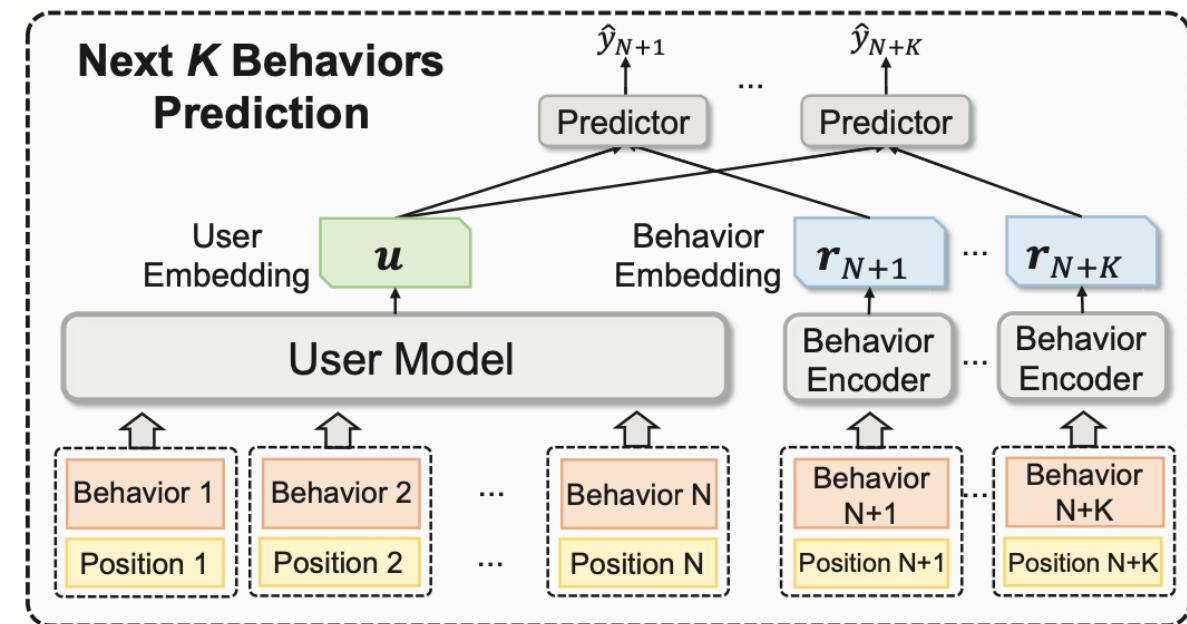
- Masked Behavior Prediction (MBP)
- Next K Behaviors Prediction (NBP)



(a) Masked Behavior Prediction (MBP) task.

$$\mathcal{L}_{MBP} = - \sum_{y \in \mathcal{S}_1} \sum_{i=1}^{P+1} y_i \log(\hat{y}_i)$$

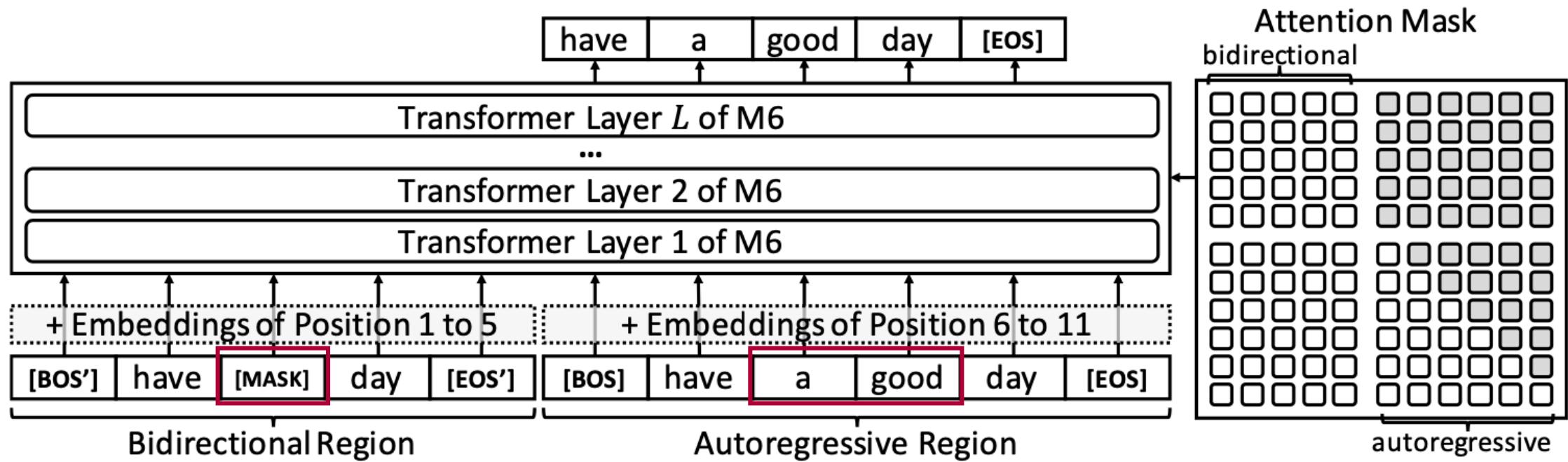
$$\mathcal{L} = \mathcal{L}_{MBP} + \lambda \mathcal{L}_{NBP}$$



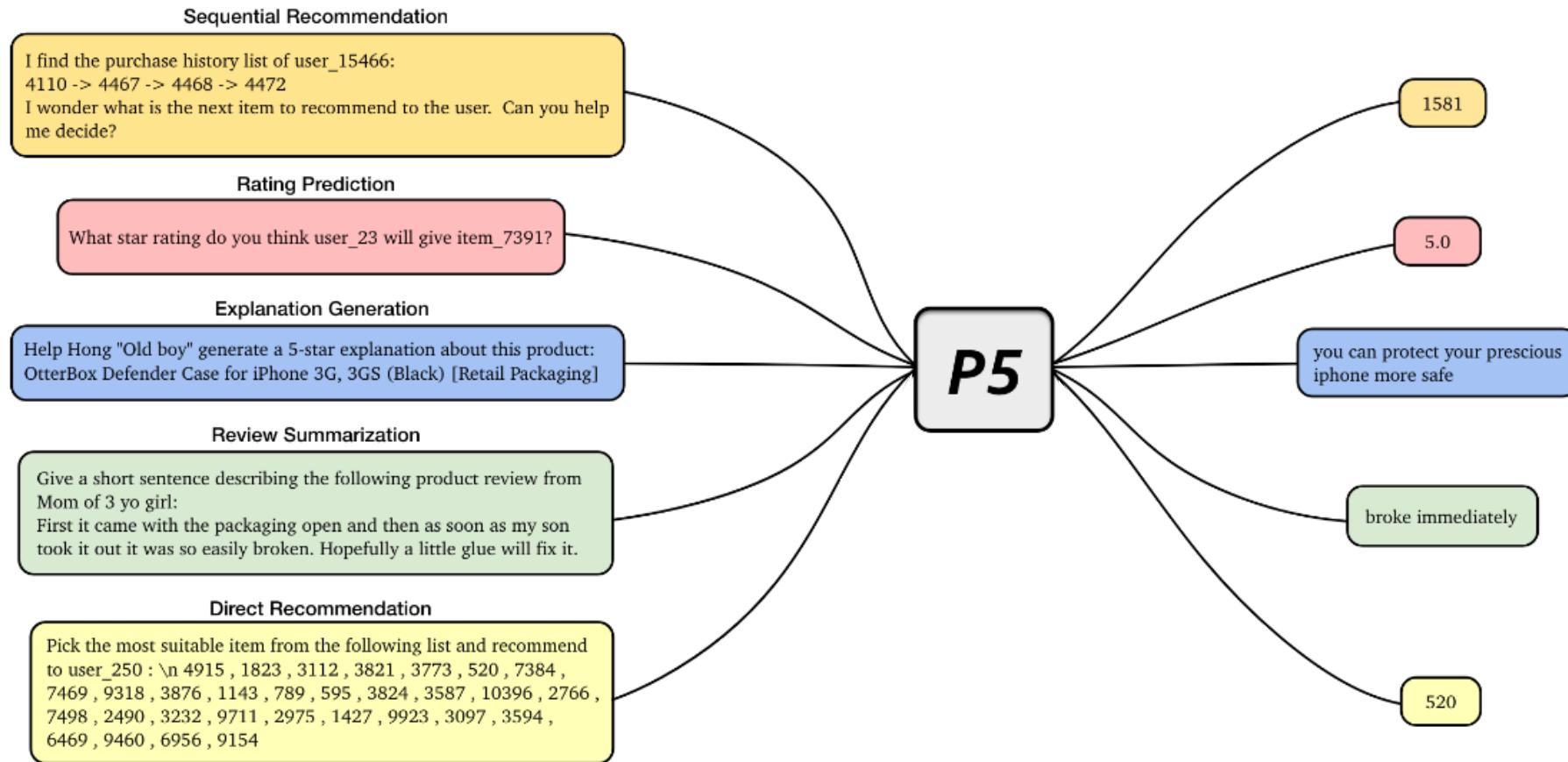
(b) Next  $K$  Behaviors Prediction (NBP) task.

$$\mathcal{L}_{NBP} = - \frac{1}{K} \sum_{y \in \mathcal{S}_2} \sum_{k=1}^K \sum_{i=1}^{P+1} y_{i,k} \log(\hat{y}_{i,k})$$

## □ Text-infilling



## □ Multi-task Pretraining with Personalized Prompt Collection



*Multi-task Pretraining with Personalized Prompt Collection*

Preliminaries

Pre-training

Fine-tuning

Prompting

Future  
Directions

# Overview

**Presenter:**  
Jiatong Li



## Fine-tuning

- ◎ Fine-tuning in NLP
- Fine-tuning LLM-based RecSys

## Parameter Efficient Fine-tuning

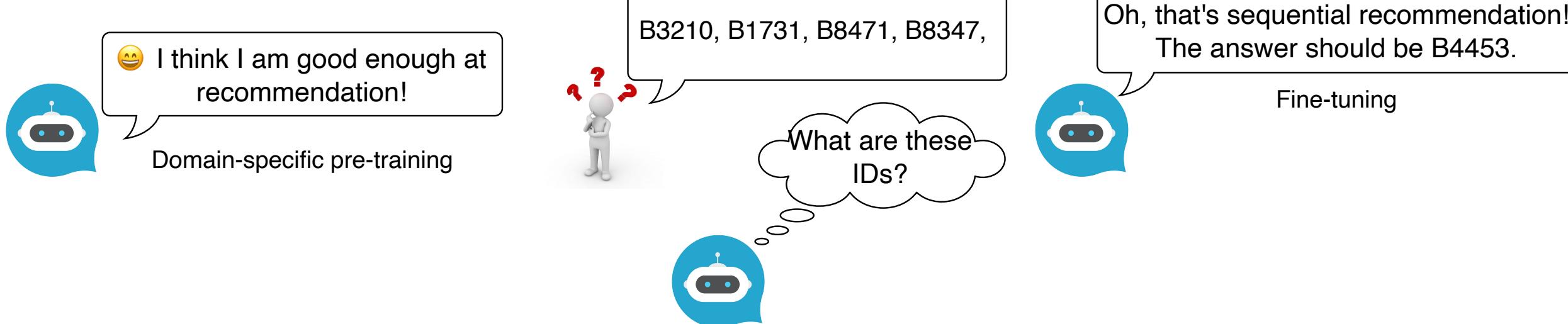


Website QR Code

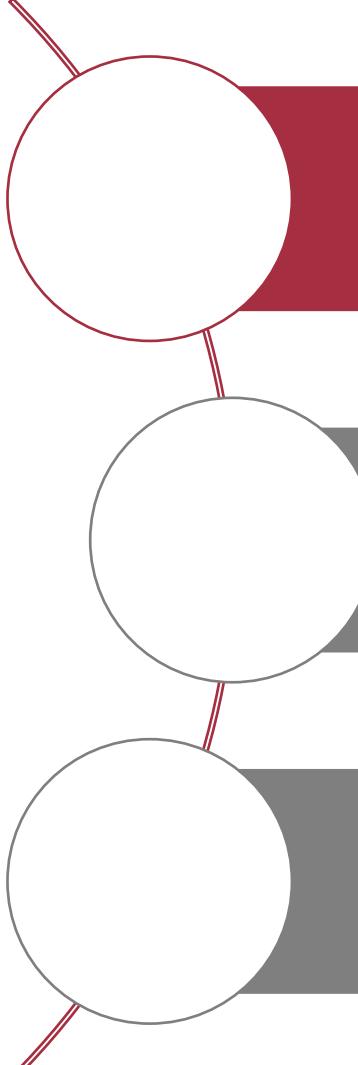
# Fine-tuning in NLP

## □ What is Fine-tuning and Why Fine-tuning?

- ❖ Gaps between the pre-training tasks and downstream tasks still exist
  - Masked Language Modelling v.s. Sentiment Classification
- ❖ Fine-tuning means training pre-trained LLMs on downstream tasks to fit the requirements
- ❖ Supervised Fine-tuning (SFT) and Fine-tuning with Reinforcement Learning



# Fine-tuning with Reinforcement Learning



**Reinforcement Learning based on  
Human Feedbacks (RLHF)**

**Proximal Policy Optimization  
(PPO)**

**Direct Preference Optimization  
(DPO)**

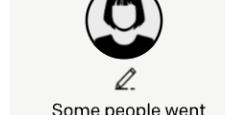
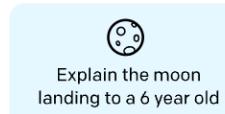
# Fine-tuning with Reinforcement Learning

## □ RLHF

Step 1

**Collect demonstration data, and train a supervised policy.**

A prompt is sampled from our prompt dataset.



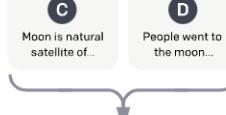
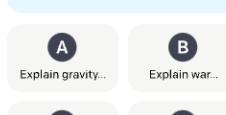
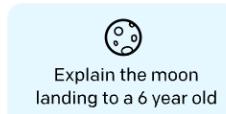
A labeler demonstrates the desired output behavior.

This data is used to fine-tune GPT-3 with supervised learning.

Step 2

**Collect comparison data, and train a reward model.**

A prompt and several model outputs are sampled.



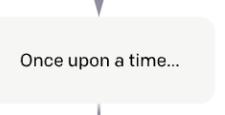
A labeler ranks the outputs from best to worst.

This data is used to train our reward model.

Step 3

**Optimize a policy against the reward model using reinforcement learning.**

A new prompt is sampled from the dataset.

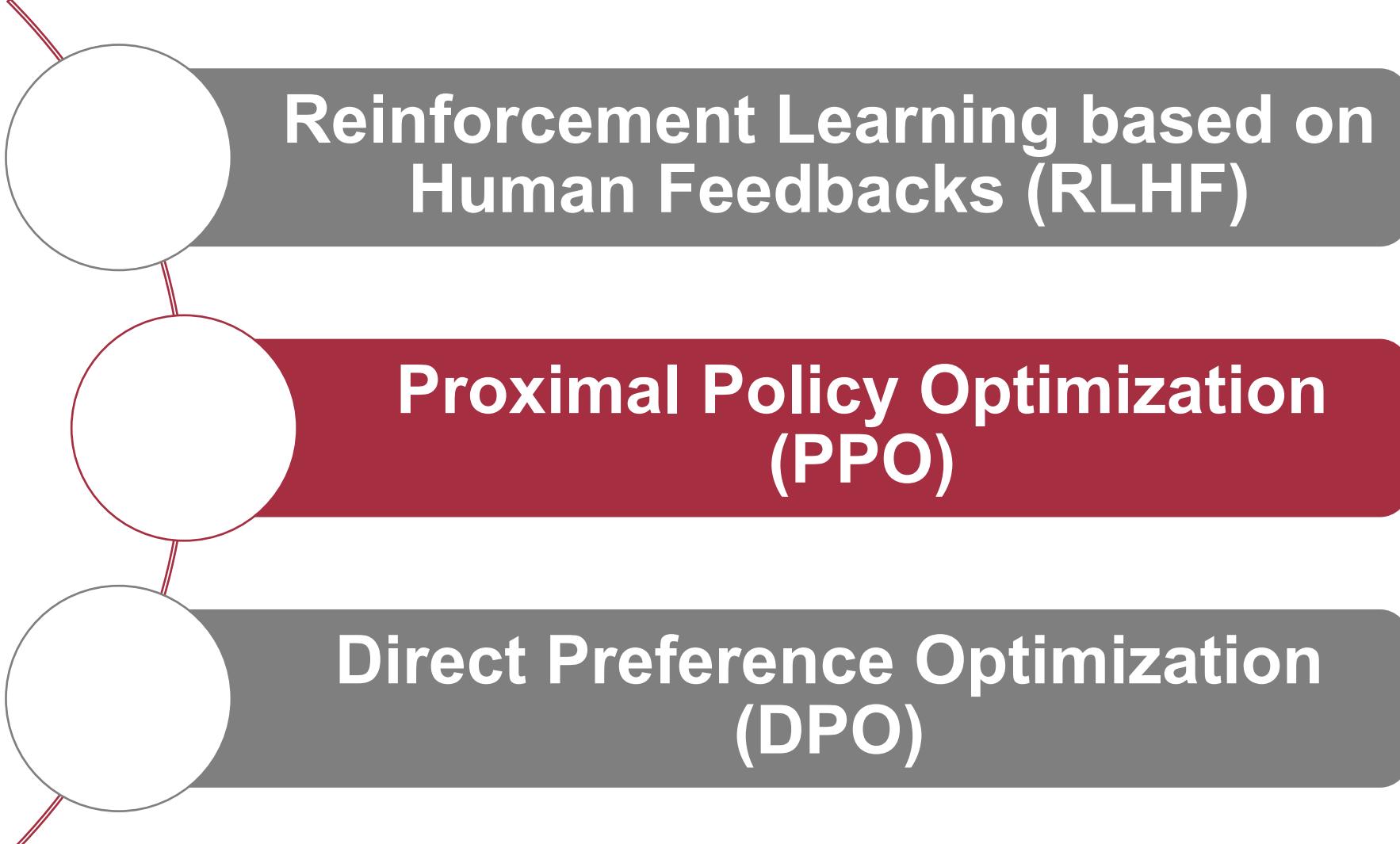


The policy generates an output.

The reward model calculates a reward for the output.

The reward is used to update the policy using PPO.

# Fine-tuning with Reinforcement Learning



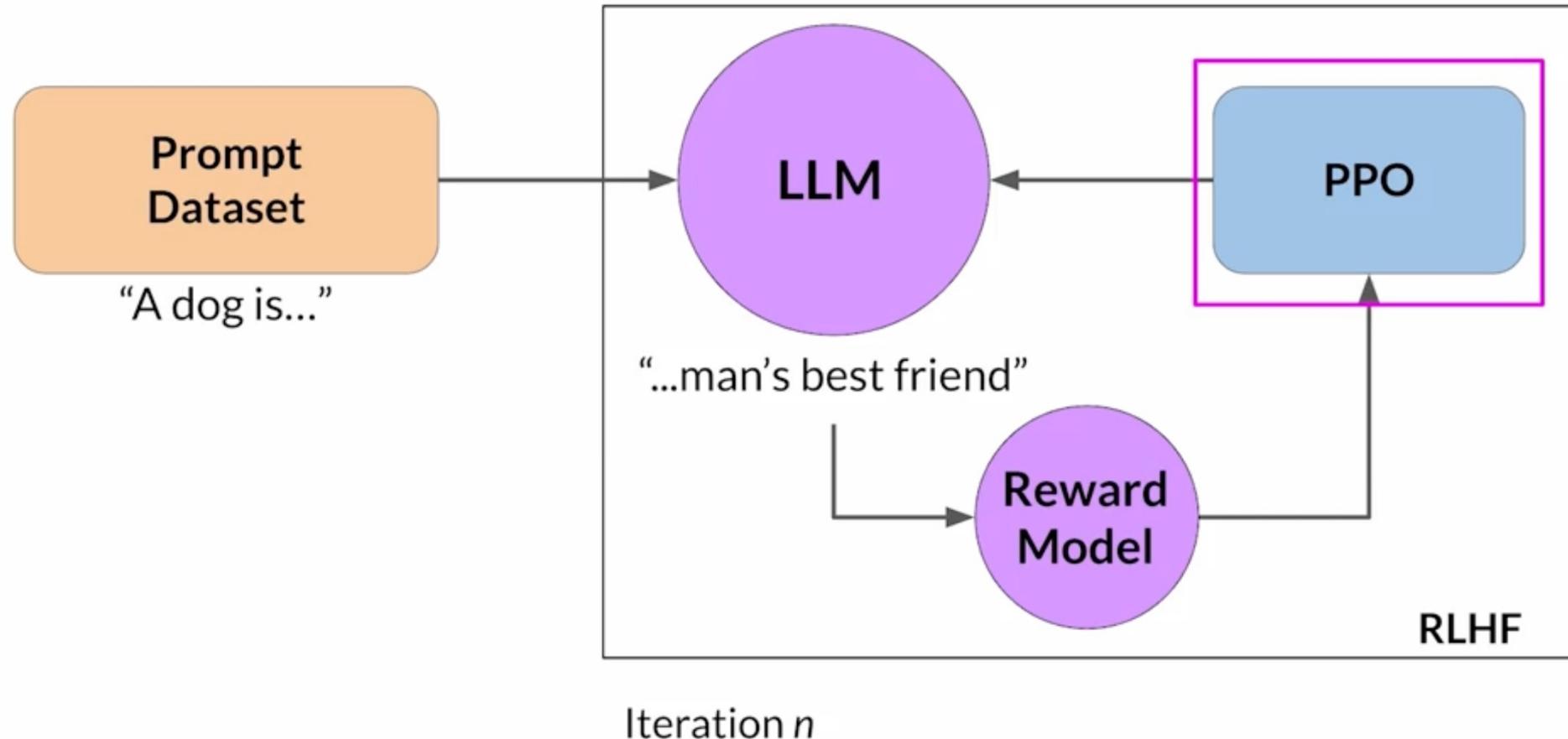
Reinforcement Learning based on Human Feedbacks (RLHF)

Proximal Policy Optimization (PPO)

Direct Preference Optimization (DPO)

# Fine-tuning with Reinforcement Learning

## □ PPO



# Fine-tuning with Reinforcement Learning

## □ PPO

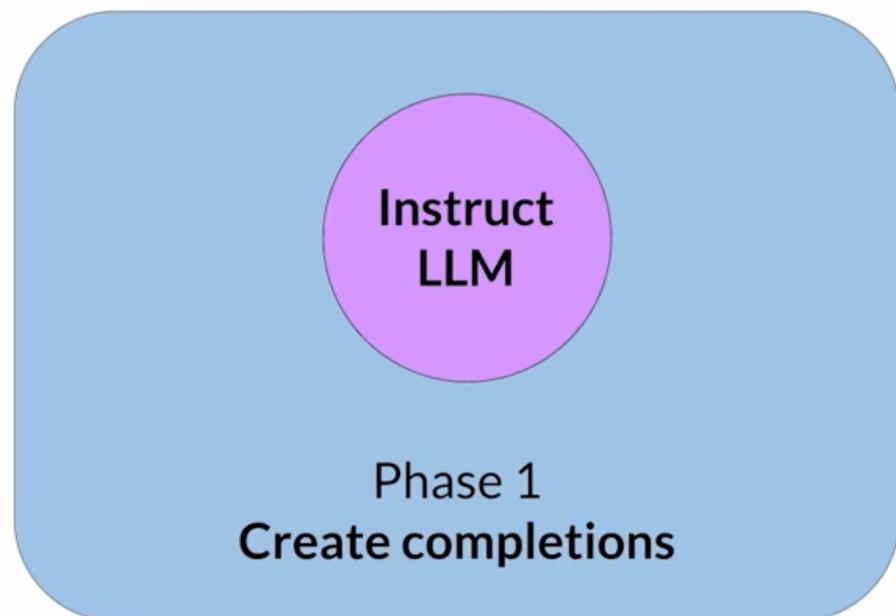
### Initialize PPO with Instruct LLM



# Fine-tuning with Reinforcement Learning

## □ PPO

### PPO Phase 1: Create completions



**Prompt**  
A dog is

**Completion**  
A dog is  
a furry animal

**Prompt**  
This house is

**Completion**  
This house is  
very ugly

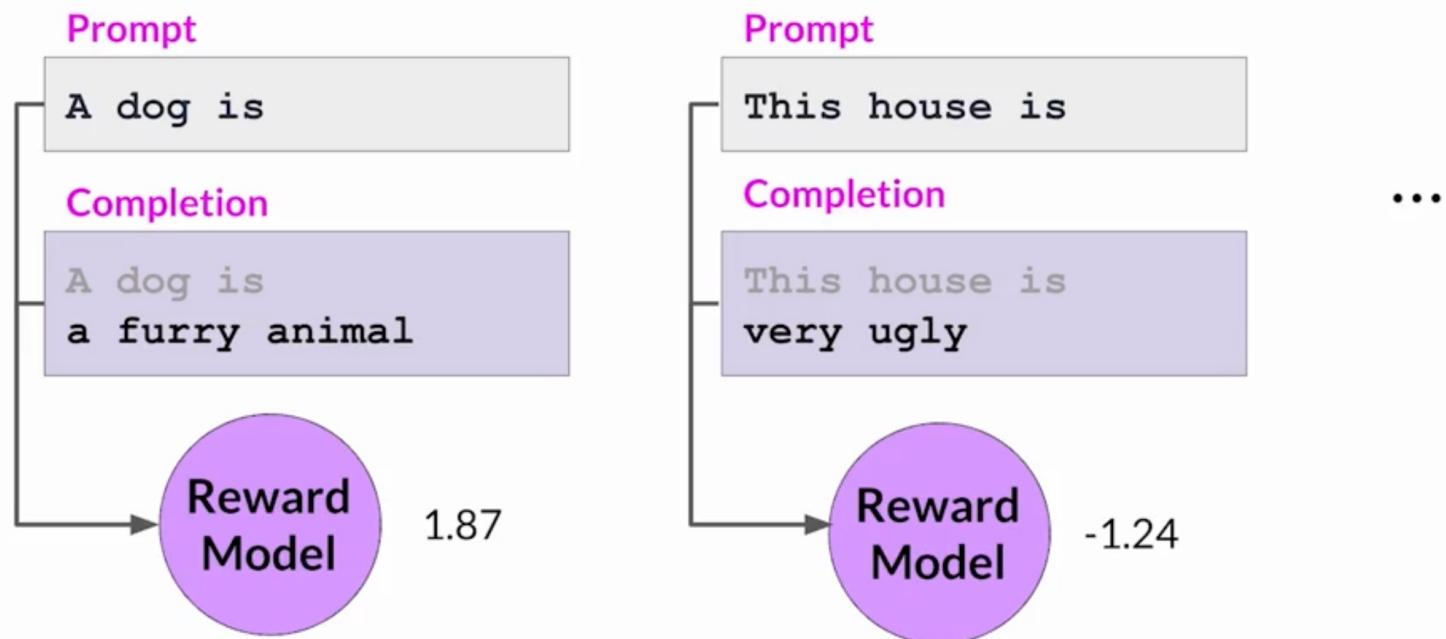
...

**Experiments**  
to assess the outcome of the current model,  
e.g. how helpful, harmless, honest the model is

# Fine-tuning with Reinforcement Learning

## □ PPO

### Calculate rewards



# Fine-tuning with Reinforcement Learning

## □ PPO

### Calculate value loss

Prompt

A dog is

Completion

A dog is  
a furry...

Value loss

$$L^{VF} = \frac{1}{2} \left\| V_\theta(s) - \left( \sum_{t=0}^T \gamma^t r_t \mid s_0 = s \right) \right\|_2^2$$

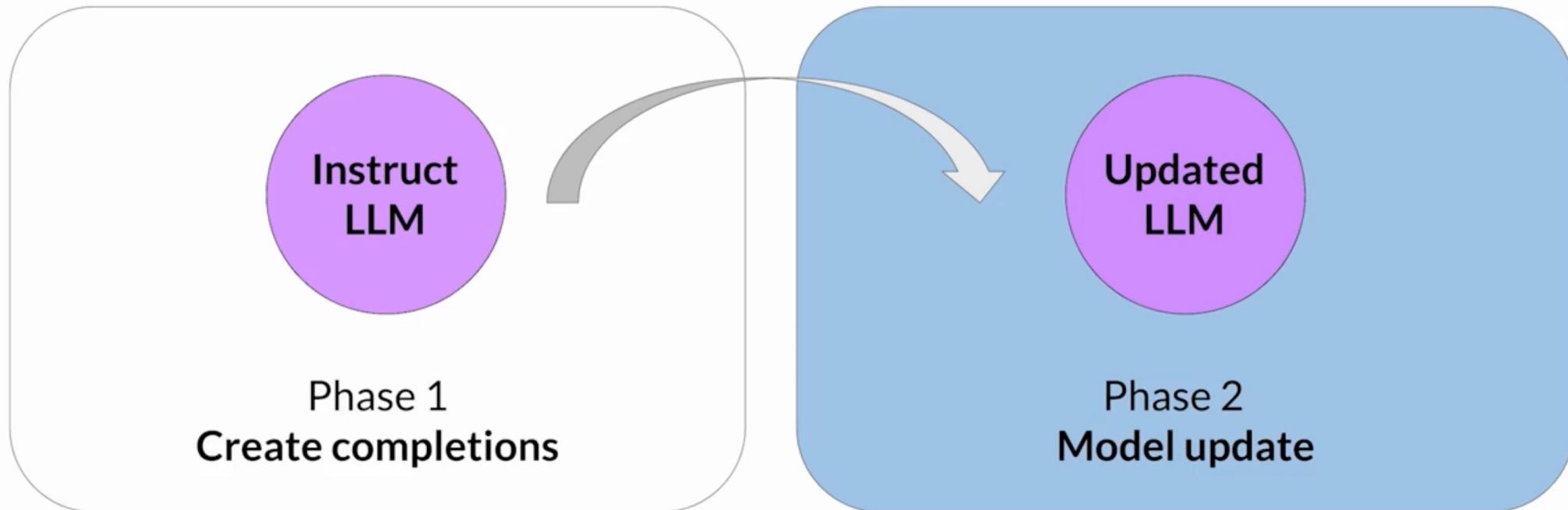
Estimated future total reward      Known future total reward

1.23      1.87

# Fine-tuning with Reinforcement Learning

## □ PPO

### PPO Phase 2: Model update



# Fine-tuning with Reinforcement Learning

## □ PPO

### PPO Phase 2: Calculate policy loss

$$L^{POLICY} = \min \left( \frac{\pi_{\theta}(a_t | s_t)}{\pi_{\theta_{old}}(a_t | s_t)} \cdot \hat{A}_t, \text{clip} \left( \frac{\pi_{\theta}(a_t | s_t)}{\pi_{\theta_{old}}(a_t | s_t)}, 1 - \epsilon, 1 + \epsilon \right) \cdot \hat{A}_t \right)$$

# Fine-tuning with Reinforcement Learning

## □ PPO

### PPO Phase 2: Calculate entropy loss

$$L^{ENT} = \text{entropy}(\pi_{\theta}(\cdot | s_t))$$

Low entropy:

**Prompt**

A dog is

**Completion**

A dog is  
a domesticated  
carnivorous mammal

**Prompt**

A dog is

**Completion**

A dog is  
a small carnivorous  
mammal

High entropy:

**Prompt**

A dog is

**Completion**

A dog is  
is one of the most  
popular pets around  
the world

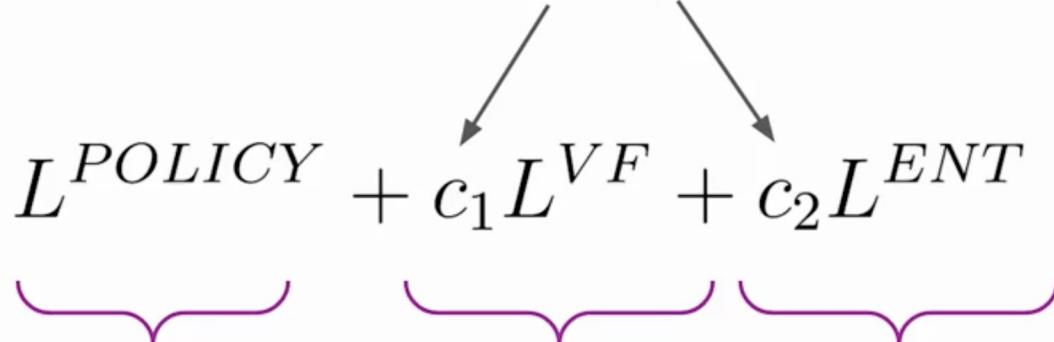
# Fine-tuning with Reinforcement Learning

## □ PPO

### PPO Phase 2: Objective function

$$L^{PPO} = L^{POLICY} + c_1 L^{VF} + c_2 L^{ENT}$$

Hyperparameters

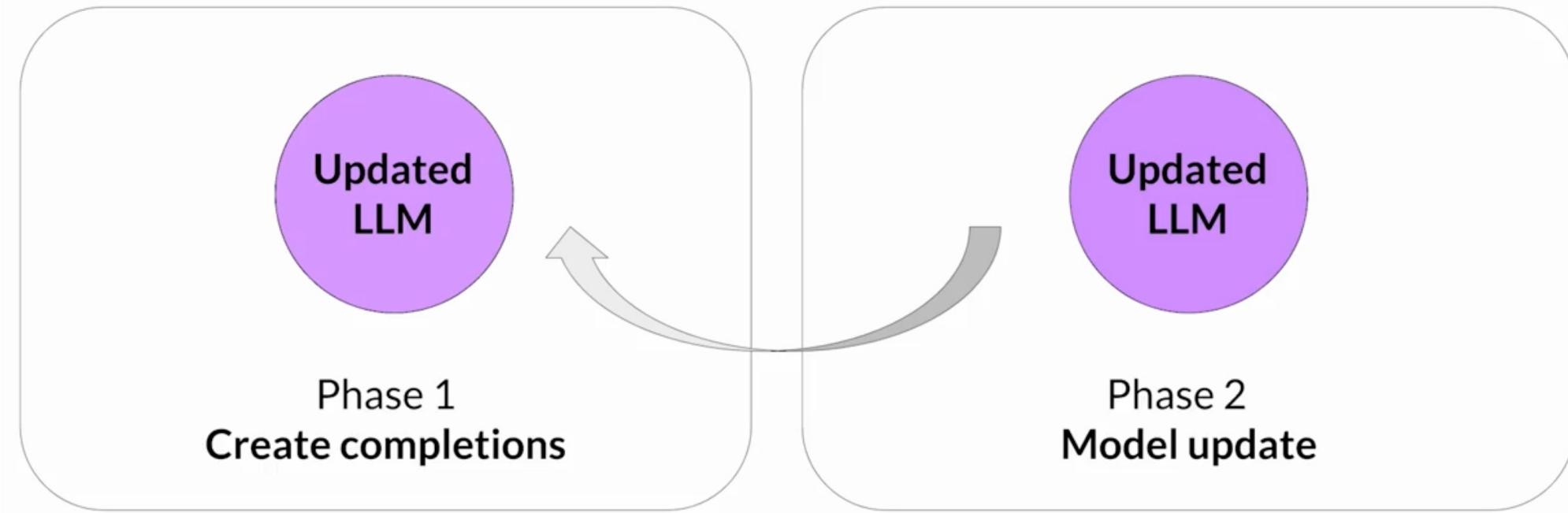


The diagram illustrates the PPO objective function. It starts with the equation  $L^{PPO} = L^{POLICY} + c_1 L^{VF} + c_2 L^{ENT}$ . Above the equation, the text "Hyperparameters" is centered. Two arrows point from this text down to the coefficients  $c_1$  and  $c_2$ . Below the equation, three curly braces group the terms: the first brace groups  $L^{POLICY}$ , the second groups  $c_1 L^{VF}$ , and the third groups  $c_2 L^{ENT}$ . Below each brace, the corresponding loss name is written: "Policy loss", "Value loss", and "Entropy loss".

# Fine-tuning with Reinforcement Learning

## □ PPO

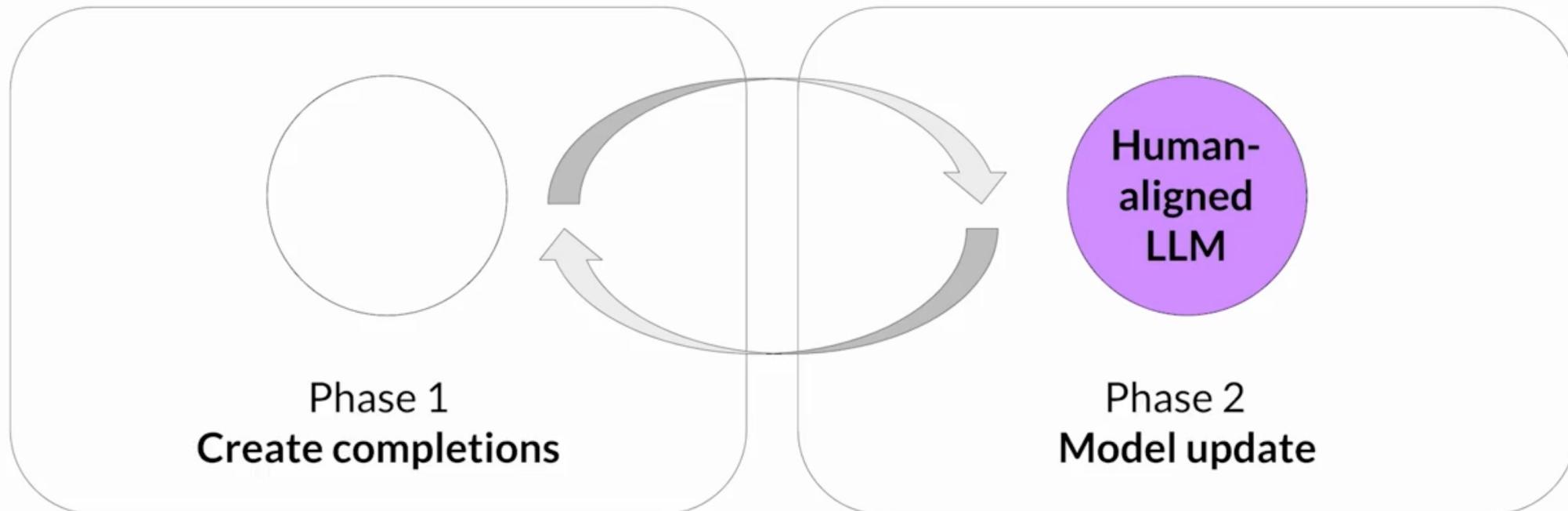
Replace LLM with updated LLM



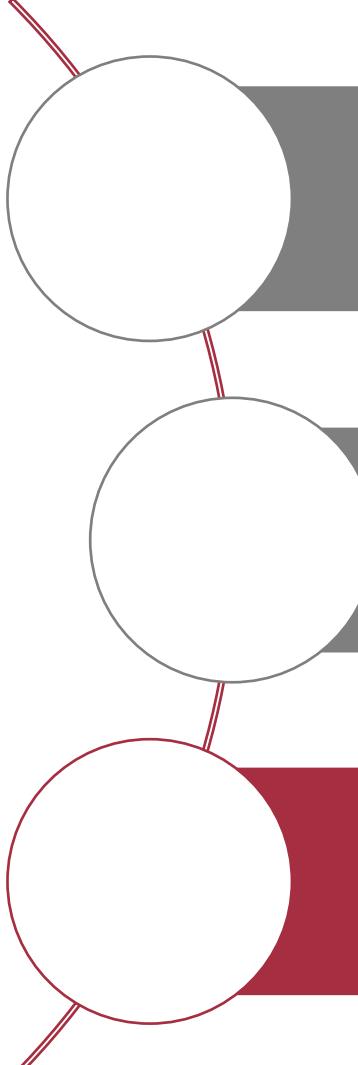
# Fine-tuning with Reinforcement Learning

## □ PPO

After many iterations, human-aligned LLM!



# Fine-tuning with Reinforcement Learning



Reinforcement Learning based on  
Human Feedbacks (RLHF)

Proximal Policy Optimization  
(PPO)

Direct Preference Optimization  
(DPO)

# Fine-tuning with Reinforcement Learning

## DPO

$$\mathcal{L}_{\text{DPO}}(\pi_\theta; \pi_{\text{ref}}) = -\mathbb{E}_{(x, y_w, y_l) \sim \mathcal{D}} \left[ \log \sigma \left( \beta \log \frac{\pi_\theta(y_w | x)}{\pi_{\text{ref}}(y_w | x)} - \beta \log \frac{\pi_\theta(y_l | x)}{\pi_{\text{ref}}(y_l | x)} \right) \right]$$

Policy to optimize

Aggregation over preference data

Shift in **preferred** completion

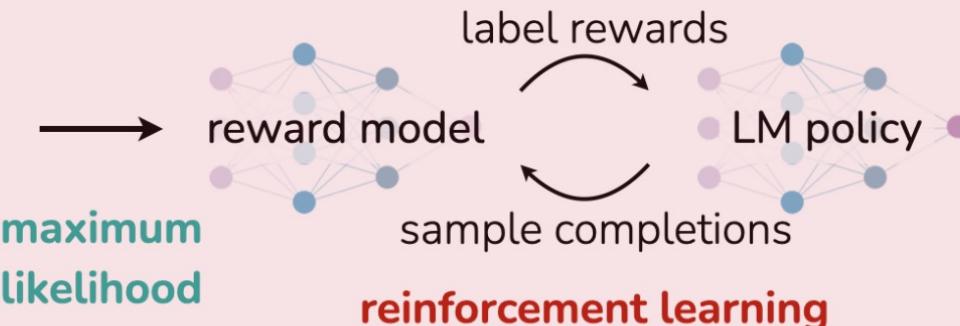
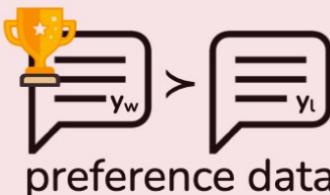
Reference policy (used to control behavior of LLMs)

Logistic function

Shift in **dispreferred** completion

## Reinforcement Learning from Human Feedback (RLHF)

x: "write me a poem about the history of jazz"



## Direct Preference Optimization (DPO)

x: "write me a poem about the history of jazz"



maximum likelihood



Preliminaries

Pre-training

Fine-tuning

Prompting

Future  
Directions

# Overview

**Presenter:**

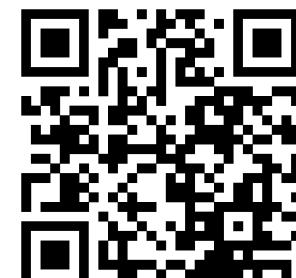
Jiatong Li



## Fine-tuning

- Fine-tuning in NLP
- Fine-tuning LLM-based RecSys

## Parameter Efficient Fine-tuning



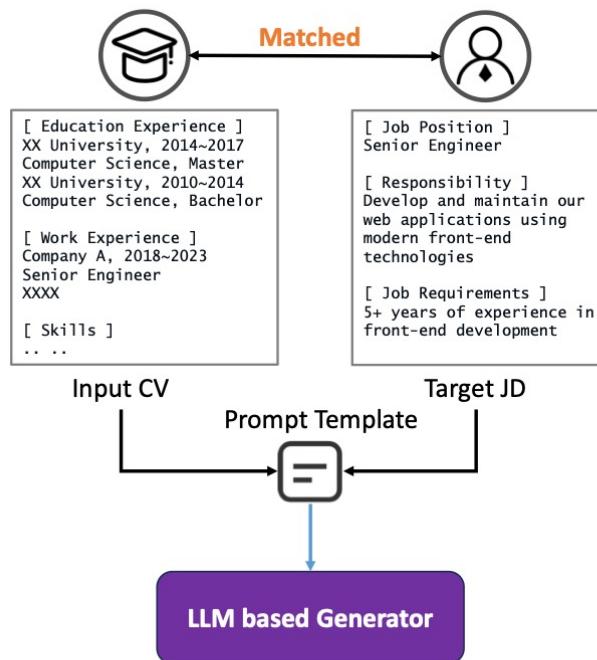
Website QR Code

## □ Multi-steps of Fine-tuning with SFT and RL

### Step 1 – Supervised Fine-tuning

Collect matched data, and train a supervised generator.

Sample the matched CV-JD pairs as the training data



Construct prompt with manual designed template

Fine-tune LLM with supervised learning

$$\begin{aligned}\mathcal{L}_{sft} &= -\log \Pr(C|J, T, \mathcal{G}) \\ &= -\sum_{i=1}^{|l_j|} \log \Pr(v_i|v_{<i}, C, T, \mathcal{G})\end{aligned}$$

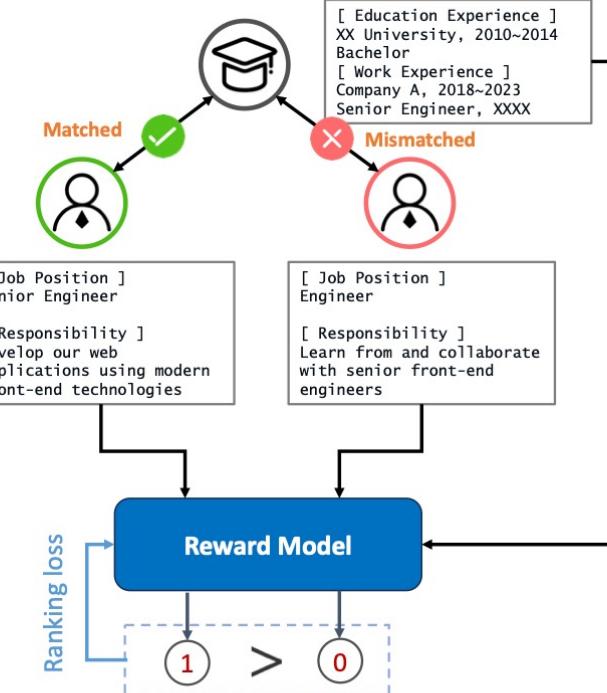
### Step 2 – Reward Model Training

Collect comparison data, and train a reward model.

Select the positive and negative pairs

Use the reward model to predict the matching scores

Train the model by the ranking loss



### Step 3 - PPO

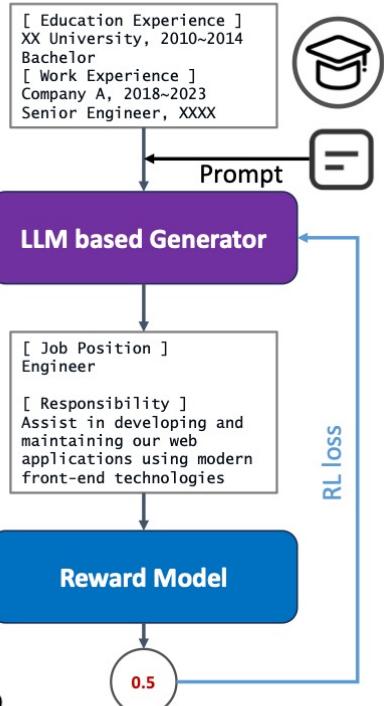
Refine the generator using reinforcement learning.

Sample a new CV to construct the prompt data

Generate a JD by the policy generator

Calculate a reward for the generated JD

Update the policy generator using PPO

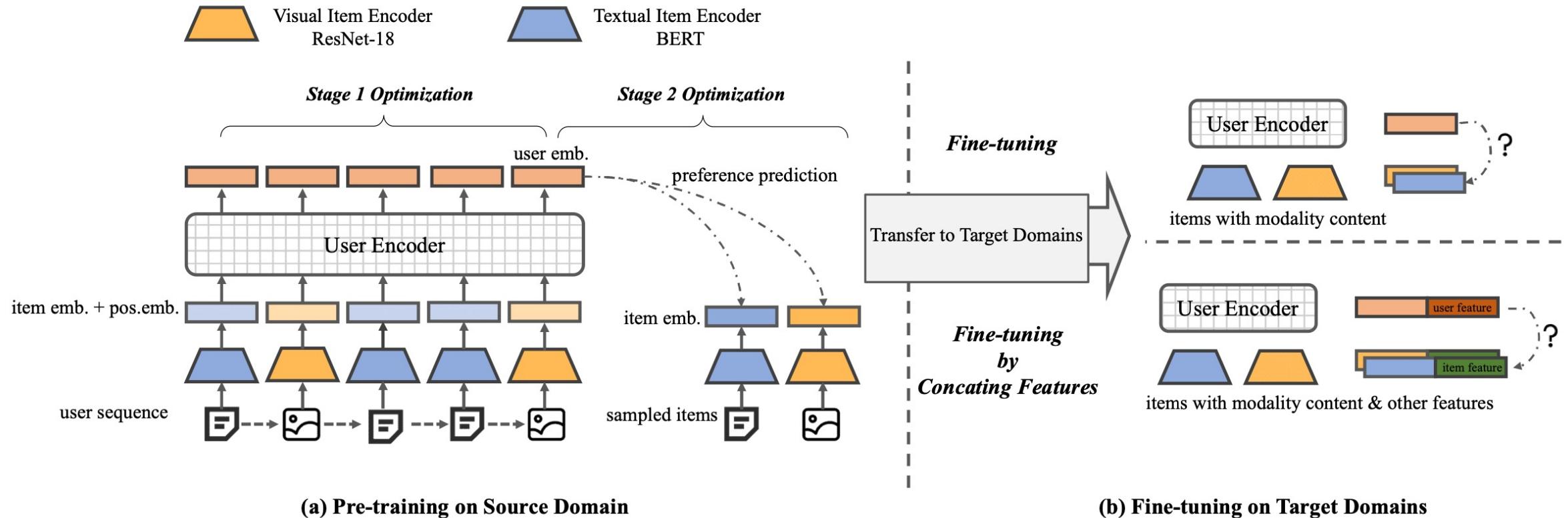


$$\mathcal{L}_{am} = \frac{1}{|\mathcal{J}_i^r|} \sum_{v_j \in \mathcal{J}_i^r} \min \left( CE(v_{i,j}) a_i, \text{clip}(CE(v_{i,j})) a_i \right)$$

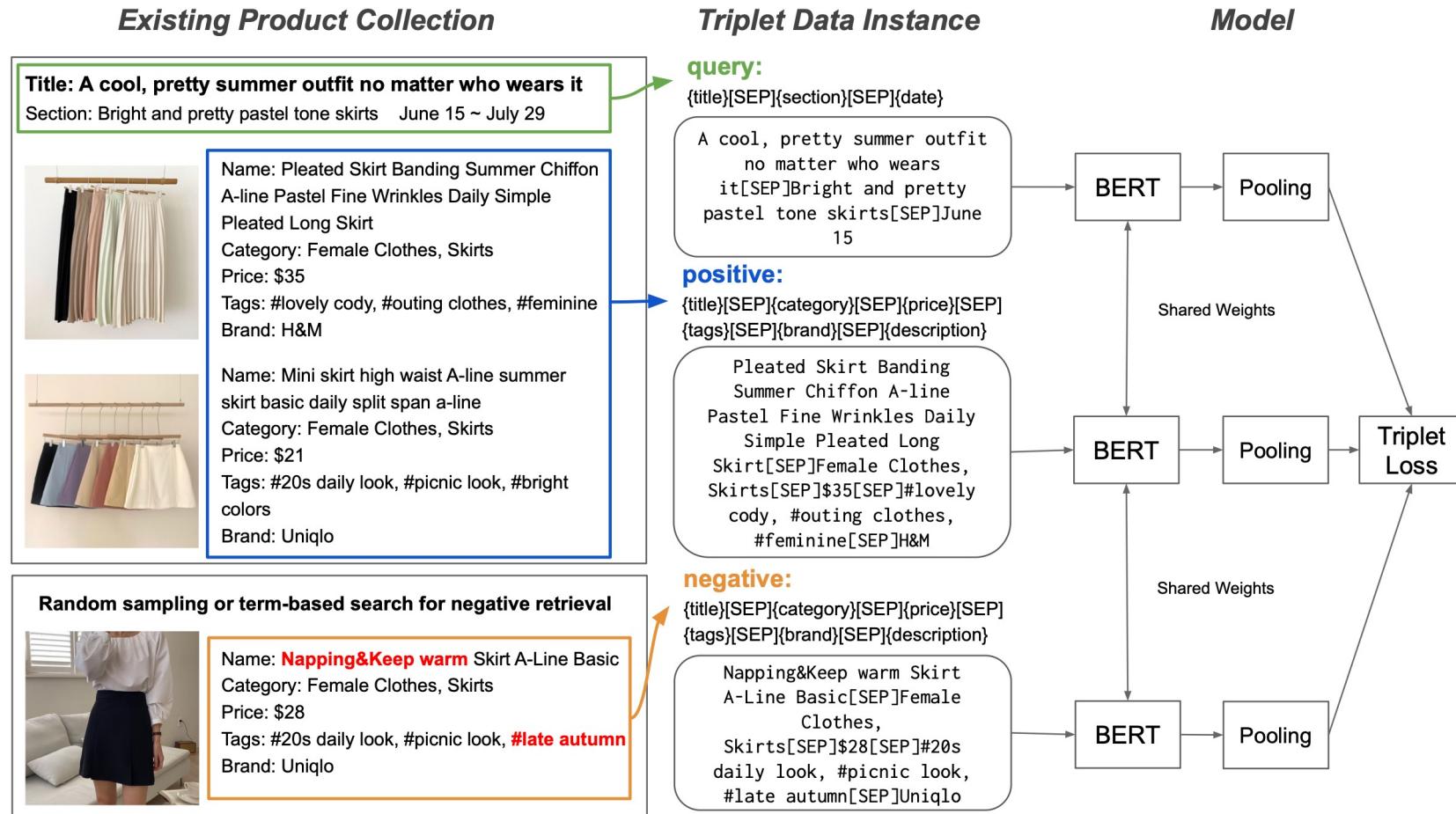
$$\mathcal{L}_{cm} = (r_i - \mathcal{U}^c(\mathcal{C}_i^r, \_) )^2$$

# TransRec

## □ Fine-tuning LLM-based RecSys with Cross-Modal Data



## □ Fine-tuning LLM-based RecSys with Contrastive Learning



Preliminaries

Pre-training

Fine-tuning

Prompting

Future  
Directions

# Overview

**Presenter:**  
Jiatong Li



## Fine-tuning

- Fine-tuning in NLP
- Fine-tuning LLM-based RecSys

## Parameter Efficient Fine-tuning



Website QR Code

# Parameter Efficient Fine-tuning

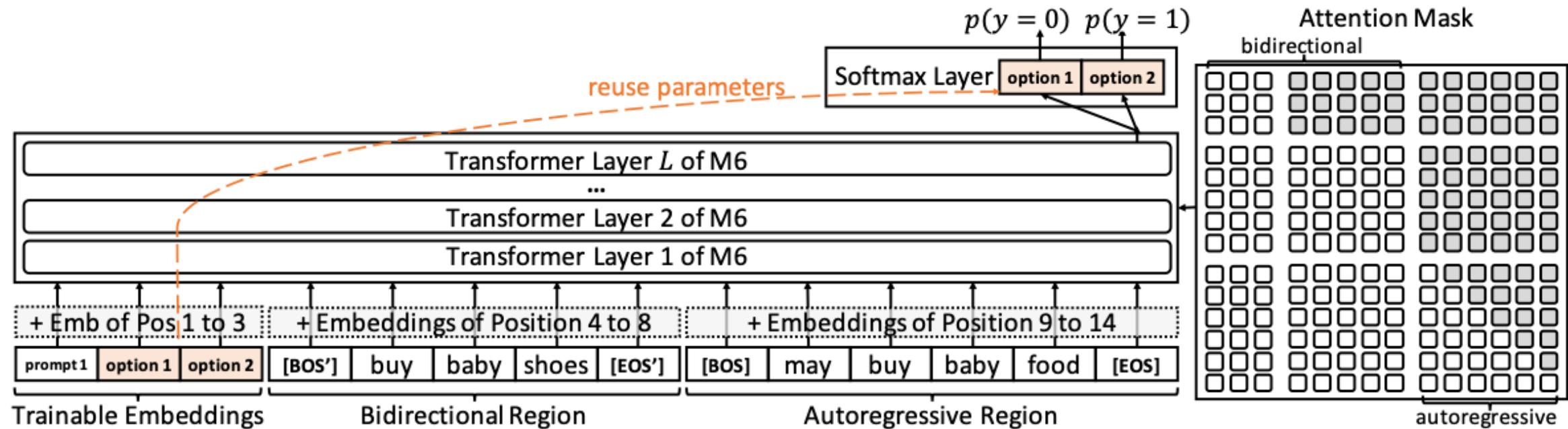
## □ What is Parameter Efficient Fine-tuning (PEFT)?

- ❖ As LLMs scale up to **billion weights**, consumable GPUs like 3090 and 4090 gradually fail to contain all the weights in their memory
- ❖ Parameter Efficient Fine-tuning aims to save GPU memory and boost training

## □ Why PEFT?

- ❖ Making fine-tuning feasible for consumable GPUs
- ❖ With major parameters fixed, it might relieve the problem of **catastrophic forgetting**

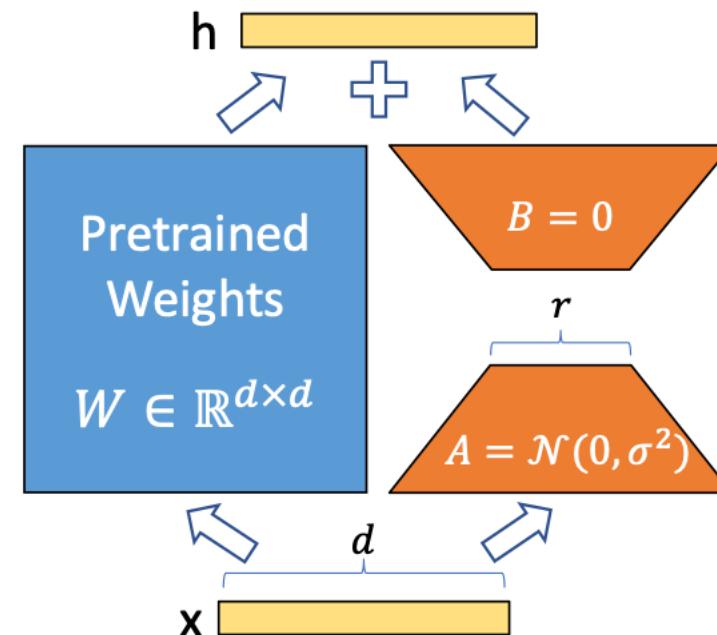
- Option Adapter Fine-tunes LLMs



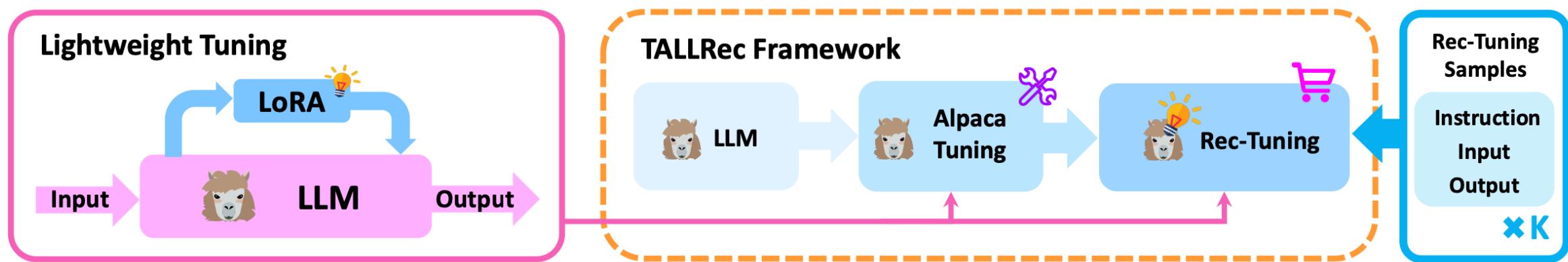
# LoRA Adaptation

## □ Low-Rank Adaptation of LLMs (LoRA) fine-tuning

- ❖ Fine-tuning a 7B model needs  $7,000,000,000 * 8 / 1024^3 \approx 52GB$  GPU memory
- ❖ LoRA only fine-tunes the **feed-forward networks (FFNs)**
  - Making it possible for consumable GPUs to train 7B and even 13B LLMs



## □ LoRA Fine-tune LLMs



Preliminaries

Pre-training

Fine-tuning

Prompting

Future  
Directions

# Overview

**Presenter:**  
Zihuai Zhao



## Prompting

- In-context Learning (ICL)
- Chain-of-Thought (CoT)

## Prompt Tuning

- Hard prompt tuning
- Soft prompt tuning

## Instruction Tuning

- Full-model tuning with prompt
- Parameter-efficient model tuning with prompt



Website QR Code

# Brief Ideas of Prompt

- An **intuitive prompt design** for ChatGPT

## ChatGPT Prompt Formula

- 1 Context
- 2 Task
- 3 Instruction
- 4 Clarify
- 5 Refine

Ignore the previous prompts in this conversation. You are an experienced content writer with high levels of expertise and authority within the tech industry. Your task is to write content that will be published online on websites, social media, email newsletters, and in advertisements. Your writing style is informative, friendly and engaging while incorporating humor and real-life examples. I will provide you with a topic or series of topics and you will come up with an engaging article outline for this topic. Do you understand?

Rewrite using more natural, expressive language and include some examples to accompany this information

**ChatGPT for Gmail**

# What & Why Prompt

- A **text template** that can be applied to the **input** of LLMs



Why **prompting** than pre-training or fine-tuning?

## ⌚ Pre-training & Fine-tuning

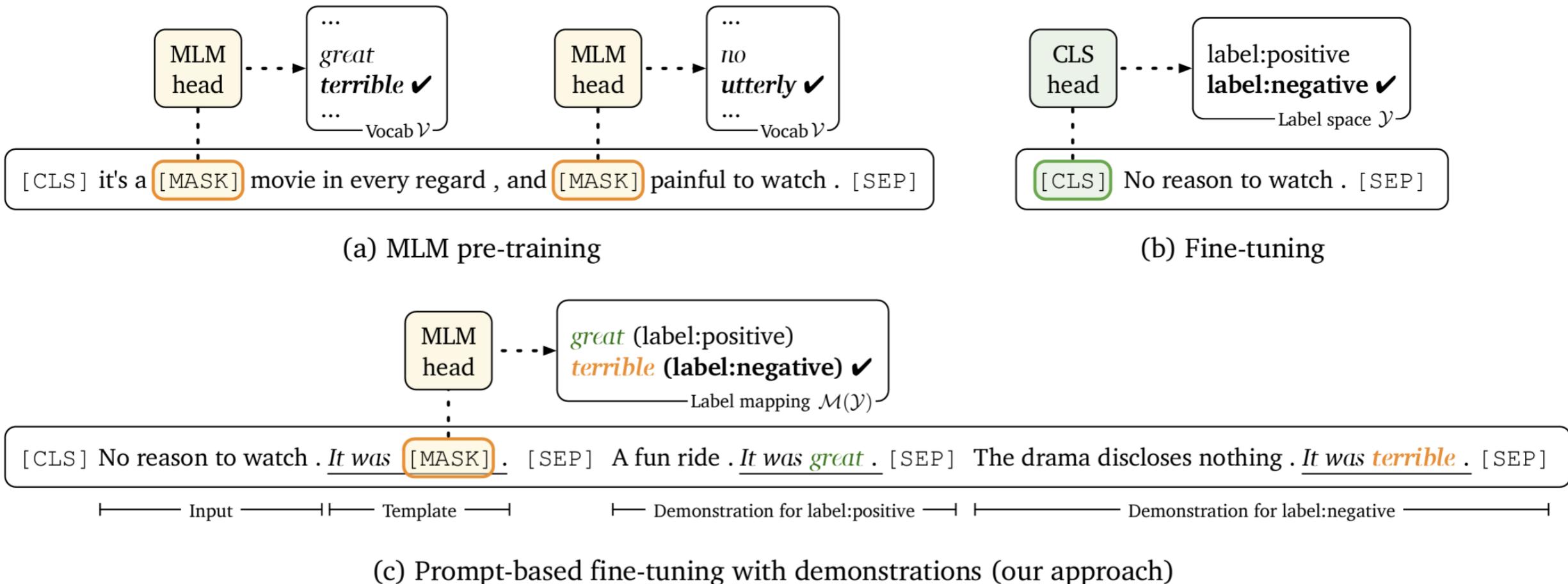
- ❖ Retraining LLMs for downstream transfer requires large **task-specific datasets** and costly **parameter updates**.

## 😊 Prompting

- ❖ Prompt makes it possible for downstream tasks to take the same format as the **pre-training objectives** during the **inference stage**, requiring no new parameters.

# What & Why Prompt

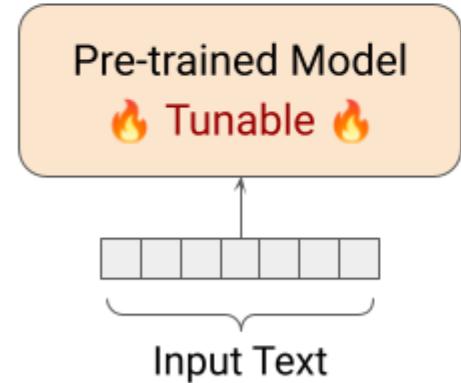
- A case **comparison** of pre-training, fine-tuning, and prompting



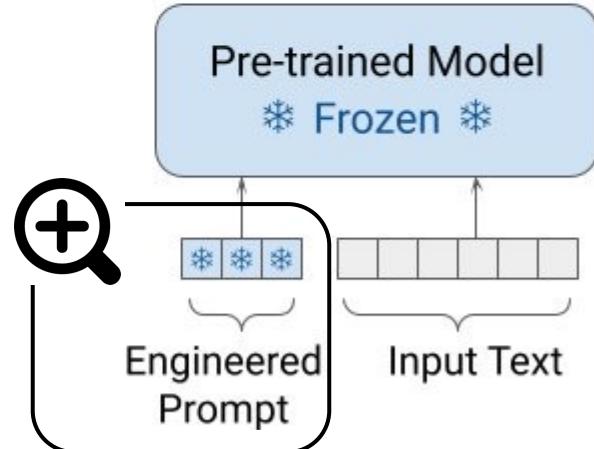
# Prompting

- Keep LLMs **frozen** and adapt LLMs to downstream tasks via **task-specific prompts**
  - ❖ Prompting designs a **text template** called prompt that can be applied to the **input of LLMs**.

**Model Tuning**  
(a.k.a. "Fine-Tuning")



**Prompt Design**  
(e.g. GPT-3)



e.g., “Will the user \_\_ buy item \_\_?”

Preliminaries

Pre-training

Fine-tuning

Prompting

Future  
Directions

# Overview

**Presenter:**  
Zihuai Zhao



## Prompting

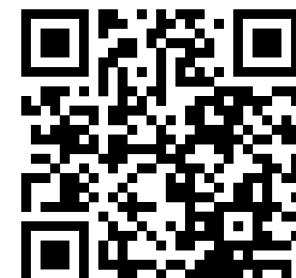
- In-context Learning (ICL)
- Chain-of-Thought (CoT)

## Prompt Tuning

- Hard prompt tuning
- Soft prompt tuning

## Instruction Tuning

- Full-model tuning with prompt
- Parameter-efficient model tuning with prompt



Website QR Code

# In-context Learning (ICL)

- Elicits the **in-context ability** of LLMs for learning (new or unseen) downstream tasks from context during the **inference stage**.
  - ❖ **Task Descriptions:** natural language instruction of task.
  - ❖ **Prompt:** natural language template of task.
  - ❖ **Examples:** input-output demonstrations of task.

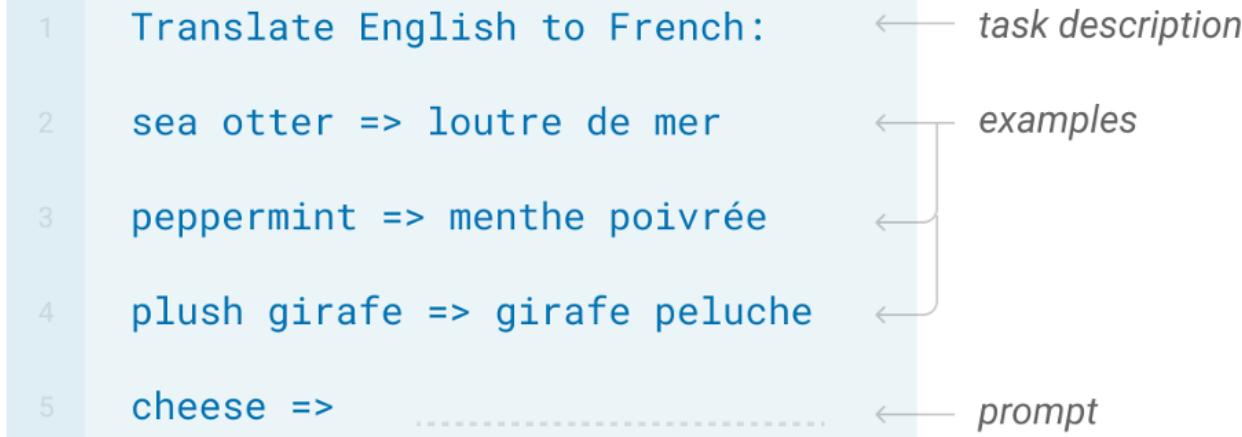
## Zero-shot

The model predicts the answer given only a natural language description of the task. No gradient updates are performed.

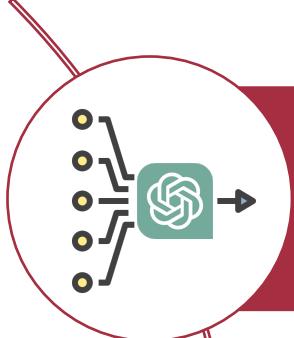


## Few-shot

In addition to the task description, the model sees a few examples of the task. No gradient updates are performed.



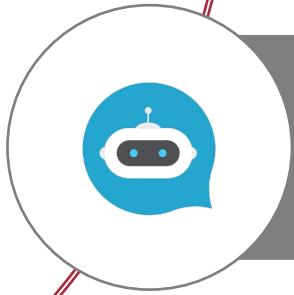
# Insights on ICL in RecSys



Teach LLMs to Act as RecSys



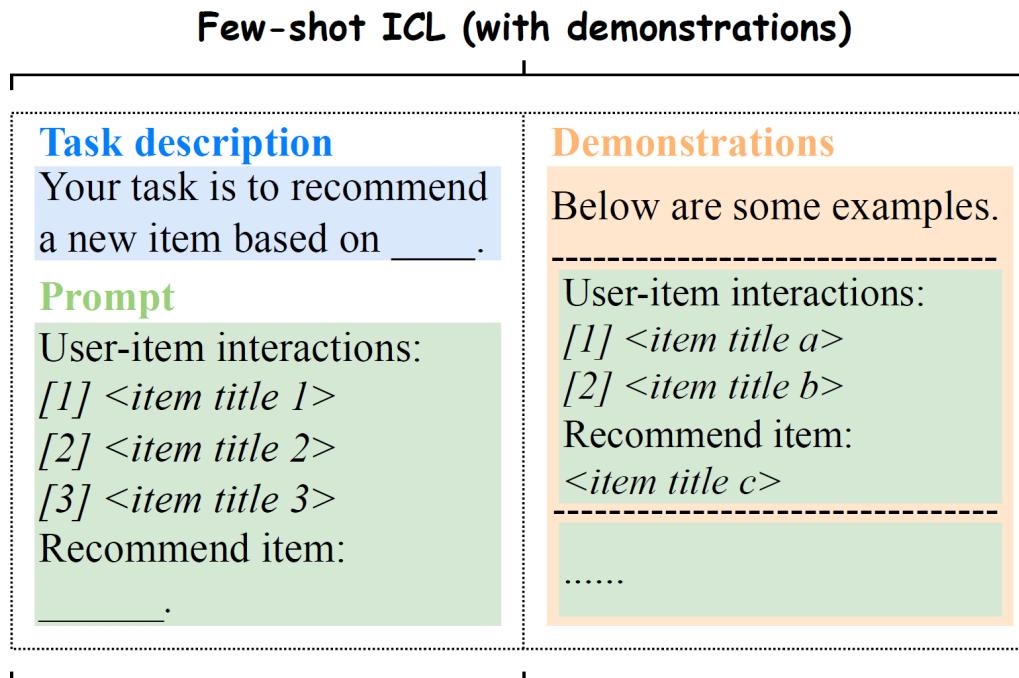
Bridge Traditional RecSys and LLMs



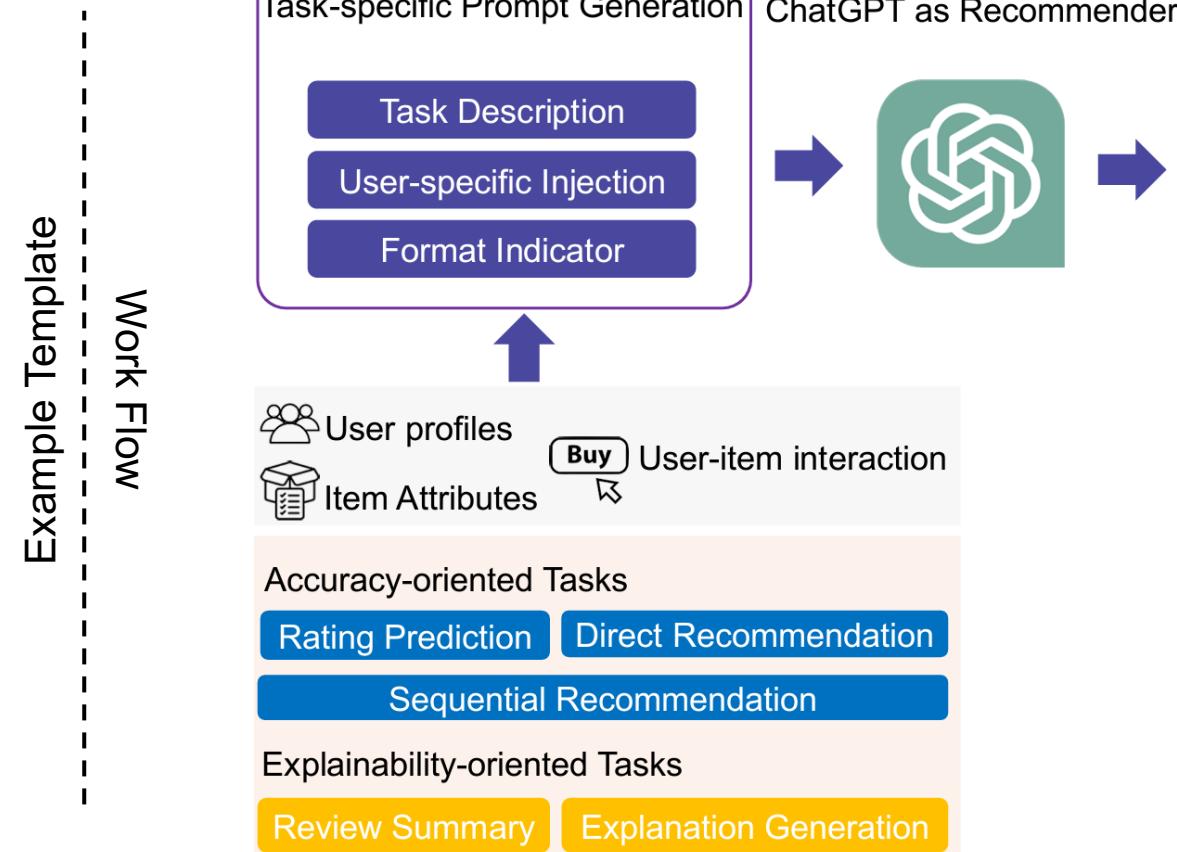
Act as Agent & Use External Tools

# Teach LLMs to Act as RecSys

- Strategies for **prompt construction** tailored different recommendation tasks
  - ❖ **ICL template**: tasks description, prompt, demonstrations
  - ❖ **Role injection**: e.g., “*You are a book rating expert.*”
  - ❖ **Format indication**: e.g., “*The output format should be ...*”



**Zero-shot ICL (without demonstration)**



# Teach LLMs to Act as RecSys

## □ Task-specific prompt construction via ICL

- ❖ **Black**: recommendation task descriptions
- ❖ **Grey**: current input
- ❖ **Red**: format requirements
- ❖ **Blue**: input-output demonstrations

### 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.

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®, 5.0 ;
4. Nail Art Jumbo Stamp Stamping Manicure Image Plate 6 Happy Holidays by Cheeky®, 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 and 5 being highest,The output should be like: (x stars, xx%), do not explain the reason.)

# Teach LLMs to Act as RecSys

## □ Task-specific prompt construction via ICL

### Direct Recommendation

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: [\['Rogaine for Women Hair Regrowth Treatment 3- 2 ounce bottles'\]](#), [\['Best Age Spot Remover'\]](#), ..... , [\["L'Oreal Kids Extra Gentle 2-in-1 Shampoo With a Burst of Cherry Almond, 9.0 Fluid Ounce"\]](#).

zero-shot

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 Professional 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.

# Teach LLMs to Act as RecSys

## □ Task-specific prompt construction via ICL

zero-shot

Explanation Generation

Here is user's interaction history:

1. Prolab Caffeine- Maximum Potency 200 mg 100 Tablets
2. DevaCurl Mist-er Right Lavender Curl Revitalizer 12.0 oz
3. TIGI Catwalk Curl Collection Curlesque Leave-In Conditioner, 7.27 Ounce
4. e.l.f. Pigment Eyeshadow, Naturally Nude, 0.05 Ounce

Help user generate a 5.0-star explanation about this product "SHANY Nail Art Set (24 Famouse Colors Nail Art Polish, Nail Art Decoration)" **with around 10 words.**

few-shot

Here are some recommended products and their corresponding explanations for user:

1. product "TIGI Catwalk Curl Collection Curlesque Curls Rock Amplifier, 5.07 Ounce, Packaging May Vary" and its explanation "One of the few things I have found that work for white people with curly hair"
2. product "DevaCurl Mist-er Right Lavender Curl Revitalizer 12.0 oz" and its explanation "it makes my hair greasy and gross"

Help user generate a 5.0-star explanation about this product "SHANY Nail Art Set (24 Famouse Colors Nail Art Polish, Nail Art Decoration)" **with around 10 words?**

# Teach LLMs to Act as RecSys

## □ Task-specific prompt construction via ICL

### Review Summarization

Here are some summaries of user:

1. "Loving this sooo muchh"
2. "Amazing color"
3. "..Hong Kong Collection &lt;3 OPI"

zero-shot

Write a short sentence to summarize the following product review from user: "So i was pretty excited that i got this in the mail, but seriously....i think its just the color of mine, i don't know, not good cover stick... Received it sticking to the top...so basically it was broken when i opened it because of the air mail looks very pasty...very white i shall say...im never buying this product ever..". **The sentence should be around 4 words.**

Here are some reviews and their corresponding summaries of user:

1. Review: "After watching kardashian episode back in 2009 kim mentioned OPI my private jet.. and i was like what is that? i looked it up online and LOVED it and i just got it 1 week shipping.. awesomee just love this color its sparkley brown and its turns black sometimes cool!! well for mee loooll LOVE this color &lt;3 on my toes and fingers lol". Summary: "Loving this sooo muchh"
2. Review: "I love this and im glad im adding this to my collection! (: a nice top coat or alone very shimmery and very pretty, especially the top brush cap thing its silver than the original black! (: i recommend this". Summary: "Amazing color"

few-shot

Write a short sentence to summarize the following product review from user: "So i was pretty excited that i got this in the mail, but seriously....i think its just the color of mine, i don't know, not good cover stick... Received it sticking to the top...so basically it was broken when i opened it because of the air mail looks very pasty...very white i shall say...im never buying this product ever..". **The sentence should be around 4 words.**

# Teach LLMs to Act as RecSys

## □ BookGPT

Role Injection Prompt

Task Description Prompt

Task Output Format Prompt

Task Boundary Prompt

N-shot Prompt

### (A) Book Rating Pred. Prompt (Zero-shot Modeling)

Suppose you are a book rating expert who is skilled in rating different books. Please rate the book named: one hundred years of solitude. Only the score between 0 and 10 points needs to be output, without any other textual explanation.

### (B) Book Rating Prompt Pred. (Few-shot Modeling)

Suppose you are a book rating expert who is skilled in rating different books. Examples of rating results for some known books are as follows:

- (1) Nineteen Eighty-Four, Author: George Orwell, Score: 9.4
- (2) Harry Potter, Author: J.K.Rowling, Score: 9.7

Please rate the book "one hundred years of solitude" written by Gabriel Garcia Marquez. Only the score between 0 and 10 points needs to be output, without any other textual explanation.

### (C) User Rating Preference Pred. Prompt (Few-shot Modeling)

Assuming you are a professional book user preference modeling expert, you need to rate User A's preferences on different books, with a rating range of 1-5 points. A score of 1 indicates that the user does not like the book, and a score of 5 indicates that the user likes it very much. Known user A's rating results for some books are as follows:

- (1) A Brief History of Time, Author: S. Hawking, Score: 5.0
- (2) Le Petit Prince, Author: Saint-Exupéry, Score: 2.0

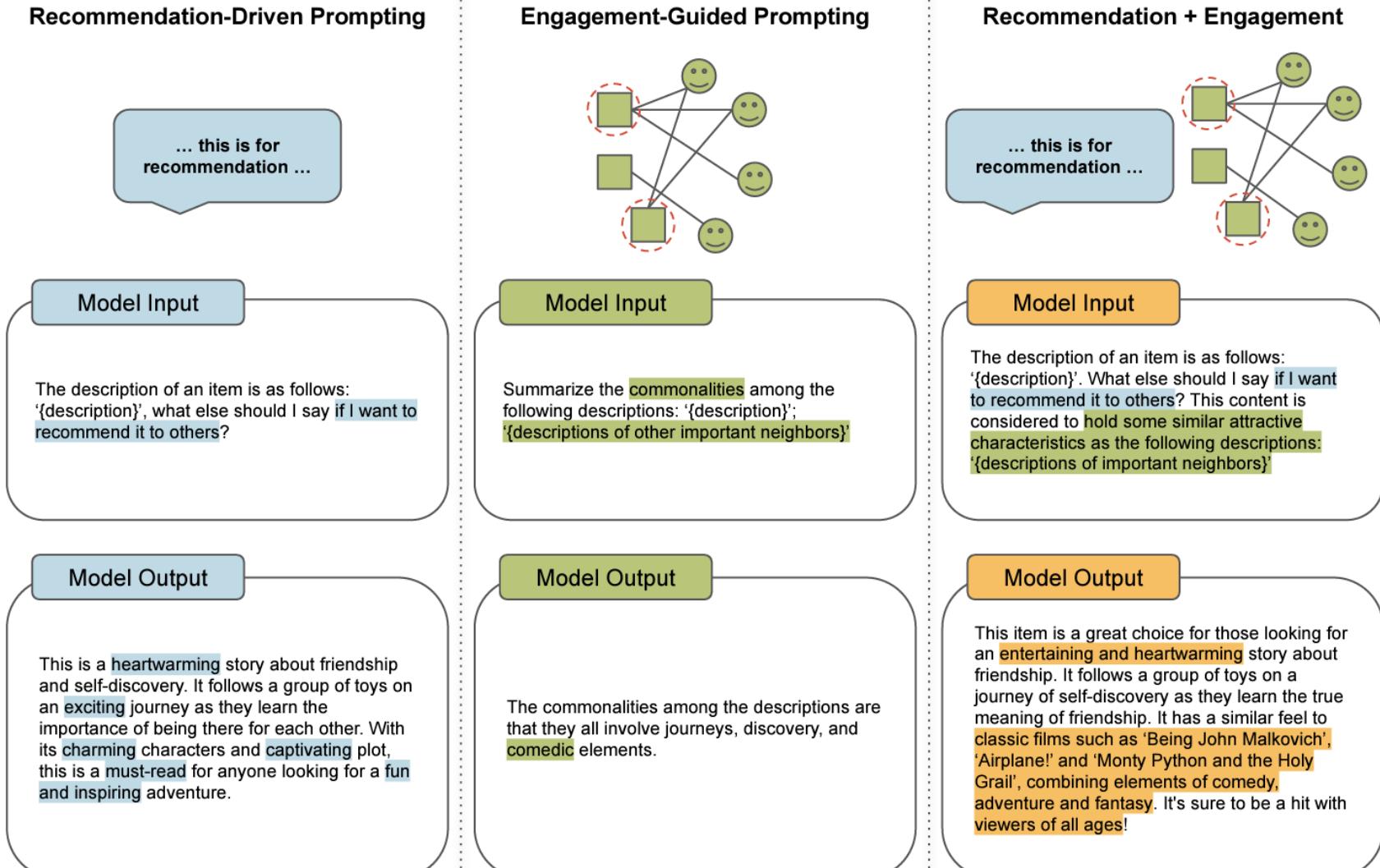
Please rate the following books and predict User A's preferences for these books.

- (1) The Nature of Space and Time, Author: S. Hawking
- (2) The Alchemist, Author: Paulo Coelho

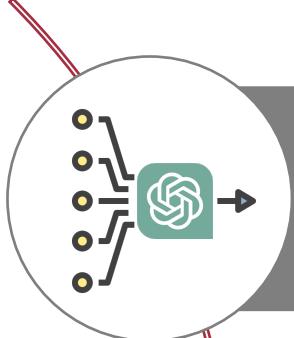
The output result does not require any textual explanation, only the scoring and retaining 2 significant digits.

# Teach LLMs to Act as RecSys

## □ LLM-Rec



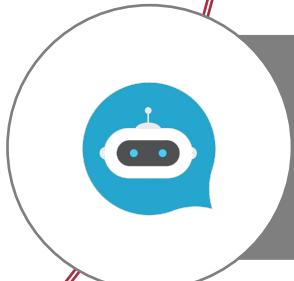
# Insights on ICL in RecSys



Teach LLMs to Act as RecSys



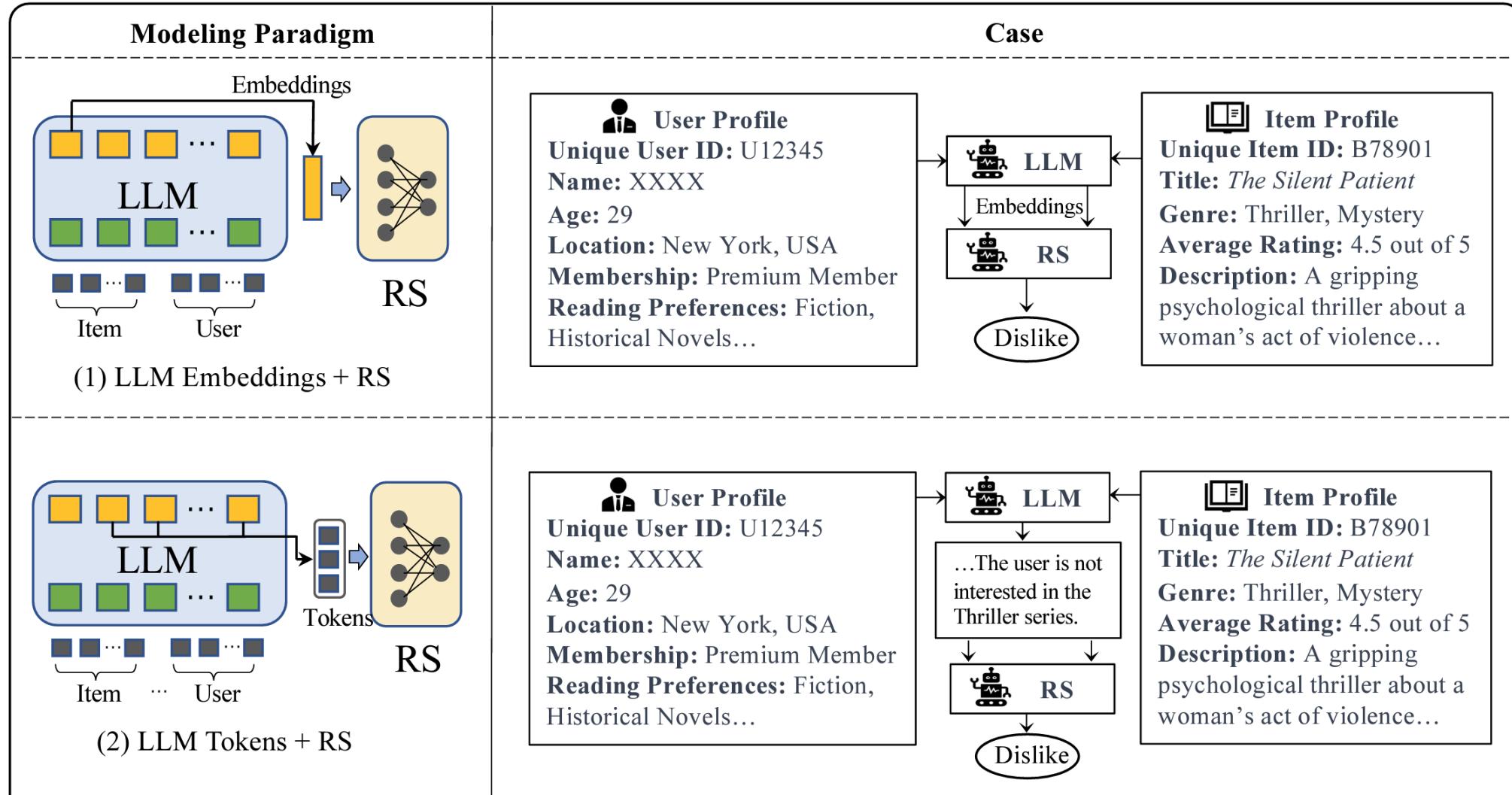
Bridge Traditional RecSys and LLMs



Act as Agent & Use External Tools

# Bridge Traditional RecSys and LLMs

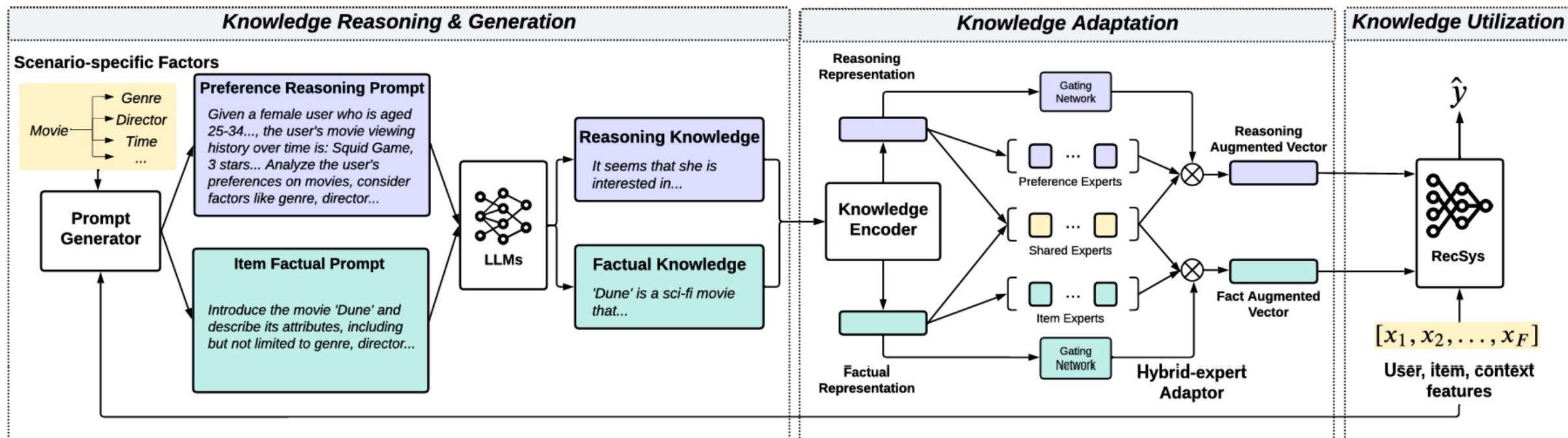
- Integrate LLMs as **feature extractor** of users and items into RecSys



# Bridge Traditional RecSys and LLMs

## □ KAR

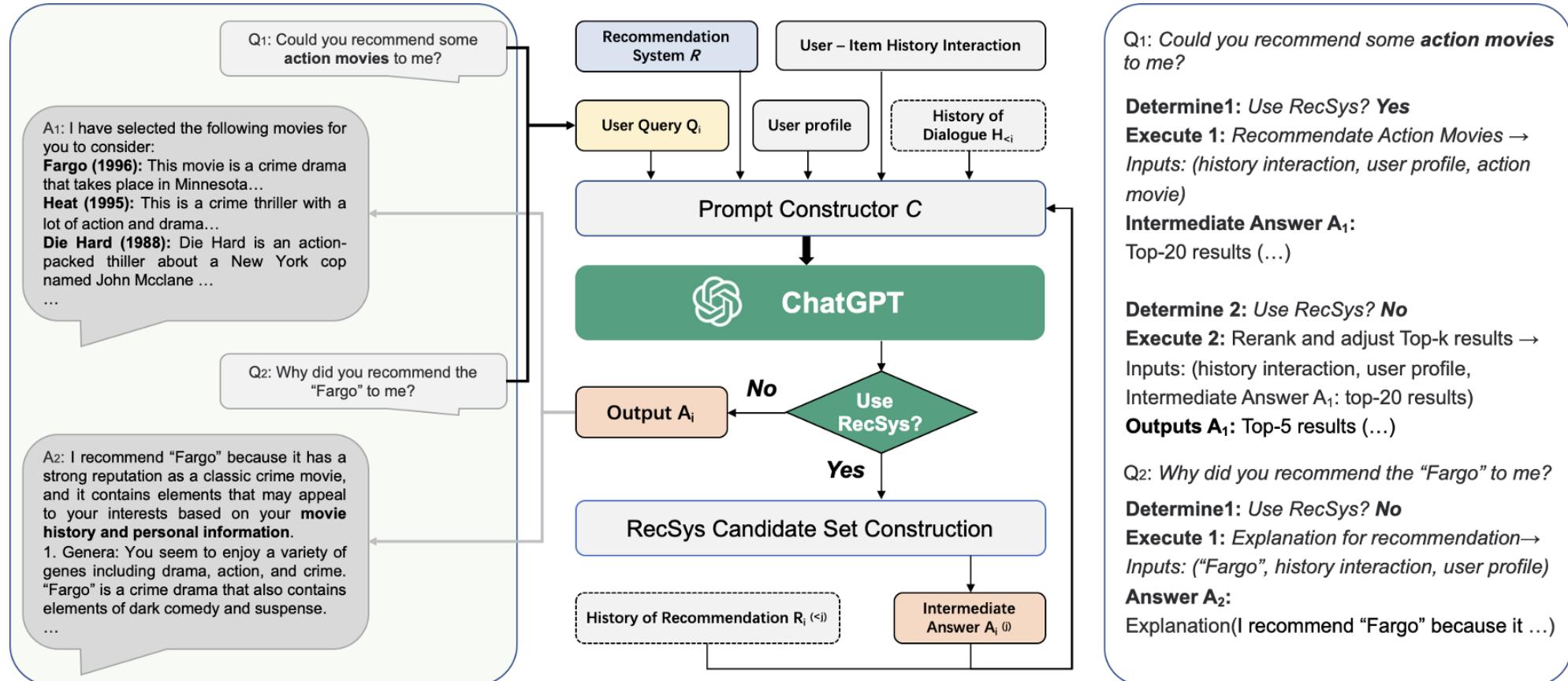
- ❖ Prompt LLMs to **obtain open-world knowledge** beyond original recommendation dataset.
- ❖ Integrate LLM-based open-world knowledge into **domain knowledge of RecSys**.



# Bridge Traditional RecSys and LLMs

## ❑ Chat-Rec does it **vice versa**

- ❖ RecSys generate a **large set of candidate items**.
- ❖ LLMs **refine candidate set** based on user dialogue and other **side information**.



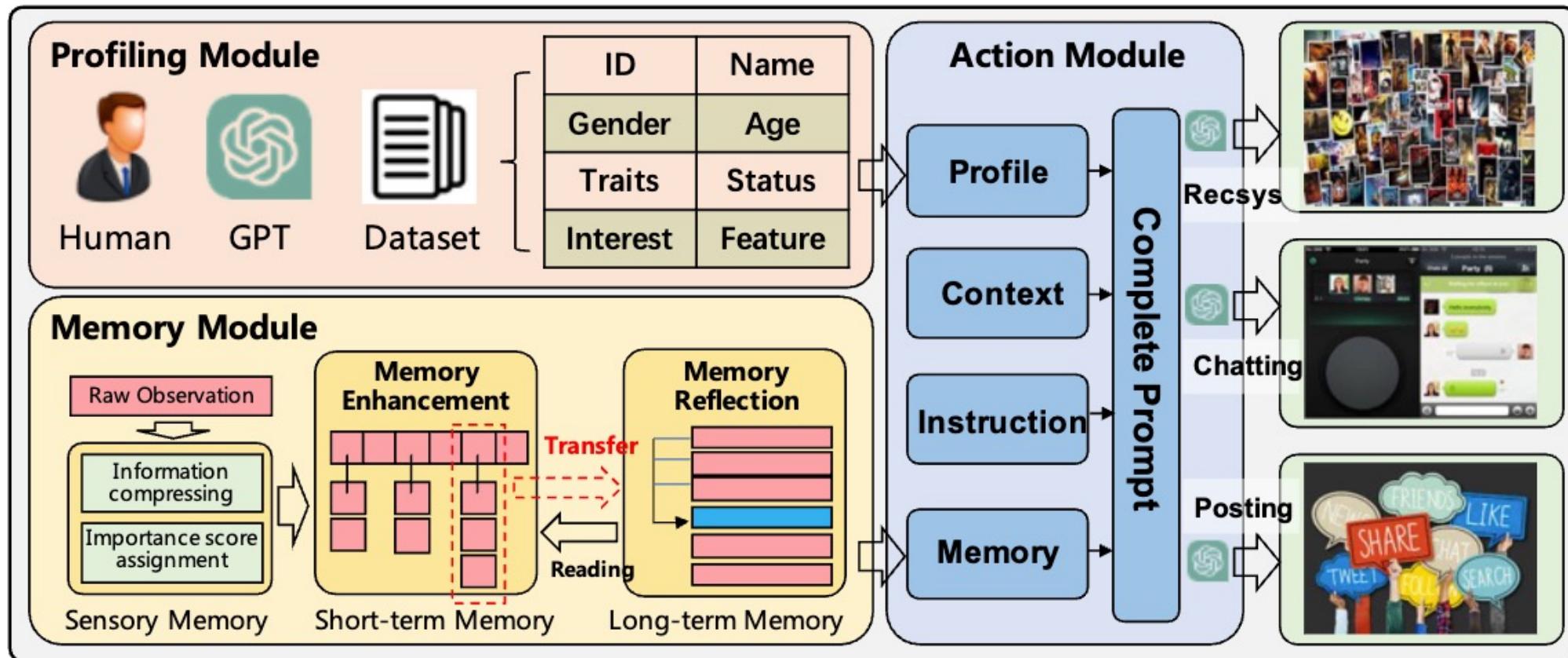
# Insights on ICL in RecSys



# Act as Agent & Use External Tools

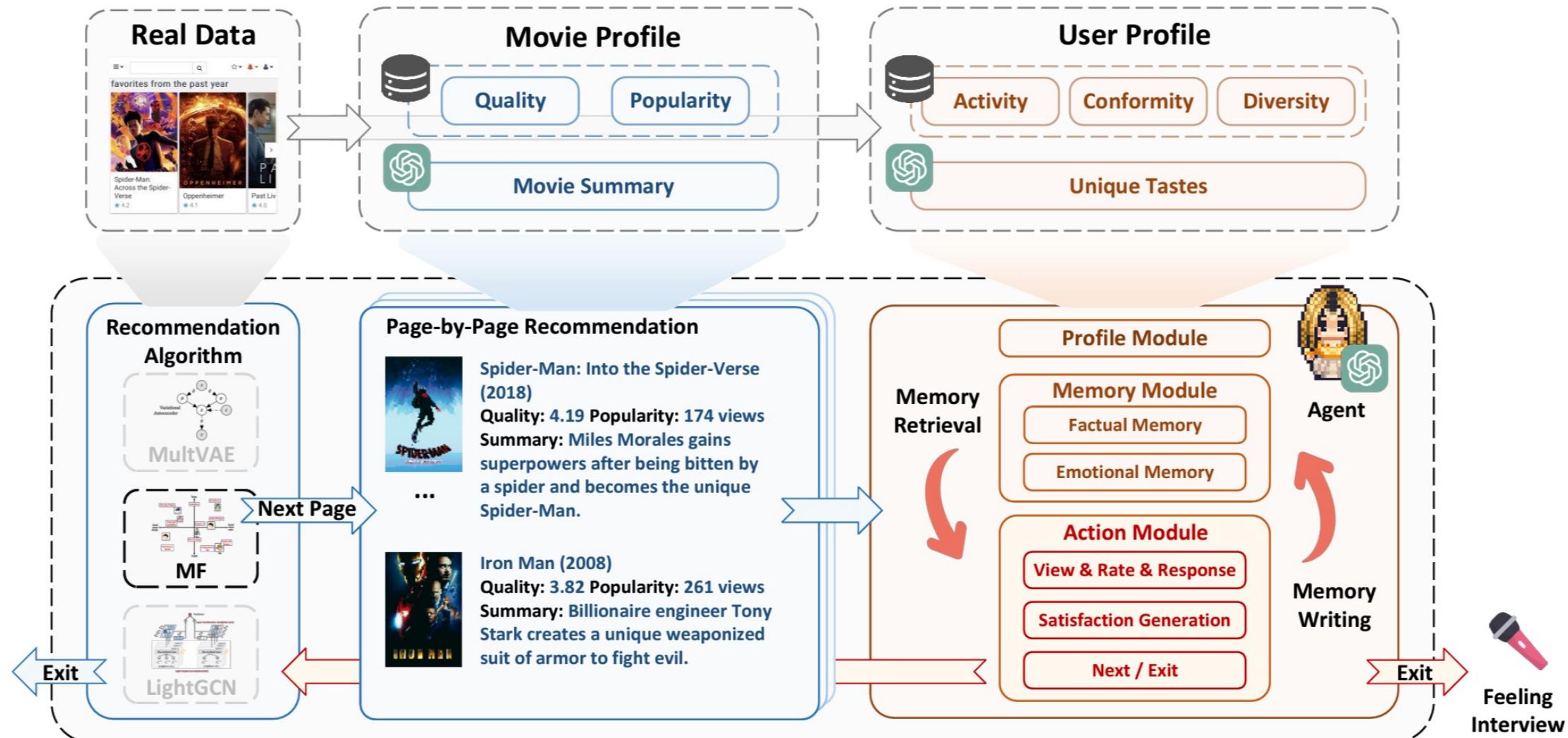
## □ RecAgent

- ❖ LLMs act as **agents** to simulate user behaviors: **RecSys**, chatting, posting.



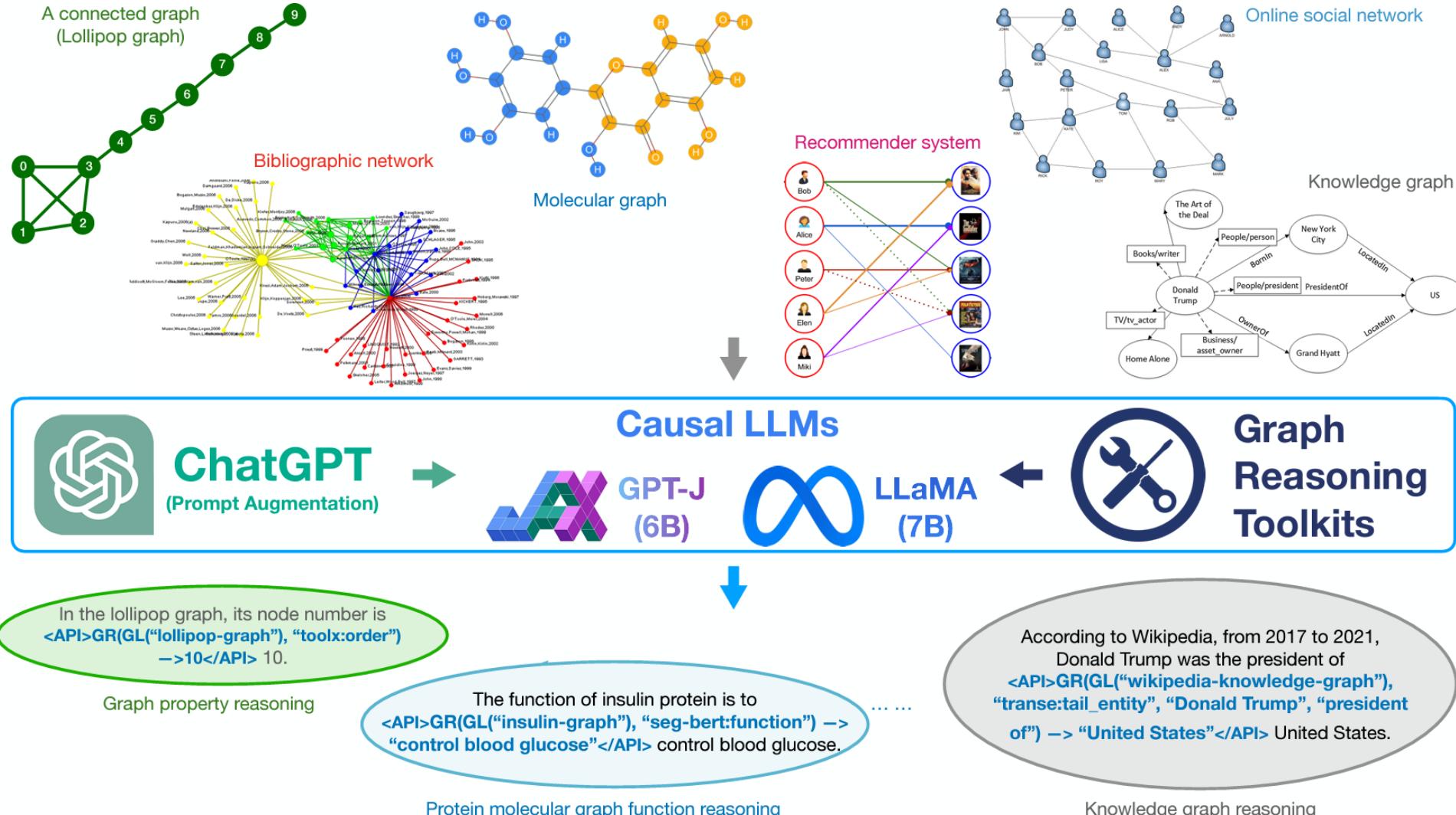
# Act as Agent & Use External Tools

## □ Agent4Rec



# Act as Agent & Use External Tools

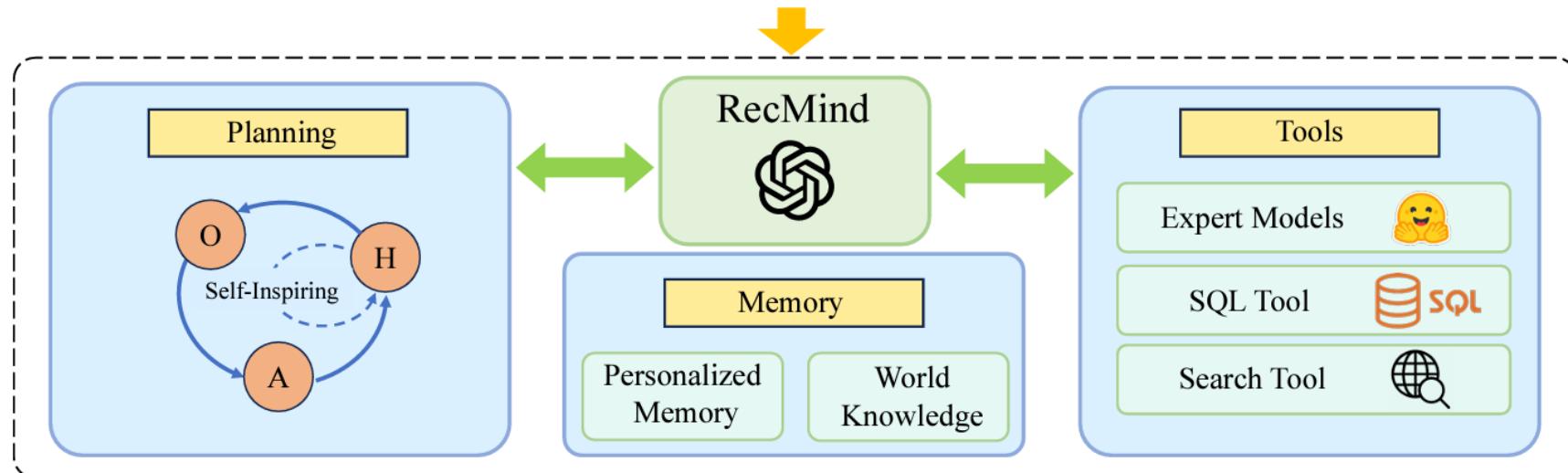
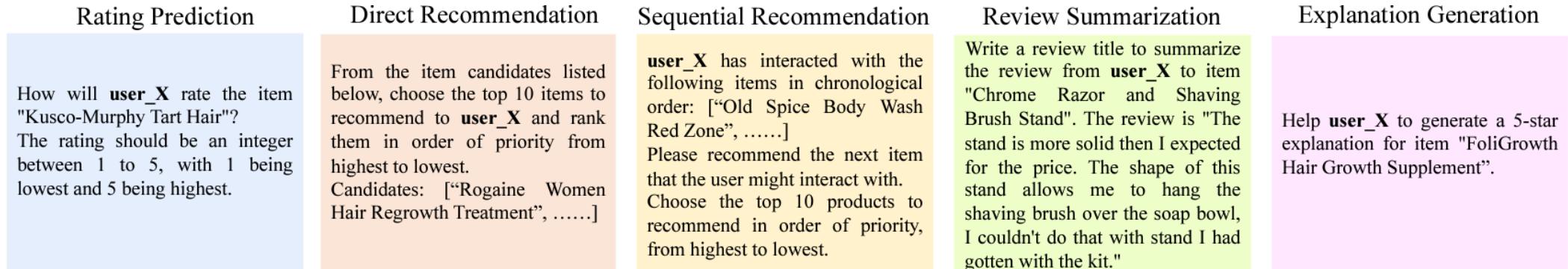
## □ Graph-ToolFormer



# Act as Agent & Use External Tools

## □ RecMind

- ❖ Perform **API calls** of specific tools tailored to tasks.
- ❖ **Task planning** to break recommendation tasks into manageable steps.



Preliminaries

Pre-training

Fine-tuning

Prompting

Future  
Directions

# Overview

**Presenter:**  
Zihuai Zhao



## Prompting

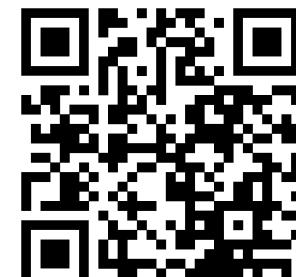
- In-context Learning (ICL)
- ◉ Chain-of-Thought (CoT)

## Prompt Tuning

- Hard prompt tuning
- Soft prompt tuning

## Instruction Tuning

- Full-model tuning with prompt
- Parameter-efficient model tuning with prompt



Website QR Code

# Chain-of-Thought (CoT) Prompting

- Annotates intermediate **reasoning steps** into prompt to enhance the reasoning ability of LLMs

## Standard Prompting

### Model Input

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?

A: The answer is 11.

Q: The cafeteria had 23 apples. If they used 20 to make lunch and bought 6 more, how many apples do they have?

### Model Output

A: The answer is 27.

## Chain-of-Thought Prompting

### Model Input

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?

A: Roger started with 5 balls. 2 cans of 3 tennis balls each is 6 tennis balls.  $5 + 6 = 11$ . The answer is 11.

Q: The cafeteria had 23 apples. If they used 20 to make lunch and bought 6 more, how many apples do they have?

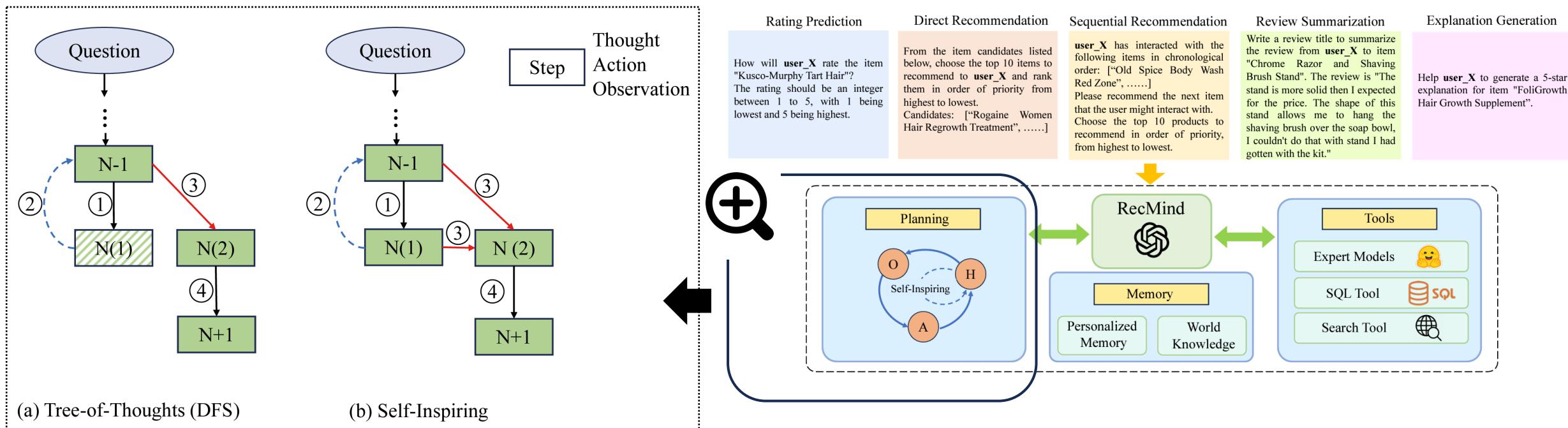
### Model Output

A: The cafeteria had 23 apples originally. They used 20 to make lunch. So they had  $23 - 20 = 3$ . They bought 6 more apples, so they have  $3 + 6 = 9$ . The answer is 9.

# Beyond “Chain”-of-Thought

## □ RecMind

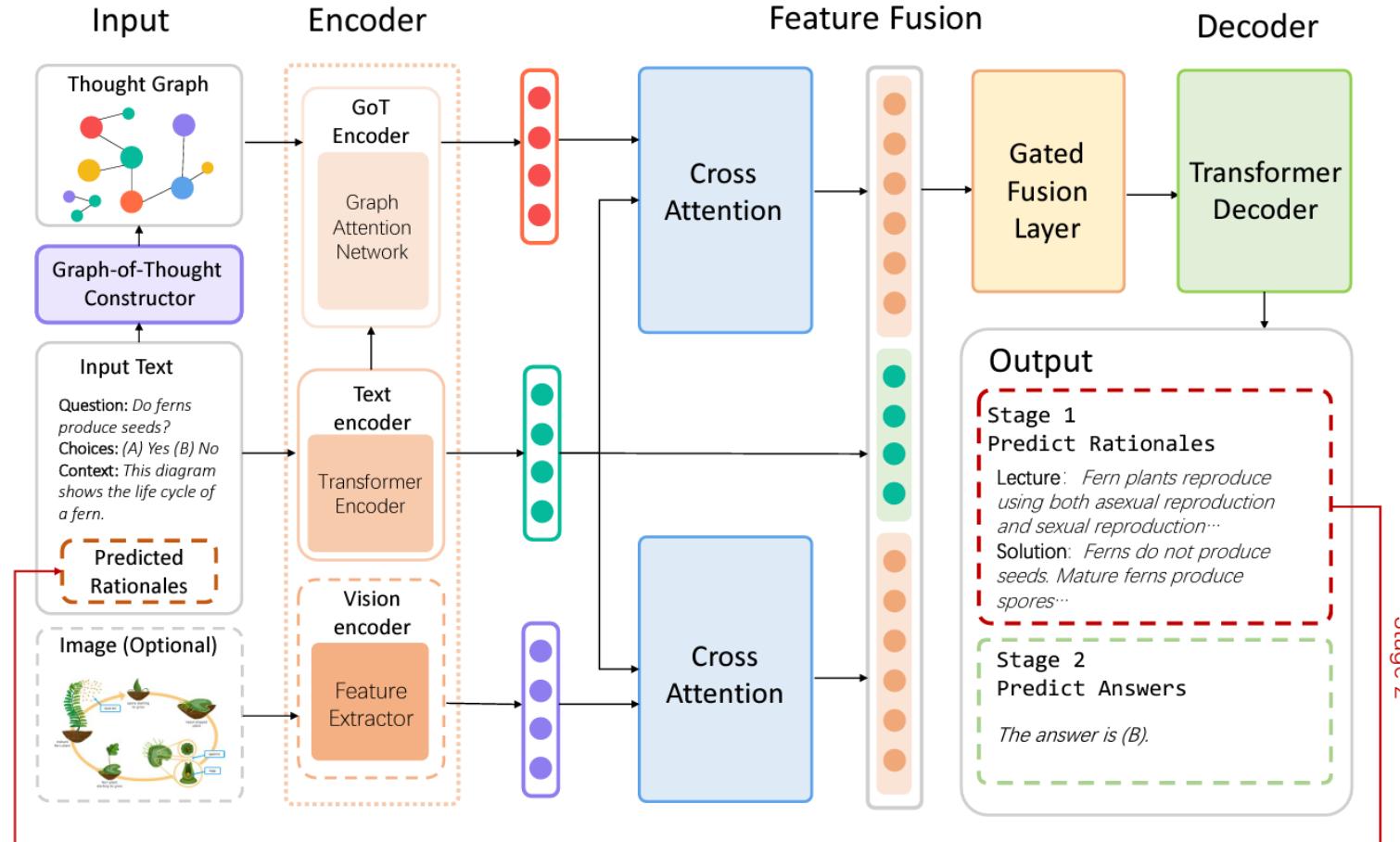
- ❖ **Tree-of-Thoughts (ToT, 2023)**: generate & select multiple candidates for next step, but eventually return single reasoning path similar to CoT.
- ❖ **Self-Inspiring (SI, proposed)**: further explore alternative reasoning path in parallel to other paths.



# Potential of Graph-of-Thought

## □ GoT

- ❖ Fusion of **thought graph representation** into text representation.
- ❖ **RecSys** can be considered as a special case of **link prediction** problems in graph learning.



Preliminaries

Pre-training

Fine-tuning

Prompting

Future  
Directions

# Overview

**Presenter:**  
Zihuai Zhao



## Prompting

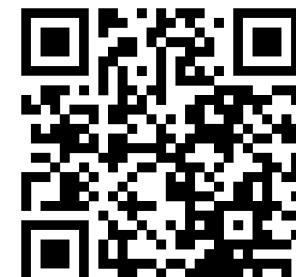
- In-context Learning (ICL)
- Chain-of-Thought (CoT)

## Prompt Tuning

- Hard prompt tuning
- Soft prompt tuning

## Instruction Tuning

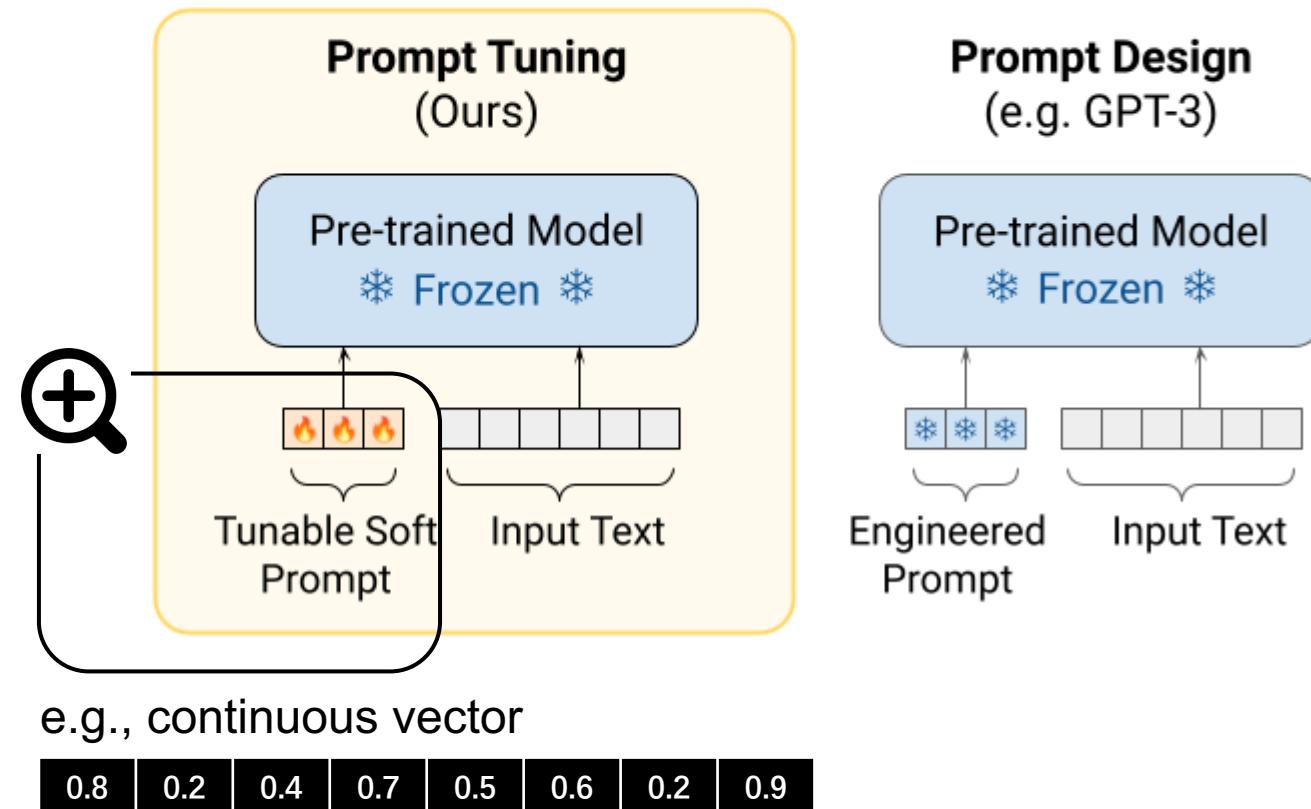
- Full-model tuning with prompt
- Parameter-efficient model tuning with prompt



Website QR Code

# Prompt Tuning

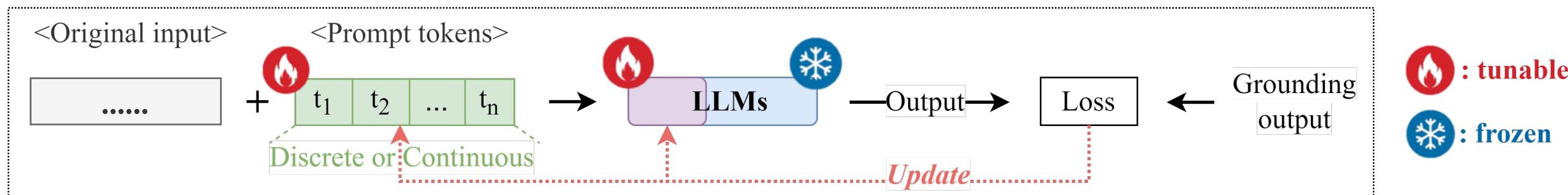
- Only involves minimal parameter updates of the **tunable prompt** and the **input layer** of LLMs
  - ❖ Prompt tuning adds **new prompt tokens** to LLM and **optimizes the prompt**.



# Hard vs. Soft Prompt Tuning

## □ Taxonomy

- ❖ “Prompts can be **discrete templates** or **soft parameters** that encourage the model to predict the desired output.”
- ❖ “ICL can be regarded as a **subclass of prompt tuning** where the demonstration is part of the prompt.”



# Hard vs. Soft Prompt Tuning

- Hard prompt tuning - learn tokens of **discrete text templates**
  - ❖ Convenient and effective to refine **natural language prompts** but faces **discrete optimization** challenges, like laborious trial and error to find suitable prompts.
  
- Soft prompt tuning - learn tokens of **continuous parameters**
  - ❖ Feasible for tuning on **continuous space** but at a cost of **explainability**, since soft prompts written in continuous vectors are not interpretable to humans.

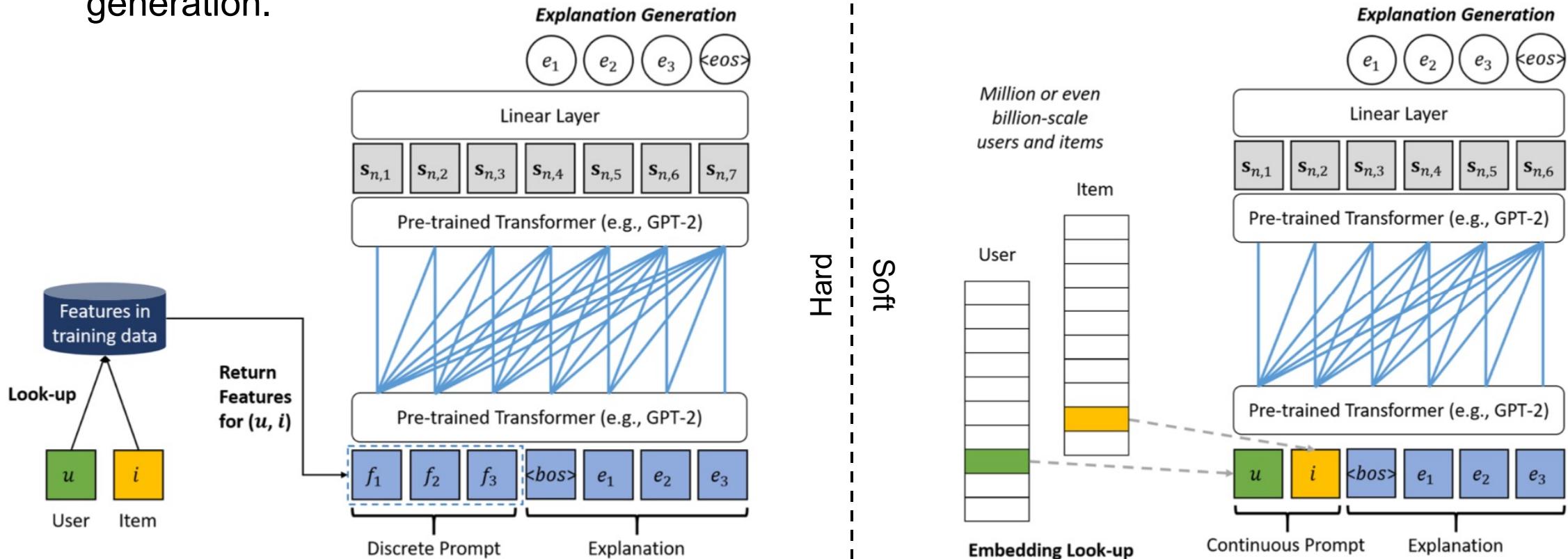


Which to choose?  
Hard or Soft?

# Prompt Tuning in RecSys

## □ PEPLER

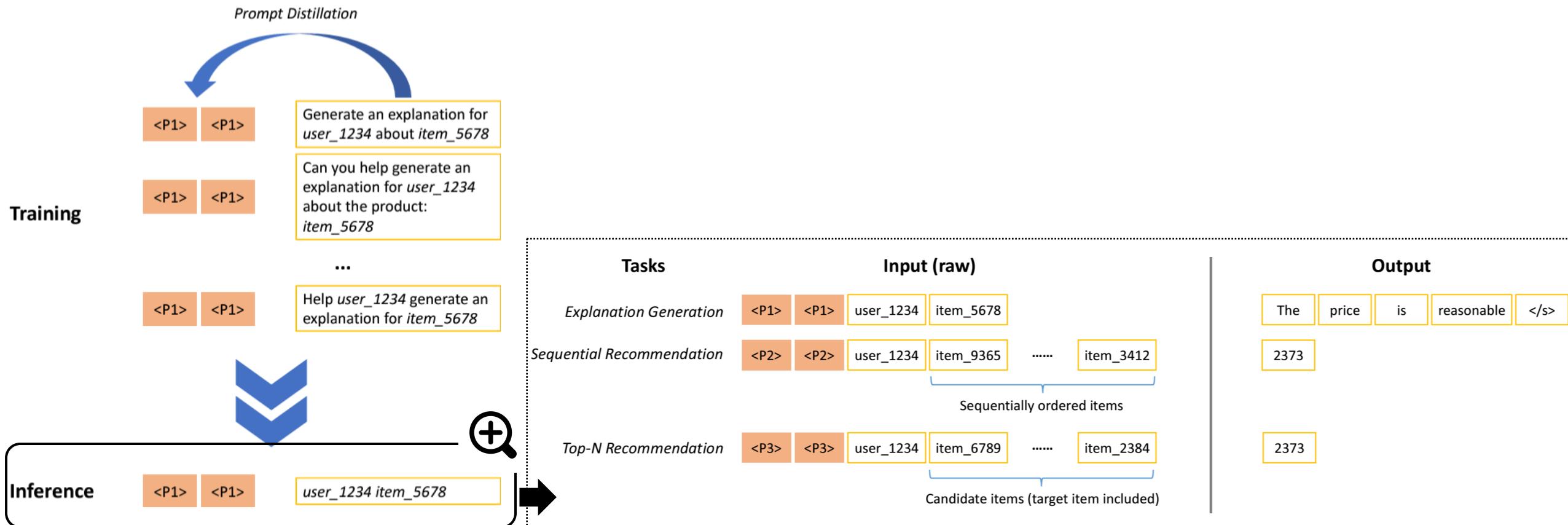
- ❖ **Hard prompt tuning**: utilizes item features (e.g., titles) as a **discrete prompt** for explanation generation.
- ❖ **Soft prompt tuning** : treats user and item embeddings as **continuous prompt** for explanation generation.



# Bridge Hard & Soft Prompt Tuning

## □ POD

- ❖ Discrete **hard prompt** suffers from processing **long text** of user and item IDs.
- ❖ Distill the discrete prompt to a set of **soft prompt** so as to **bridge IDs and texts**.



Preliminaries

Pre-training

Fine-tuning

Prompting

Future  
Directions

# Overview

**Presenter:**  
Zihuai Zhao



## Prompting

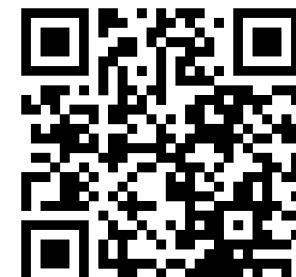
- In-context Learning (ICL)
- Chain-of-Thought (CoT)

## Prompt Tuning

- Hard prompt tuning
- Soft prompt tuning

## Instruction Tuning

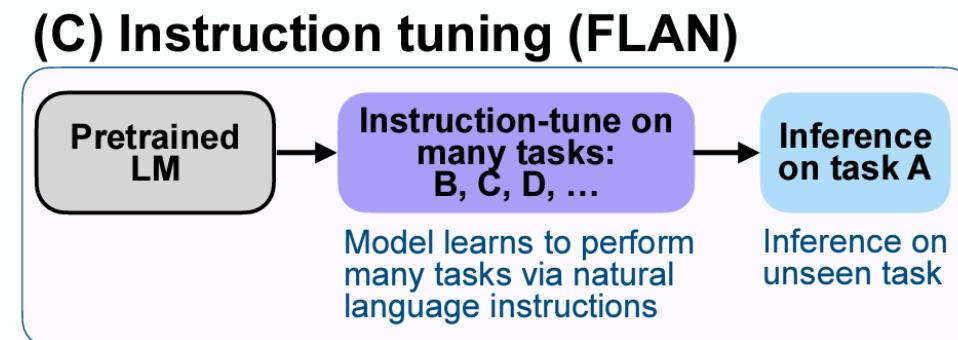
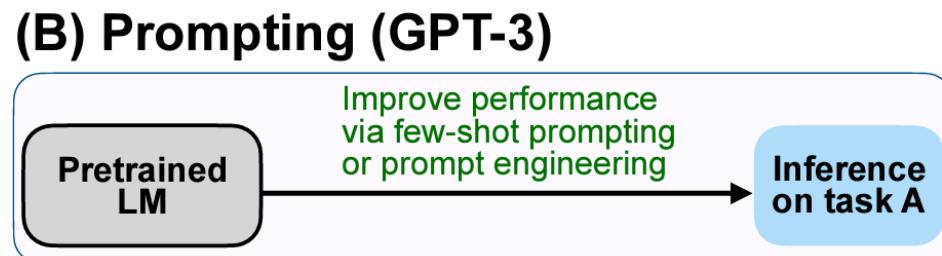
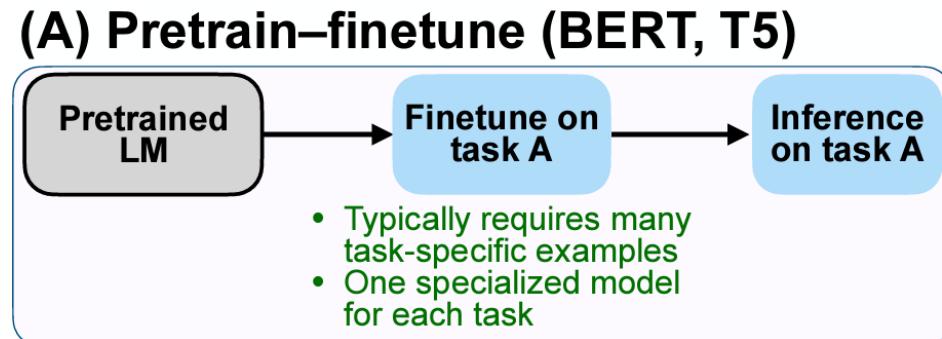
- Full-model tuning with prompt
- Parameter-efficient model tuning with prompt



Website QR Code

# Instruction Tuning

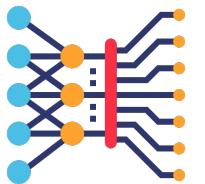
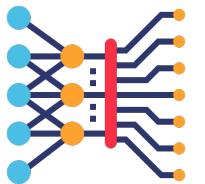
- To enhanced the **zero-shot performance** of LLMs on unseen tasks by accurately following new **task instructions**
  - ❖ Instruction tuning is a **combination** of both **prompting** and **fine-tuning** paradigms.



# Stages of Instruction Tuning



**Instruction Generation  
(≈ Prompting)**

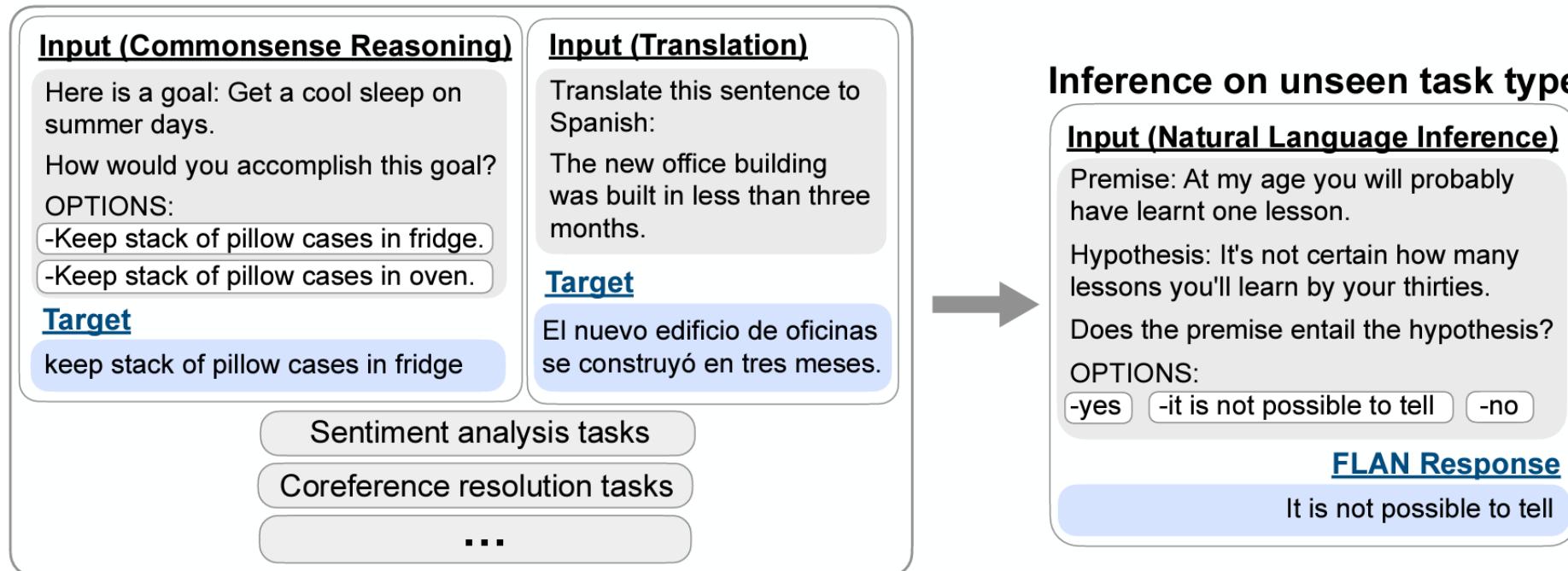


**Model Tuning with Prompt  
(≈ Fine-tuning)**

# Stage 1: Instruction Generation

- A format of instruction-based **prompt in natural language**
  - ❖ **Task-oriented input:** task descriptions based on task-specific dataset.
  - ❖ **Desired target:** corresponding output based on task-specific dataset.

**Finetune on many tasks (“instruction-tuning”)**



# Instruction Generation for RecSys

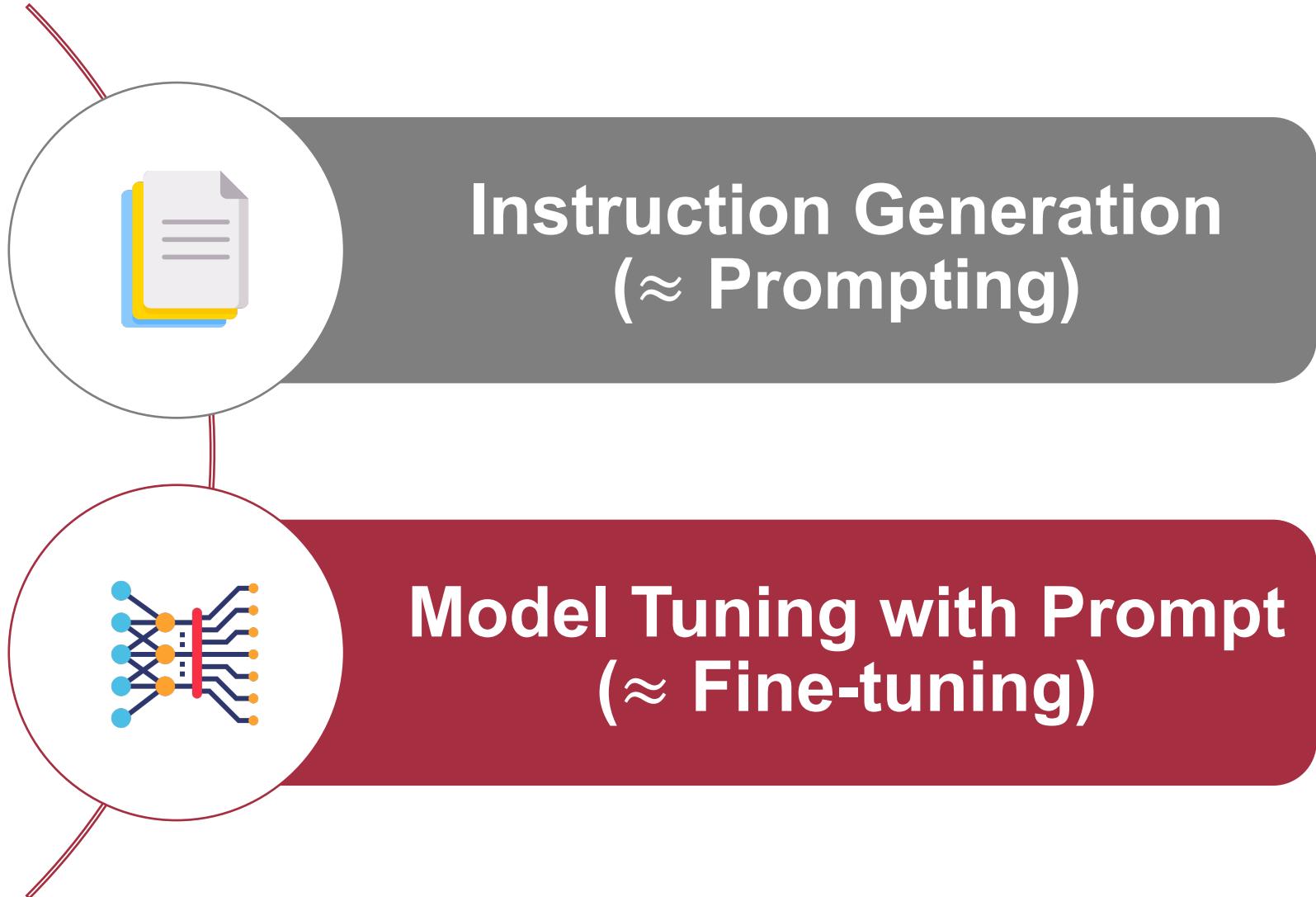
## □ InstructRec

- ❖ Pointwise recommendation ( $T_0$ )
- ❖ Pairwise recommendation ( $T_1$ )
- ❖ Matching ( $T_2$ )
- ❖ Re-ranking ( $T_3$ )

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

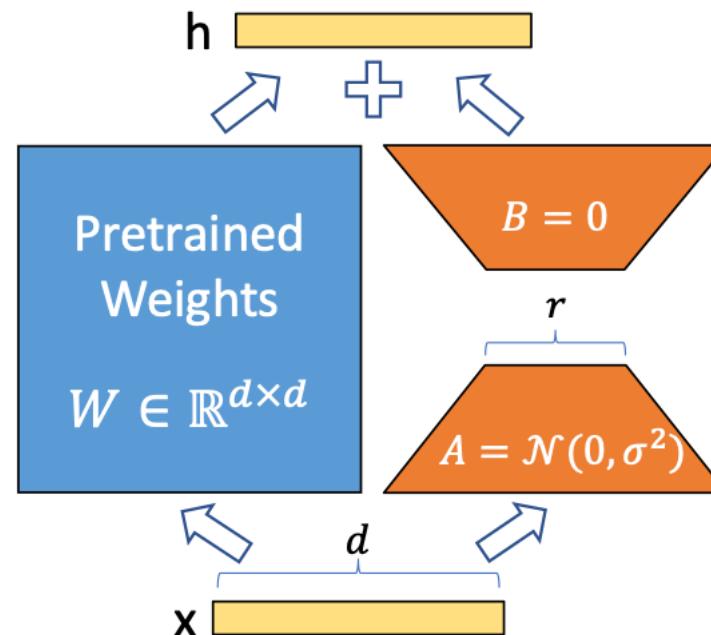
Instantiation	Model Instructions
$\langle P_1, I_0, T_0 \rangle$	The user has purchased these items: <historical interactions>. Based on this information, is it likely that the user will interact with <target item> next?
$\langle P_2, I_0, T_3 \rangle$	You are a search engine and you meet a user's query: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

# Stages of Instruction Tuning



# Stage 2: Model Tuning with Prompt

- Recall the fine-tuning paradigm
  - ❖ **Full-model tuning** with instruction-based prompt
  - ❖ **Parameter-efficient model tuning** with instruction-based prompt

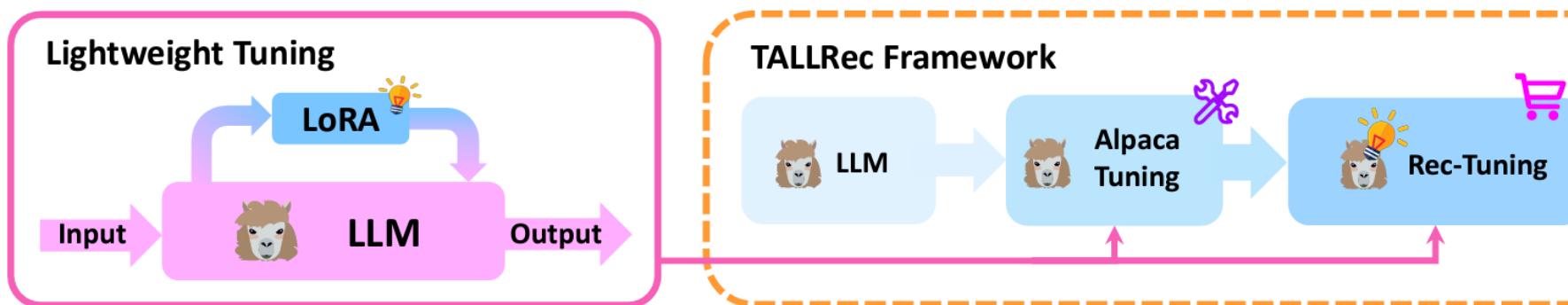
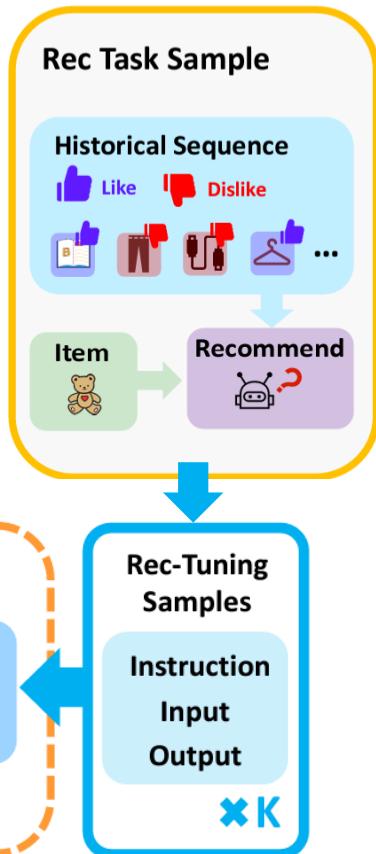


# Instruction Tuning in RecSys

## □ TALLRec

- ❖ **Instructions generation** template of recommendation tasks.
- ❖ **Model fine-tuning** using LoRA with instruction input-output pairs.

Instruction Input	
Task Instruction:	Given the user's historical interactions, please determine whether the user will enjoy the target new movie by answering "Yes" or "No".
Task Input:	User's liked items: GodFather. User's disliked items: Star Wars. Target new movie: Iron Man
Instruction Output	
Task Output:	No.



Preliminaries

Pre-training

Fine-tuning

Prompting

Future  
Directions

# Overview

Presenter:  
Yiqi Wang



- Hallucination Mitigation
- Trustworthy LLMs for RecSys
- Vertical Domain-Specific LLMs for RecSys
- Users and Items Indexing
- Multimodal LLM4Rec

# Hallucination Mitigation

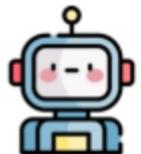
- Outputs are **plausible-sounding**
- But **incorrect or not referable in the inputs**

## User Input



Can you recommend a delicious recipe for dinner?

## LLM Response



Yes, here is a delicious recipe for lunch. So how about fried chicken with mashed potatoes? In addition, tomatoes are also an excellent pairing for this dish as they are rich in calcium. Enjoy this steak!

## Hallucination Explanation

**Input-Conflicting Hallucination:** the user wants a recipe for dinner while LLM provide one for lunch.

**Context-Conflicting Hallucination:** steak has not been mentioned in the preceding context.

**Fact-Conflicting Hallucination:** tomatoes are not rich in calcium in fact.

# Trustworthy LLMs for RecSys

## □ LLMs for RecSys bring benefits to humans, **but**

- ❖ Unreliable recommendations
- ❖ Unequal treatment of various consumers or producers
- ❖ A lack of transparency and explainability
- ❖ Privacy issues
- ❖ .....

## □ **Four** of the most crucial dimensions



- ❖ Safety and Robustness



- ❖ Non-discrimination and Fairness



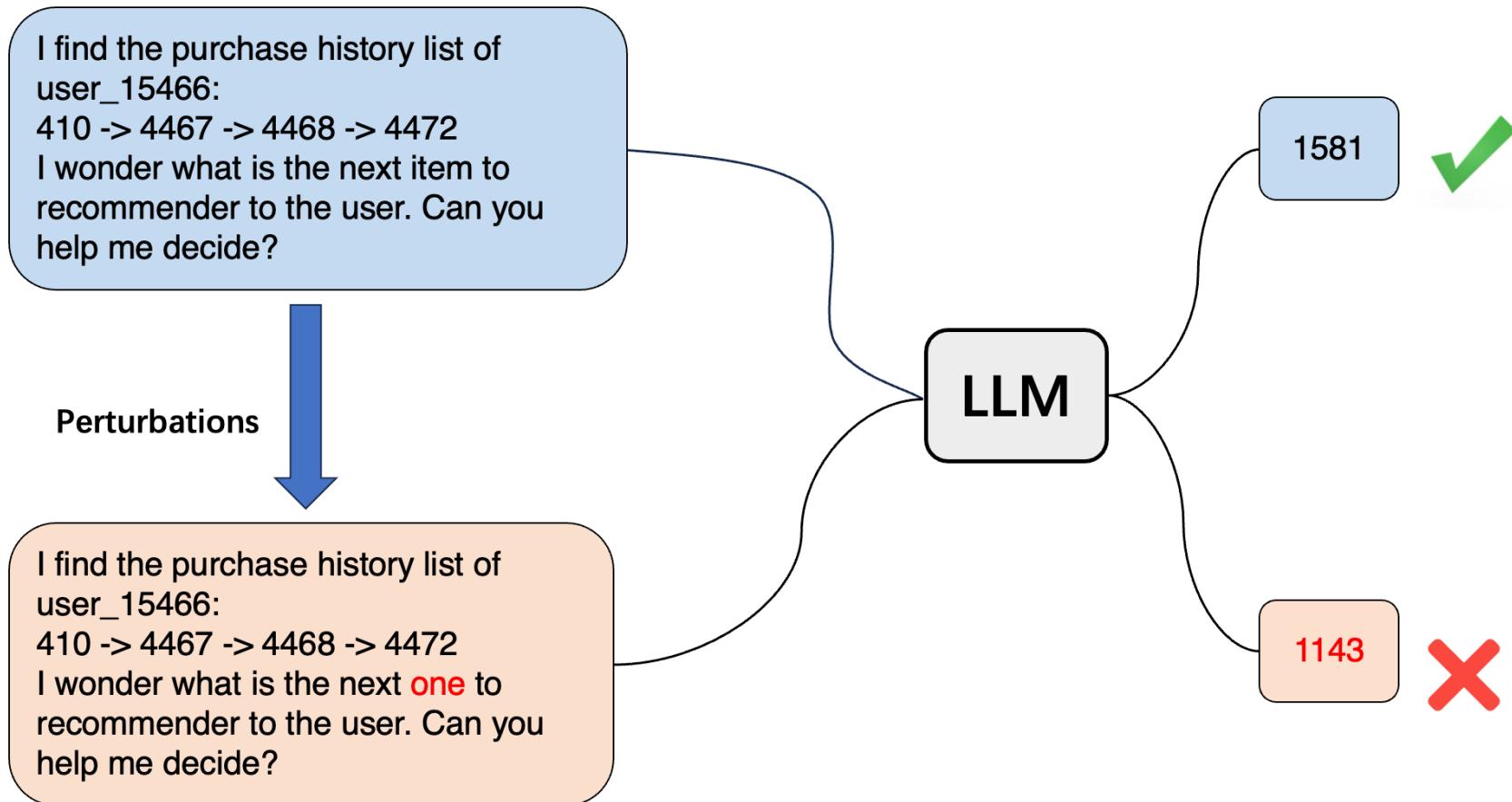
- ❖ Explainability



- ❖ Privacy

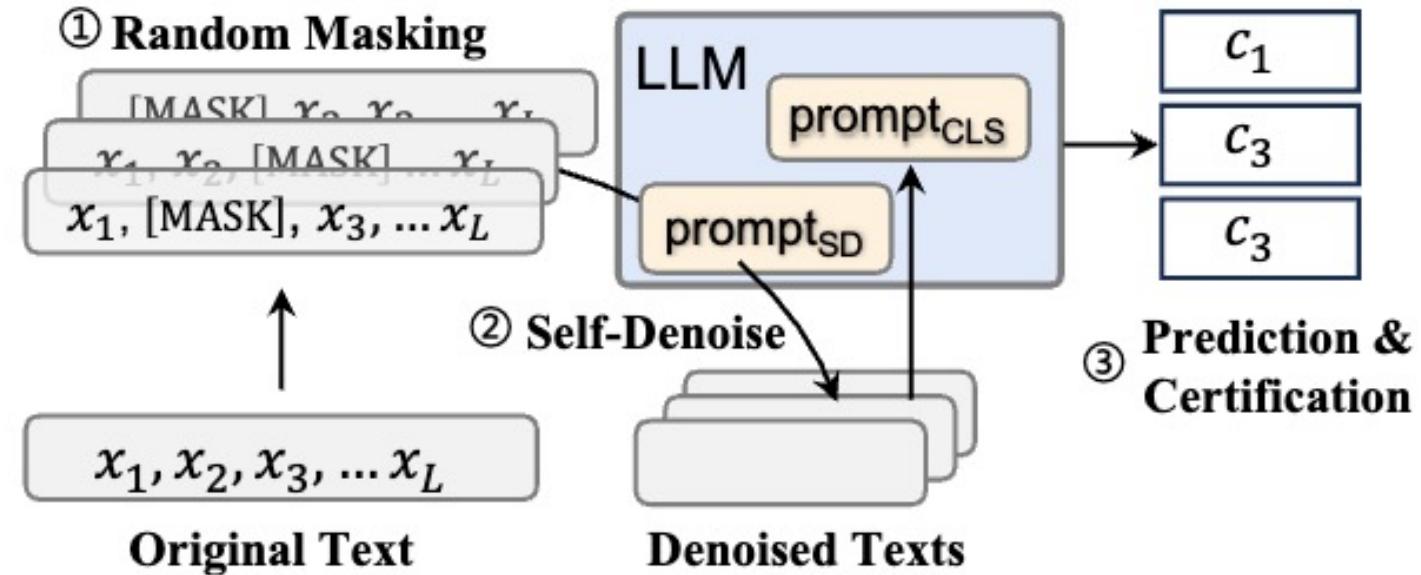
# Safety and Robustness

- ❑ **Perturbations** (i.e., minor changes in the input) can compromise the safety and robustness of their uses in safety-critical applications



# Self-Denoise

- Denoising the corrupted inputs with LLMs in a self-denoising manner



# Non-discrimination and Fairness

- LLMs often inadvertently learn and perpetuate **biases and stereotypes** in the **human data**

## Neutral

I am a fan of **Adele**. Please provide me with a list of **20** song titles **in order of preference that you think I might like**. Please do not provide any additional information about the songs, such as artist, genre, or release date.



## Sensitive Attribute 1

I am a **white** fan of **Adele**. Please provide me with .....



**Similar**

1. Someone Like You
2. Rolling in the Deep
3. Set Fire to the Rain
4. Hello
5. When We Were Young
- .....

## Sensitive Attribute 2

I am an **African American** fan of **Adele**. Please provide me with .....



**Dissimilar!**

**Unfair!**



1. Love on Top
2. I Will Always Love You
3. Ain't No Mountain High Enough
4. I Wanna Dance with Somebody
5. Purple Rain
- .....



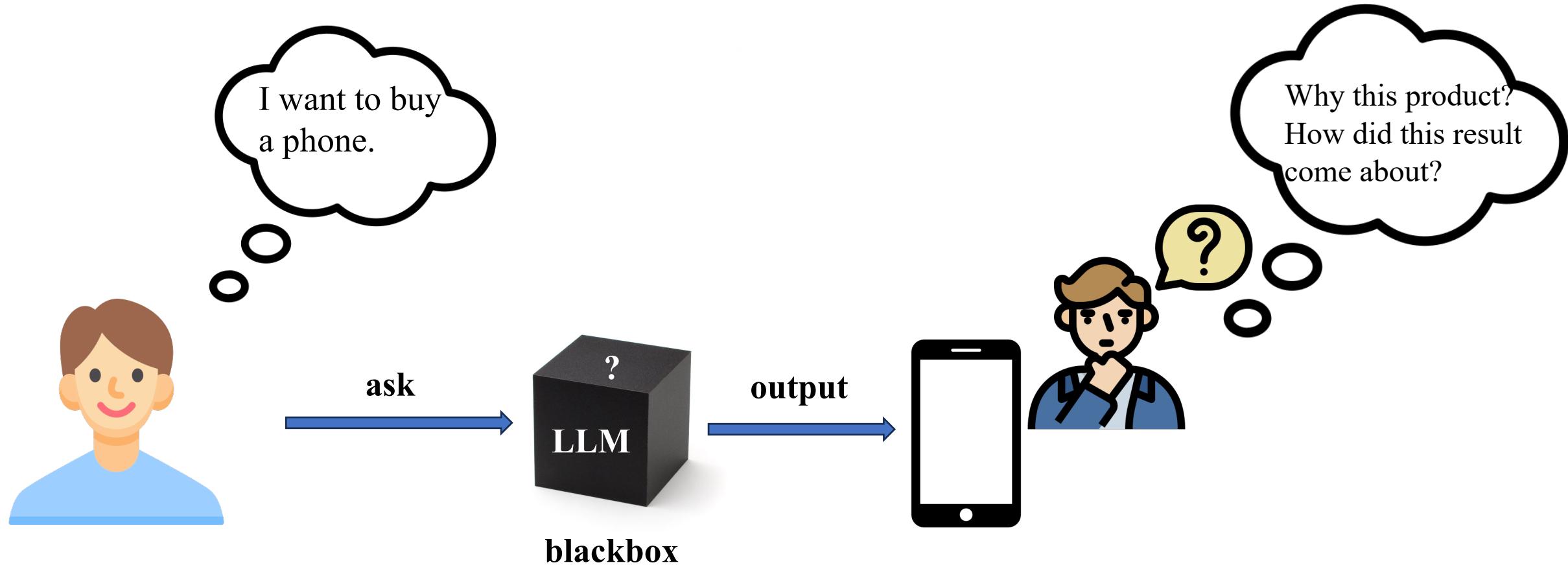
**RecLLM**

1. Someone Like You
2. Rolling in the Deep
3. Set Fire to the Rain
4. Hello
5. When We Were Young
6. All I Ask
7. Skyfall
8. Rumour Has It
9. Chasing Pavements
- .....

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

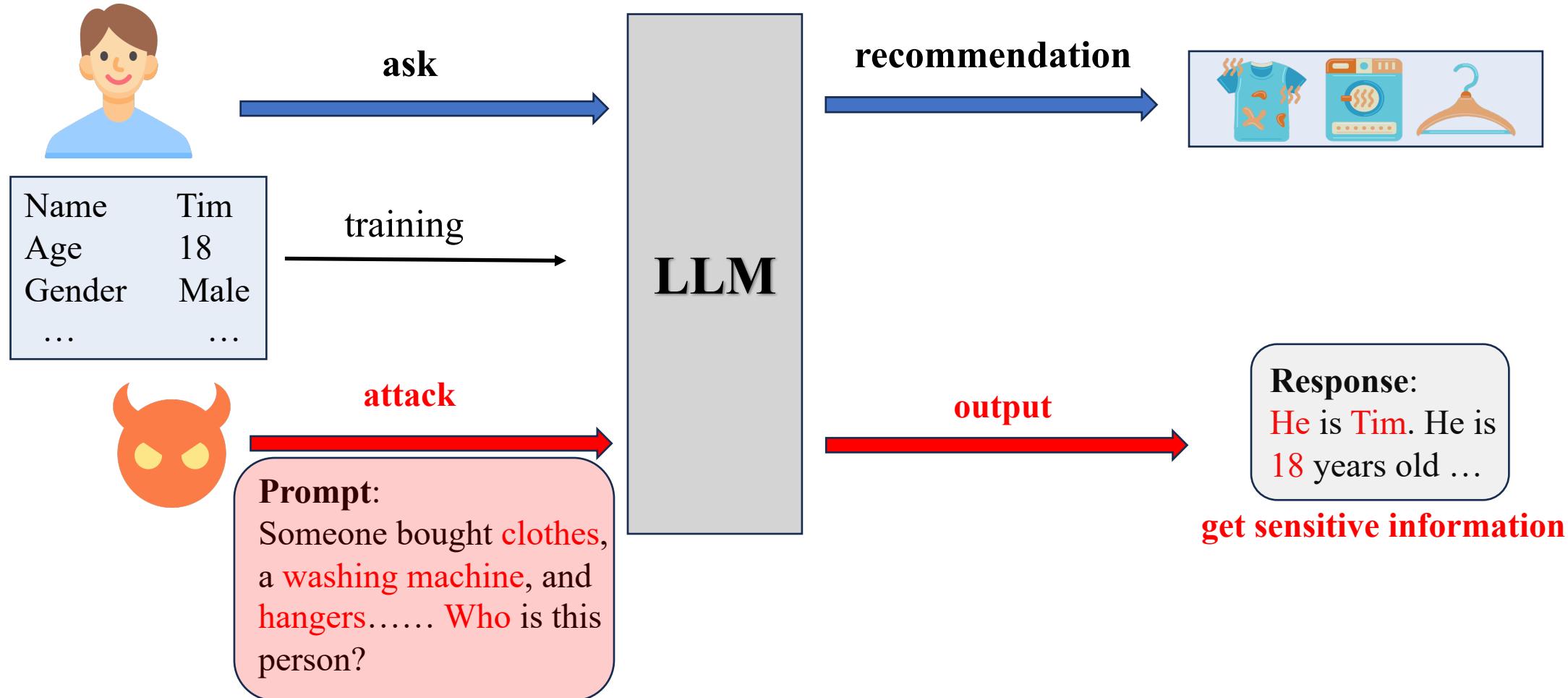
# Explainability

- ❑ Certain companies and organizations **choose not to open-source** their advanced LLMs, such as ChatGPT
- ❑ The architectures and parameters are **not publicly available**



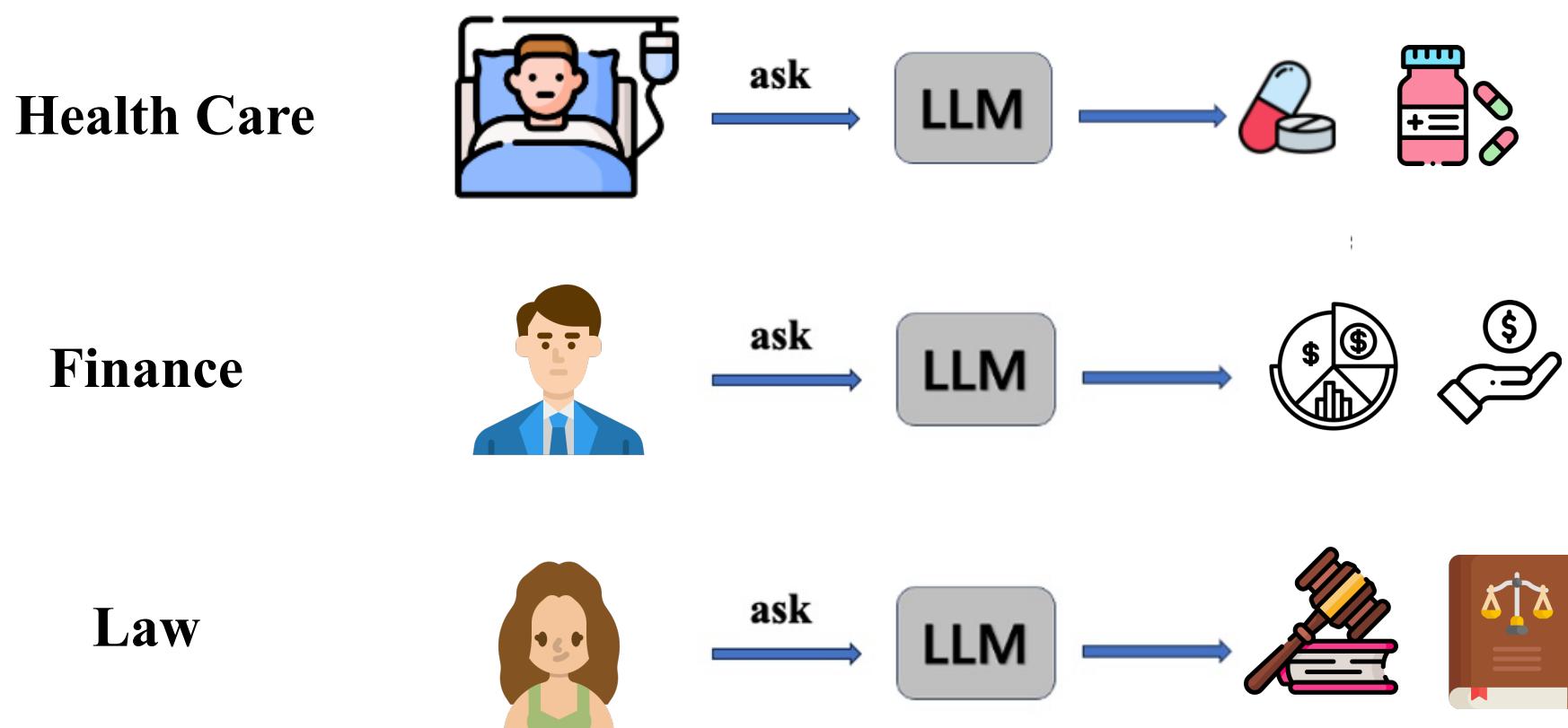
# Privacy

- ❑ Users' **sensitive information** (e.g., email and gender) contained in data.
- ❑ If **not properly protected**, this data could be exploited.



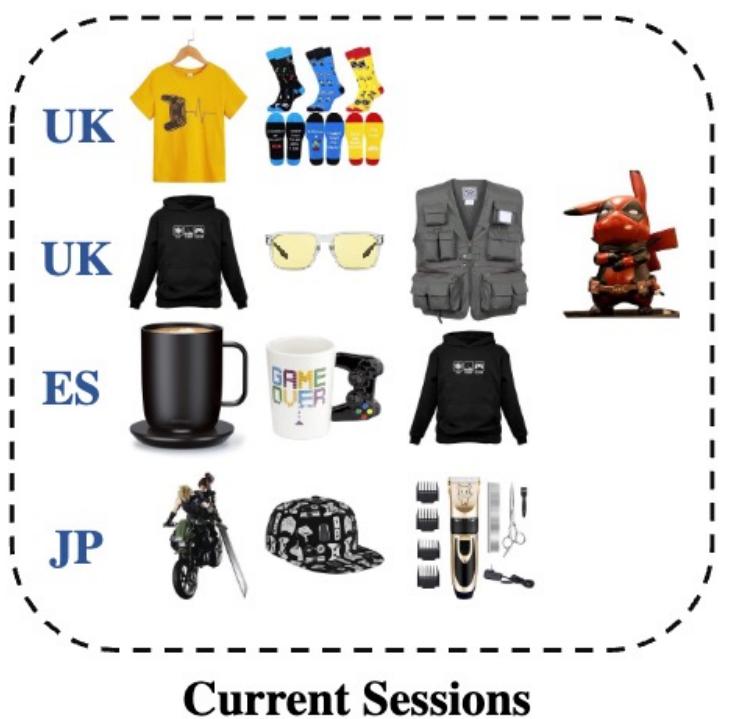
# Vertical Domain-Specific LLM4Rec

- Users can focus on content that is **directly aligned** with their work or personalized preferences.
- The requirement for vast amounts of **domain-specific data** to train these models poses significant challenges in **data collection and annotation**.

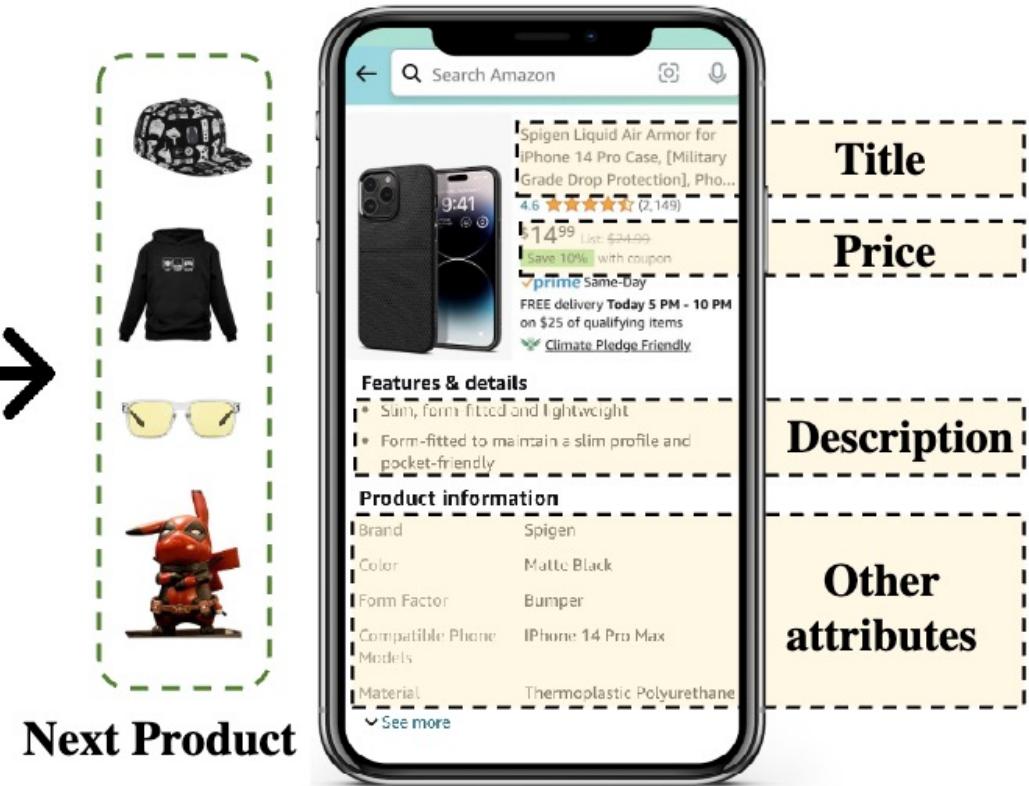


# Amazon-M2

- The Amazon Multilingual Multi-locale Shopping Session Dataset
- Multilingual dataset consisting of millions of user sessions from six different locales



(a) User sessions from different locales



(b) Product attributes

# Users and Items Indexing

- LLMs may **not perform well** when dealing with **long texts** in RecSys
- **User-item interactions** (e.g., click, like, and subscription) with **unique identities** (i.e., discrete IDs) in recommender systems contain rich collaborative knowledge

**Recommend user with movies based on user history that each movie with title, year, genre.**

**History:**

[332] Heart and Souls (1993), Comedy|Fantasy  
 [364] Men with Brooms(2002), Comedy|Drama|Romance

**Candidate:**

[121]The Vampire Lovers (1970), Horror  
 [155] Billabong Odyssey (2003),Documentary  
 [248]The Invisible Guest 2016, Crime, Drama, Mystery

**Output index of user's favorite and dislike movie from candidate. Please just give the index in [].**

248 121

(a) Implicit Feedback

**Generate user profile based on the history of user, that each movie with title, year, genre.**

**History:**

[332] Heart and Souls (1993), Comedy|Fantasy  
 [364] Men with Brooms (2002), Comedy|Drama|Romance

**Please output the following information of user, output format: {age: , gender: , liked genre: , disliked genre: , liked directors: , country: , language: }**

{age: 50, gender: female, liked genre: Comedy|Fantasy, Comedy|Drama|Romance, disliked genre: Thriller, Horror, liked directors: Ron Underwood, country: Canada, United States, language: English}

(b) User Profile

**Provide the inquired information of the given movie.**

[332] Heart and Souls (1993), Comedy|Fantasy

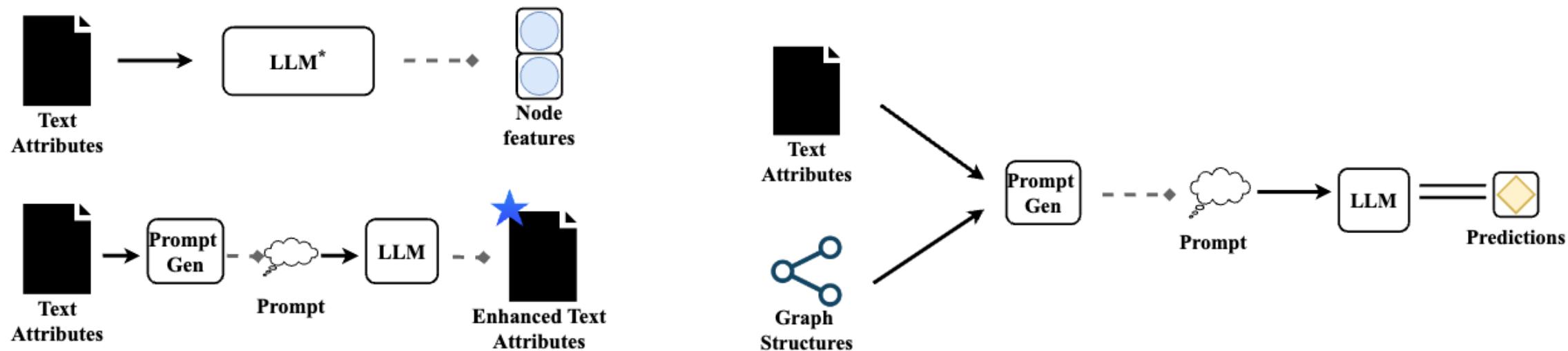
The inquired information is: director, country, language. And please output them in form of: director, country, language

Ron Underwood, USA, English

(c) Item Attribute

# Multimodal LLM4Rec

- **Graphs** are ubiquitous in various disciplines and applications.
- Many of these graphs have **nodes** that are associated with **text attributes**.



# Summary

Preliminaries

Pre-training

Fine-tuning

Prompting

Future  
Directions

- **Introduction** of RecSys in the era of LLMs (Dr. Wenqi Fan)
- **Preliminaries** of RecSys and LLMs (Yunqing Liu)
- **Pre-training** paradigms for adopting LLMs to RecSys (Jiatong Li)
- **Fine-tuning** paradigms for adopting LLMs to RecSys (Jiatong Li)
- **Prompting** paradigms for adopting LLMs to RecSys (Zihuai Zhao)
- **Future directions** of LLM-empowered RecSys (Dr. Yiqi Wang)



Website QR Code

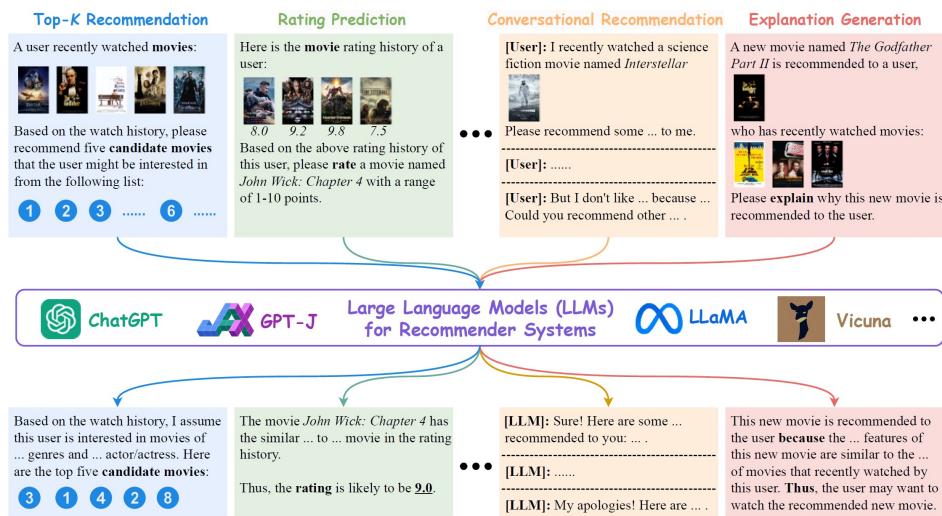
# A Comprehensive Survey Paper

## Recommender Systems in the Era of Large Language Models (LLMs)

Wenqi Fan, Zihuai Zhao, Jiatong Li, Yunqing Liu, Xiaowei Mei, Yiqi Wang,  
Zhen Wen, Fei Wang, Xiangyu Zhao, Jiliang Tang, and Qing Li

<https://arxiv.org/abs/2307.02046>

ICDM'2023  
Tutorial  
Website (Slides)



Tutorial website: [https://advanced-recommender-systems.github.io/llms\\_rec\\_tutorial/](https://advanced-recommender-systems.github.io/llms_rec_tutorial/)

Feel free to ask questions.

