Journal of Intelligent Information Systems
https://doi.org/10.1007/s10844-024-00915-3

RESEARCH

MLLM4Rec : multimodal information enhancing LLM for sequential recommendation

Yuxiang Wang[1] · **Xin Shi**[1] · **Xueqing Zhao**[1]

Received: 23 August 2024 / Revised: 12 December 2024 / Accepted: 17 December 2024

Abstract
In recent years, With the advent of large language models (LLMs) such as GPT-4, LLaMA, and ChatGLM, leveraging multimodal information (e.g., images and audio) to enhance recommendation systems has become possible. To further enhance the performance of recommendation systems based on large language models (LLMs), we propose MLLM4Rec, a sequence recommendation framework grounded in LLMs. Specifically, our approach integrates multimodal information, with a focus on image data, into LLMs to improve recommendation accuracy. By employing a hybrid prompt learning mechanism combined with role-playing for model fine-tuning, MLLM4Rec effectively bridges the gap between textual and visual representations, enabling text-based LLMs to "read" and interpret images. Moreover, the fine-tuned LLM is utilized to rank retrieval candidates, thereby maintaining its generative capabilities while optimizing item ranking according to user preferences. Extensive experiments were conducted on four publicly available benchmark datasets to evaluate the proposed method. The results demonstrate that MLLM4Rec outperforms partially LLM-based recommendation models, traditional sequential recommendation models, and sequential recommendation models pre-trained with multi-modal information in terms of NDCG, MRR, and Recall metrics.

Keywords Sequential recommendation · Large language model · Instruction tuning · Ranking

1 Introduction

In recent years, with the increasing prosperity of e-commerce and web applications, recommendation systems (RecSys) have become an indispensable and important part of our daily

Xin Shi and Xueqing Zhao contributed equally to this work.

✉ Yuxiang Wang
wangyuxiang123@outlook.com

Xin Shi
lrysx@163.com

Xueqing Zhao
zhaoxueqign@xpu.edu.cn

[1] Xi'an Polytechnic University, Lintong, Xi'an 710600, Shannxi, China

Published online: 30 December 2024

lives, providing users with personalized suggestions that match their preferences. Sequential recommendation (SR) is a highly regarded research topic in the field of recommendation systems, aiming to predict the next item a user might like based on their interaction history. Practical applications of sequential recommendation include predicting the next purchase, recommending the next song, or suggesting the next point of interest in a travel itinerary. Due to its high practical value, various algorithms have been proposed over the past few years, including methods that utilize additional item information (such as item categories). Early methods, such as using Markov chains (He & McAuley, 2016; Rendle et al., 2010), employed simple sequential models to capture sequence patterns in user behavior data. However, these methods faced limitations in capturing long-term dependencies and handling variable-length sequences. With the advent of deep learning technologies, more advanced methods have been proposed. Among them, GRU4Rec (Hidasi et al., 2015) employs the Gated Recurrent Unit (GRU) mechanism, SASRec (Kang & McAuley, 2018) was the first to use the Transformer (Vaswani et al., 2017) architecture in recommendation models, and BERT4Rec (Sun et al., 2019) adopts the BERT (Devlin et al., 2018) model.

With the rapid rise of large language models (LLMs) such as GPT-4 (Achiam et al., 2023), LLaMA (Touvron et al., 2023), and ChatGLM (Zeng et al., 2022; Du et al., 2021), aligning various modal features like images and audio with text enables LLMs to understand and reason about other modalities. The key to achieving this alignment lies in transforming the hidden representations from specific modality encoders (such as images encoded by ViT or Stable Diffusion, and audio encoded by HiFiGAN) into text token embeddings for LLMs (Yang et al., 2023). This allows LLMs to reason about the input modalities and generate corresponding text responses. By leveraging the generalization capabilities and commonsense reasoning of these large models (Shi et al., 2024), it is both rational and practical to enhance the performance of traditional sequential recommendation models (Hidasi et al., 2015; Kang & McAuley, 2018) and address challenging recommendation problems. However, LLM-based recommendation models still face two major challenges: (1) Due to the limitations of LLMs, many LLM-based recommendation models struggle to utilize additional multimodal information, resulting in the neglect of critical user-item interaction data. (2) When employing multimodal LLM-based recommendation models, the length of user-item interaction sequences often exceeds the processing capacity of multimodal LLMs, making it difficult to handle excessive multimodal information, such as images. Alternatively, processing this information can be too time-consuming, leading to inefficient inference.

In this work, we propose a framework to enhance the recommendation performance of large language models using multimodal information, which we call **M**ultimodal Information Enhancing **L**arge **L**anguage **M**odel **F**or Sequential **R**ecommendation (MLLM4Rec). Specifically, we first use text to describe each image, a process that helps bridge the gap between text and image modality representations. Subsequently, to enable the large language model to better learn user interaction features, we propose a hybrid prompt learning method and introduce role-playing to assist in fine-tuning the model. Finally, the fine-tuned large language model retains its generative capabilities while learning to rank items based on user preferences. Experiments on four public benchmark datasets demonstrate that our proposed MLLM4Rec framework outperforms partially LLM-based recommendation models, traditional sequential recommendation models, and sequential recommendation models pre-trained with multi-modal information. The main contributions of this paper are summarized below:

- We propose a framework to enhance the recommendation performance of large language models using multi-modal information. Unlike frameworks that use pure text information

for parameter-efficient fine-tuning, our proposed framework incorporates image information during fine-tuning, enabling pure text-based LLMs to acquire the ability to "read" images.

- We introduce a hybrid prompt learning mechanism. Specifically, we use text to describe each image and concatenate the title and text description of each candidate item. This process helps bridge the gap between text and image modality representations. Additionally, to mitigate the generative hallucination problem of LLMs, we further propose role-playing, which confines the understanding capabilities of LLMs within a specific scope.
- We conducted extensive experiments on four public benchmark datasets (ML-100K, Beauty, Video_Games, and Toys_Games), and the results on three evaluation metrics (NDCG, MRR, Recall) indicate that our proposed framework consistently outperforms traditional SR models and SR models that utilize multi-modal information.

In theory, the proposed model integrates multimodal information with large language models, leveraging hybrid prompt learning and role-playing methods to overcome the limitations of traditional recommendation systems in effectively handling multimodal data. In practice, by employing instruction-based fine-tuning, the LLM demonstrates strong generalization capabilities in zero-shot and few-shot scenarios, enabling it to adapt to unseen items in new domains.

The rest of this paper is organized as follows: the related work is given in Section 2; the proposed method is described in Section 3; Simulation experiments are presented in Section 4; and finally, the conclusions of this paper are given in Section 5.

2 Related work

2.1 Sequential recommendation

In the literature on recommendation systems, sequential recommendation is a highly regarded research topic, aiming to predict the next item a user might like based on their interaction history (Wang et al., 2022a, b; Xie et al., 2023). Various methods have been proposed to address the task of sequential recommendation. Early methods primarily used Markov chains (He & McAuley, 2016; Rendle et al., 2010) to compute the item-item transition probability matrix and predict the next item. However, these methods are not well-suited for handling complex sequential patterns. Additionally, with the development of deep learning methods, various deep neural networks have been proposed to capture user preferences in historical interaction sequences. These methods include recurrent neural networks (RNN) (Hidasi & Karatzoglou, 2018; Hidasi et al., 2015; Quadrana et al., 2017), convolutional neural networks (CNN) (Tang & Wang, 2018; Yuan et al., 2019). However, these solutions often fail to capture long-term dependencies between arbitrary items in the sequence. In light of this, attention-based networks (Kang & McAuley, 2018; Li et al., 2017; Sun et al., 2019; Lei et al., 2022; Gao et al., 2024) have been widely adopted and have achieved promising results in sequential recommendation.

To further enhance the performance of sequential recommendation, recent studies have integrated auxiliary information about items, such as image data. IDA-SR (Mu et al., 2022), UniSRec (Hou et al., 2022), and VQ-Rec (Hou et al., 2023) have achieved multimodal recommendation by learning item representations from foundational models in the NLP field. In terms of image representation in recommendation systems, (Wei et al., 2019) and

Meng et al. (2020) input image features extracted from ResNet-50 (He et al., 2016) into the recommendation model. Although the aforementioned studies have made significant progress in their respective areas, they have not extensively explored the application of LLMs in sequential recommendation.

2.2 LLM-based recommendation

In recent years, the emergence of Large Language Models (LLMs) has opened a productive research direction for sequential recommendation. These models repeatedly complete a sequence in an auto-regressive mode until they fulfill the prompt task (Radford & Narasimhan, 2018). Researchers have primarily focused on how to leverage these models in sequential recommendation tasks. Existing studies on LLMs in recommendation systems can be divided into two categories: Discriminative LLMs for Recommendation (DLLM4Rec) and Generative LLMs for Recommendation (GLLM4Rec) (Wu et al., 2023). The former is primarily based on BERT models (Devlin et al., 2018), aligning the representations learned by LLMs with the recommendation domain, while the latter mostly employs models like ChatGPT and ChatGLM to transform recommendation tasks into natural language tasks, such as by providing item embeddings (Hou et al., 2023) or responding to recommendation tasks within prompts, which may include zero-shot (Ding et al., 2021) or few-shot examples (Geng et al., 2022). MINT (Mysore et al., 2023) used InstructGPT (a 175B parameter LLM) to generate a synthetic narrative query. This query was then filtered using a smaller language model, and retrieval models were trained using the synthetic queries and user items. However, this approach did not consider decomposing the topics within textual descriptions, which could lead to noisy prompts and unclear targets. KAR (Xi et al., 2024) addressed this issue by introducing factorization prompts, enabling accurate reasoning about user preferences and factual knowledge. In addition to using LLMs as recommendation systems, some studies have also utilized LLMs to construct model features. GENRE (Liu et al., 2023) introduced three prompts to conduct three feature enhancement sub-tasks for news recommendation using LLMs. Hou et al. (2024) introduced demonstration examples by augmenting the input interaction sequence itself, demonstrating that in-context learning can enhance the recommendation abilities of LLMs in most tasks. Some studies also fine-tune LLMs to learn how to better perform specific recommendation tasks (Hou et al., 2022). GPTRec (Petrov & Macdonald, 2023) is a generative sequential recommendation model based on GPT-2. TALLRec (Bao et al., 2023) adopts the Parameter Efficient Fine-Tuning (PEFT) method, also known as LoRA. This approach surpasses traditional recommendation models in mitigating cold start issues and handling the complexities of cross-domain recommendation scenarios. GenRec (Ji et al., 2024) leverages the generative capabilities of LLMs to directly generate target recommendation items. In addition to directly fine-tuning LLMs, some studies have proposed using prompt learning to achieve better performance. For instance, UniCRS (Wang et al., 2022) freezes the parameters of LLMs and trains soft prompts for response generation and item recommendation through prompt learning.

In fact, generative LLMs are just the latest in a wave of innovations in the field of Natural Language Processing (NLP), which has led to the creation of new sequential recommendation models. Previous work in the LLM-based field has achieved remarkable success, but we observe that existing approaches tend to input single-modal information into the models, without considering the use of auxiliary information such as images. Moreover, current large language models still face efficiency or performance limitations when processing large amounts of multimodal information.

3 Methodology

3.1 Problem description

We consider a recommendation system containing a set of users U and a set of items I, where the user set is represented as $U = \{u_1, u_2, \ldots, u_n\}$ and the item set is represented as $I = \{i_1, i_2, \ldots, i_m\}$. Our recommendation framework takes the user interaction history $\mathbf{x}$ from the dataset X as input. Specifically, $\mathbf{x}$ is a sequence of interaction items arranged in chronological order $[x_1, x_2, \ldots, x_T]$, where each item is defined in the item space I ($x_i \in I$, $i = 1, 2, \ldots, T$), and these items are represented by unique IDs. During the retrieval phase, we take X as input and finally output K candidate items (where $K = 20$) for each X. In the ranking phase, we use item titles to understand user behavior, and our framework outputs a ranking score $\hat{y} \in \mathbb{R}^{|I|}$, with the ground truth denoted by $y \in I$.

To optimize, we denote the model as $\mathbf{f}$ with parameters θ (i.e., $\hat{y} = \mathbf{f}(\theta; \mathbf{x})$), which includes an efficient retrieval model $\mathbf{f}_{\text{retriever}}$ and an LLM-based ranking model $\mathbf{f}_{\text{ranker}}$ ($\mathbf{f} = \mathbf{f}_{\text{ranker}} \circ \mathbf{f}_{\text{retriever}}$). Ideally, the item with the highest score in $\hat{y}$ should be the ground truth item y (i.e., $y = \arg\max \hat{y}$).

Therefore, the goal of our framework is to maximize the score of the true item, which is equivalent to minimizing the expected negative log-likelihood loss L of the parameters θ over the dataset X:

$$\min_{\theta} \mathbb{E}_{(x,y)\sim X} \left[\mathcal{L}(f(\theta; x), y)\right] \tag{1}$$

where $\boldsymbol{f}$ represents an efficient retrieval model $\boldsymbol{f}_{\text{retriever}}$ and a ranking model based on an LLM $\boldsymbol{f}_{\text{ranker}}$ ($\boldsymbol{f} = \boldsymbol{f}_{\text{ranker}} \circ \boldsymbol{f}_{\text{retriever}}$); $\boldsymbol{\theta}$ denotes the parameters; y indicates that the highest scoring item in $\hat{y}$ should be the ground truth item; $\hat{y}$ represents $\hat{y} = \boldsymbol{f}(\boldsymbol{\theta}; \boldsymbol{x})$; $\mathcal{L}$ denotes negative log likelihood loss; X denotes dataset; E for Expectations.

Our framework first filters out potentially interesting items using an efficient retrieval model, and then precisely ranks them with an LLM-based ranking model to maximize the score of the true items, ultimately optimizing the negative log-likelihood loss to enhance recommendation accuracy.

3.2 Overview of MLLM4Rec

Our proposed sequential recommendation framework based on LLMs and multi-modal information learning is illustrated in Fig.1. The framework is divided into three main parts: data processing, prompt construction, instruction tuning, and candidate ranking. Our framework is inspired by Yue et al. (2023) and first employs a lightweight sequential recommendation model (LRURec (Yue et al., 2024)) to retrieve candidate items based on the user's interaction history. Additionally, we utilize a multi-modal pre-trained language model, Blip2, to convert images of items interacted with by users into textual descriptions in a zero-shot manner. The historicords, retrieved items, and these textual descriptions are then fed into the LLM in text form through a designed prompt template. To reduce the hallucination problem of LLMs, we add role components to the prompt template, selecting different roles based on different datasets. Finally, instead of generating the next item as traditional recommendation models do, we adopt a word conversion method that transforms the output logits into a probability distribution over the candidate items. This approach aims to efficiently rank the items.

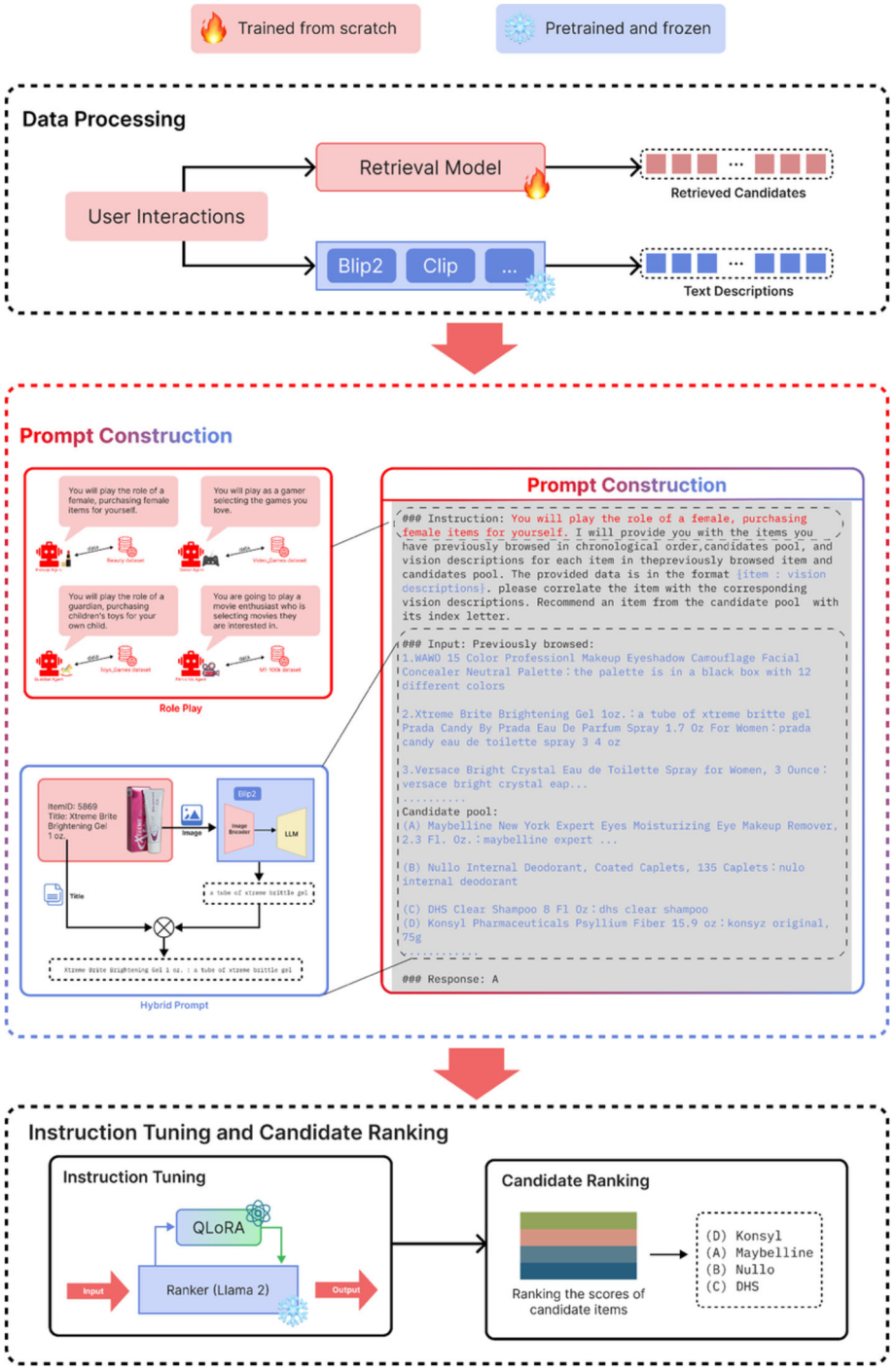


Fig. 1 Overall framework of MLLM4Rec. Fire represents trainable parameters, and ice represents frozen parameters. The framework is divided into three main parts: data processing, prompt construction, instruction tuning, and candidate ranking

3.3 Retrieval model

To compare with the original framework (Yue et al., 2023) and demonstrate the effectiveness of our approach, we adopted the same implementation strategy. Theoretically, the retrieval model and ranking model are decoupled, allowing any model to be substituted for these two components.

We employed LRURec as the retrieval model, which consists of the following parts: (1) an embedding module; (2) LRU blocks with Position Feed-Forward Networks (PFFN); (3) a prediction layer. During training, LRURec processes the input sequence using LRU blocks with PFFNs and optimizes through autoregressive training to capture user interests. During inference, LRURec generates item scores by computing the dot product between item embeddings and prediction features. Finally, we collect the top k (in our experiments $k = 20$) recommended candidate items from LRURec for each input sequence to be used in the subsequent ranking stage.

3.4 Prompt construction

In order to reduce the generative illusions of the model (Boz et al., 2024; Lin et al., 2023), we propose a role-playing learning method, with different openings for different datasets. We assemble role-plays and image-texts as our text input into the large language model, and the following is our template on role-plays in Fig. 2:

To address the issue where existing LLMs are unable to handle massive multimodal information simultaneously, either due to technical limitations or efficiency problems, we pioneeringly used Blip2 to process image information. Specifically, we used images of each item in the dataset. We connected each item's title with the converted text using a ":", and used this method to input data into the large language model for learning, which we refer to as hybrid prompt learning.

Beauty dataset

You will play the role of a female, purchasing female items for yourself.

Video_Games dataset

You will play as a gamer selecting the games you love.

Toys_Games dataset

You will play the role of a guardian, purchasing children's toys for your own child.

ML-100K dataset

You are going to play a movie enthusiast who is selecting movies they are interested in.

Fig. 2 Template on role-plays

Thus, the "materials" needed to construct the final template have been collected, forming our final template Fig. 3:

where the history, candidate items and labels are replaced by a history item title, a candidate item title and a label item for each data example, respectively.

Taking the Beauty dataset as an example, the red and blue striped area in Fig. 1 represents our generated template. First, we specify the role the model should play at the beginning, then inform the model of the input format. Next, we concatenate the titles of the user's historical records with the image text using a colon ":", sorted in chronological order. The order of the candidate items is determined by the scores generated by the lightweight sequential recommendation model, with letters prefixed to them.

3.5 Ranker

We chose Llama 2 as the base model for ranking (Touvron et al., 2023). Following the method described in Yue et al. (2023), we used a simple vocabulary converter to transform the output of the LLM (i.e., the scores of all tokens) into ranking scores for the candidate items. Specifically, index letters were used to identify the candidate items, and the actual items were mapped to the corresponding index letters. Then, the candidate item scores could be calculated by retrieving the logits of the index letters from the head of the LLM. Therefore, ranking inference requires only a single forward pass to obtain scores for all candidate items, significantly improving ranking efficiency.

We adopted instruction-based fine-tuning, a technique proven effective in recent LLM developments (Taori et al., 2023; Wang et al., 2022; Wei et al., 2021). Additionally, we used QLoRA for quantization, which significantly reduces the GPU memory resources needed for training large models. During training, we used the index letters of the actual items as labels to learn to maximize the scores of the actual items. During inference, MLLM4Rec retrieves logits via the vocabulary converter and ranks the relevant items. By using index letters and the vocabulary converter, the LLM model retains its generative capabilities while learning to rank items based on user preferences. To reduce the input length for the LLM, we set the maximum number of user historical items to 20 and rank the top 20 candidate items from the retrieval model.

Instruction:
You will play the role of a female, purchasing female items for yourself. I will provide you with the items you have previously browsed in chronological order, candidates pool, and vision descriptions for each item in the previously browsed item and candidates pool. The provided data is in the format {item : vision descriptions}. Please correlate the item with the corresponding vision descriptions. Recommend an item from the candidate pool with its index letter.
Input:
Previously browsed: { history : vision descriptions }; Candidate pool: { history : vision descriptions }
Response: { label }

Fig. 3 Prompt template

4 Simulation experiments

4.1 Dataset description

In this paper, we selected three sub-category datasets from the Amazon review dataset[1], namely Beauty, Video_Games, and Toys_Games as well as MovieLens-100K (ML-100K)[2], which is used in movie recommender systems. We followed the methodology used in a previous study (Yue et al., 2023) to process these datasets, retaining the five-core datasets and filtering out users and items with fewer than five interactions. Subsequently, we grouped interactions by users and sorted them in ascending order according to timestamps. For the three Amazon datasets and ML-100K datasets, images related to the items in the Amazon datasets were crawled as visual modality information. Additionally, we used BLIP2[3] to convert the obtained images into textual descriptions. The statistics of the preprocessed datasets are shown in Table 1.

4.2 Baseline Methods

We compare the proposed MLLM4Rec with the following baseline methods:

- **LlamaRec** (Yue et al., 2023): LlamaRec is a two-stage framework that uses a large language model for ranking-based recommendations. In the first stage, it uses a small-scale sequential recommender to retrieve candidates based on the user's interaction history. In the second stage, it uses an LLM for ranking, adopting a verbalizer-based approach to convert the output into a probability distribution over the candidates.
- **P5** (Geng et al., 2022; Xu et al., 2023): P5 is an llm-based recommendation model that proposes a flexible and unified text-to-text paradigm called "Pretrain, Personalized Prompt, and Predict Paradigm".
- **RecFormer** (Li et al., 2023): RecFormer models user preferences and item characteristics as language representations that can generalize to new items and datasets.
- **ECL-SR** (Zhou et al., 2023): This method proposes an equivariant contrastive learning framework for sequential recommendation. ECL-SR can build robust contrastive learning based on mild feature-level augmentation and intrusive sequence-level augmentation, thereby learning more informative representations for sequential recommendation.
- **STOSA** (Fan et al., 2022): STOSA proposes a self-attention-based sequential recommendation model, consisting of three modules: random embeddings, Wasserstein self-attention, and a regularization term in the BPR loss.
- **MMSRec** (Song et al., 2023): MMSRec is a multi-modal sequential recommendation model based on Transformer.

4.3 Evaluation metrics

In our evaluation, we adopted the leave-one-out strategy, where the last item in each data example is used for testing, the second-to-last item is used for validation, and the remaining items are used for training. The evaluation metrics include Mean Reciprocal Rank (MRR@k),

[1] https://cseweb.ucsd.edu/~jmcauley/datasets/amazon/links.html

[2] https://grouplens.org/datasets/movielens/100k/

[3] https://huggingface.co/Salesforce/blip2-opt-2.7b

Table 1 Dataset statistics after preprocessing

Datasets	#Users	#Items	#Interact	#Density
Beauty	22332	12086	198215	7e-4
Video_Games	12601	6258	120622	1.5e-3
Toys_Games	19124	11758	165247	7e-4
ML-100K	610	3650	100837	4e-2

Normalized Discounted Cumulative Gain (NDCG@k), and Recall (Recall@k), where $k \in [5, 10]$. The detailed formulae for the three evaluation indicators are as follows:

1. Mean Reciprocal Rank (MRR@k) denotes the reverse order of the average result, indicating whether the item to be recommended is more visible to the user, emphasizing "order". The formula is given by:

$$\mathrm{MRR@}k = \frac{1}{|Q|}\sum_{i=1}^{|Q|}\frac{1}{\mathrm{p}_i} \tag{2}$$

where $|Q|$ denotes the total number of users and p_i denotes the position of the real access value of the i-th user in the recommendation list. @k means that only the first k items are counted to see if a correct response exists, and if it doesn't, it is invalidated.

2. Recall@k represents the proportion of correctly predicted positive samples among all positive samples, indicating the proportion of user-item interaction records that are included in the final prediction list. The formula is:

$$\mathrm{Recall@}k(u) = \frac{|R_k(u) \cap T(u)|}{|T(u)|} \tag{3}$$

where $R_k(u)$ is the set of top k recommended items for user u, $T(u)$ is the set of relevant items for user u, $|R_k(u) \cap T(u)|$ is the number of relevant items that appear in the top k recommendations, $|T(u)|$ is the total number of relevant items for user u.

3. Normalized Discounted Cumulative Gain (NDCG@k) is a popular metric for evaluating ranking quality. A recommender system usually returns a list of items for a user, and assuming that the list is of length K, the difference between the sorted list and the real list of user interactions can be evaluated using NDCG@K.To calculate NDCG@k, follow these steps:

(1) Cumulative Gain (CG) is calculated for the first k items of the list as follows:

$$\mathrm{CG}_k = \sum_{i=1}^{k}\mathrm{rel}_i \tag{4}$$

where rel_i denotes the relevance (score) of the recommendation result at position i,k denotes the size of the recommendation list to be examined.

(2) Discounted Cumulative Gain (DCG) DCG for the top k recommended items is given by:

$$\mathrm{DCG}_k = \sum_{i=1}^{k}\frac{\mathrm{rel}_i}{\log_2(i+1)} \tag{5}$$

where rel_i denotes the relevance score of the i-th item in the list.

(3) Ideal Discounted Cumulative Gain (IDCG) is the DCG of the ideal ranking:

$$\mathrm{IDCG}_k = \sum_{i=1}^{k} \frac{\mathrm{rel}_{\mathrm{ideal}_i}}{\log_2(i+1)} \tag{6}$$

where $\mathrm{rel}_{\mathrm{ideal}_i}$ represents the relevance score of the i-th item in the ideal ranking.

(4) Normalized Discounted Cumulative Gain (NDCG) is obtained by normalizing DCG with IDCG:

$$\mathrm{NDCG}_k = \frac{\mathrm{DCG}_k}{\mathrm{IDCG}_k} \tag{7}$$

We saved the model with the highest validation score for evaluation (Recall@10 for retrieval tasks and NDCG@10 for ranking tasks). The prediction results are ranked against all items in the dataset.

4.4 Implement details

For the benchmark models mentioned, we utilized the source codes provided by the authors (Yue et al., 2023; Li et al., 2023; Zhou et al., 2023; Fan et al., 2022; Song et al., 2023), including the preprocessing of the datasets, and strictly followed the steps described in the README.md. Our framework inherits part of the settings from Yue et al. (2023), including training with the AdamW optimizer, a learning rate of 0.001, and a maximum of 500 training epochs. Validation is conducted every 500 iterations, and early stopping is triggered if the validation performance does not improve for 20 consecutive epochs. To determine the hyperparameters, we performed a grid search over weight decay [0, 1e-2] and dropout rates [0.1, 0.2, 0.3, 0.4, 0.5]. For the datasets, we set the maximum sequence length to 50. For our ranker, we used up to 20 historical items and ranked the top 20 candidate items returned by the retrieval model. Titles were truncated if they exceeded 32 tokens. We quantized the Llama 2-based ranker using QLoRA, with 8 as the LoRA dimension, 32 as α, and a dropout rate of 0.05. The learning rate for LoRA was set to 1e-4, targeting the Q and V projection matrices. The model was fine-tuned for one epoch, with validation conducted every 100 iterations. Similarly, the model with the best validation performance was saved for evaluation on the test set. Finally, we implemented our method on Pytorch 2.0.0 + 32G DDR4 RAM + RTX 4090 24G.

4.5 Recommended performance

Discussion on main results We conducted a comparative analysis of the proposed MLLM4Rec with several baseline methods on four public datasets, as shown in Table 2. In the table, each major row represents an evaluation metric, and each column represents a recommendation method. For clarity, the best results are highlighted in bold, and the second-best results are underlined.

According to the experimental results, we observe that MLLM4Rec outperforms the LLM-based model (P5) and traditional pre-trained models using multimodal data (MMSRec) and conventional sequential recommendation models (RecFormer, ECL-SR, and STOSA) in most cases. Specifically, MLLM4Rec shows an average improvement of approximately 17.47% in NDCG@10, 1.41% in Recall@10, and 33.97% in MRR@10 across the four datasets

Table 2 Recommended results of our proposed MLLM4Rec with several other different methods under NDCG, MRR and Recall

Dataset	Metric	MLLM4Rec	P5	MMSRec	RecFormer	ECL-SR	STOSA	Improv.
Beauty	Recall@5	**0.0633**	0.0446	0.0551	0.0358	0.0488	0.0592	6.93%
	Recall@10	0.0870	0.0613	**0.0934**	0.0597	0.0813	0.0840	-6.85%
	NDCG@5	**0.0454**	0.0333	0.0327	0.0208	0.0265	0.0413	9.93%
	NDCG@10	**0.0531**	0.0387	0.045	0.0284	0.0369	0.0492	7.93%
	MRR@5	**0.0396**	0.0213	0.0253	0.0158	0.0191	0.0267	48.31%
	MRR@10	**0.0427**	0.0301	0.0304	0.019	0.0234	0.0245	40.46%
Video_Games	Recall@5	**0.110**	0.0612	0.0805	0.0641	0.0697	0.0697	36.65%
	Recall@10	**0.1593**	0.0845	0.1371	0.0854	0.1181	0.1091	16.19%
	NDCG@5	**0.0733**	0.0403	0.0479	0.0510	0.0385	0.0456	43.73%
	NDCG@10	**0.0893**	0.0501	0.0660	0.0579	0.054	0.0582	35.30%
	MRR@5	**0.062**	0.0355	0.0372	0.0468	0.0283	0.0324	32.48%
	MRR@10	**0.0679**	0.0432	0.0446	0.0496	0.0347	0.0329	36.90%
Toys_Games	Recall@5	**0.0636**	0.0508	0.0529	0.0442	0.0545	0.0537	16.70%
	Recall@10	0.0842	0.0712	**0.0891**	0.0819	0.0848	0.0750	-5.50%
	NDCG@5	**0.0470**	0.0345	0.0308	0.0236	0.0284	0.0380	23.68%
	NDCG@10	**0.0538**	0.0386	0.0424	0.0358	0.0381	0.0448	20.09%
	MRR@5	**0.0416**	0.0211	0.0236	0.0169	0.0199	0.0226	76.27%
	MRR@10	**0.0444**	0.0273	0.0284	0.0219	0.0238	0.0208	56.34%
ML-100K	Recall@5	**0.0541**	0.0532	0.0512	0.0499	0.0521	0.0441	1.69%
	Recall@10	**0.1016**	0.0975	0.0901	0.0910	0.0998	0.0909	1.80%
	NDCG@5	**0.0333**	0.0317	0.0323	0.0298	0.0321	0.0311	3.10%
	NDCG@10	**0.0485**	0.0435	0.0455	0.0412	0.0433	0.0430	6.59%
	MRR@5	**0.0264**	0.0211	0.0210	0.0233	0.0223	0.0221	13.30%
	MRR@10	**0.0326**	0.0319	0.0299	0.0311	0.0309	0.0315	2.19%

The best results are in bold and the second best results are underlined

compared to the second-best models. These results confirm that utilizing multimodal information for instruction fine-tuning LLMs can significantly enhance the ranking performance of LLMs, while the application of LLMs in the recommendation field has become an unstoppable historical trend.

Discussion on whether the model uses multimodal information Additionally, we compared the performance of using multimodal information versus not using it, using the same `lora_micro_batch_size` (with Toys_Games and Video_Games set to 2, Beauty set to 6 and ML-100k set to 3) and the same dataset, as shown in Table 3. We found that enhancing the large language model with multimodal information generally improved the model's performance compared to the non-enhanced version. However, for several evaluation metrics in the Beauty dataset, the performance was either equal or declined. This may be related to the content of the dataset. For example, in the Video_Games category, titles are often abstract, such as Dirt 3, Tomb Raider II, and Super Mario 64. Large language models typically do not understand the content, making it difficult to recommend related content accurately. When multimodal information is added, such as text descriptions of game posters, the large language model can grasp their meaning, which is also a crucial factor for consumers deciding whether to purchase. In contrast, the titles in the Beauty category already contain enough information, making the introduction of multimodal information less important.

Table 3 Performance comparison of MLLM4Rec(Using Multimodal Information) and LlamaRec(Not Using It) across different datasets

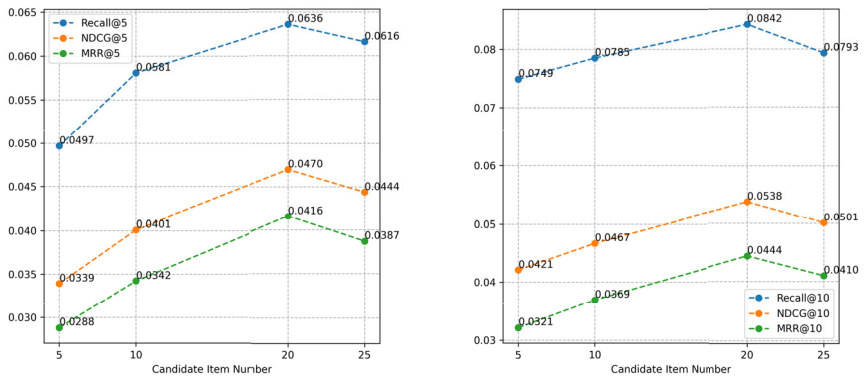
Dataset	Metric	LlamaRec	MLLM4Rec	Improv.
Beauty	Recall@5	**0.0643**	0.0633	–
	Recall@10	**0.0873**	0.0870	–
	NDCG@5	**0.0457**	0.0454	–
	NDCG@10	0.0531	**0.0531**	eq
	MRR@5	0.0396	0.0396	eq
	MRR@10	0.0426	**0.0427**	2.34%
Video_Games	Recall@5	0.101	**0.110**	8.91%
	Recall@10	0.149	**0.1593**	6.91%
	NDCG@5	0.0689	**0.0733**	6.39%
	NDCG@10	0.0844	**0.0893**	5.81%
	MRR@5	0.0584	**0.0620**	6.16%
	MRR@10	0.0648	**0.0679**	4.78%
ML-100K	Recall@5	0.0525	**0.0541**	3.05%
	Recall@10	0.0984	**0.1016**	3.25%
	NDCG@5	0.0309	**0.0333**	7.77%
	NDCG@10	0.0457	**0.0485**	6.13%
	MRR@5	0.0238	**0.0264**	10.92%
	MRR@10	0.0299	**0.0326**	9.03%

The best results are in bold

Discussion on historical item number In Section 3.2, we mentioned that the prompt text input to the LLM-based recommendation system includes information about items from the user's historical interactions and the candidate items. In Sections 3.3 and 3.5, we set the number of candidate items to 20. In this section, we explore the impact of different numbers of candidate items on the results, using the Toys_Games dataset as an example. The results are shown in Fig. 4. We found that as the number of candidate items increased, the model's performance initially improved, indicating that increasing the number of candidate items can enhance model performance to some extent. However, an excessive number of candidate items may lead to performance degradation. When the number of historical items is set to 20, all three metrics reached higher values, suggesting that this number of historical items effectively improves model performance.

5 Conclusion

With the advancement of Large Language Models (LLMs), their potential in recommendation systems is gradually being recognized. In this work, we explore the feasibility of using LLMs for recommendation in the face of large amounts of multimodal data. The MLLM4Rec framework proposed in this paper significantly improves the performance of sequence recommendation by combining multimodal information and large language models. Specifically, through text-described images and a hybrid cue learning approach, we overcome the gap between text and image modalities and utilize role-playing to reduce the generative illusion problem. Experimental results show that MLLM4Rec achieves excellent recommendation results on different datasets. Future work can further optimize the cue templates and role settings, and explore the fusion of more modal information to further improve the accuracy and usefulness of the recommendation system.



(a) Performance of MLLM4Rec on Toys_Games dataset with different number of candidate items and K=5.

(b) Performance of MLLM4Rec on Toys_Games dataset with different number of candidate items and K=10.

Fig. 4 The performance of MLLM4Rec with different candidate item number on the Toys_Games dataset, with the left graph (a) shows the results for K=5 candidate items, and the right graph (b) shows the results for K=10 candidate items

Author Contributions X. Z. : Primarily responsible for mentoring, providing a lab environment, and guiding the direction of thesis writing. Y. W. : The main writing of the thesis, the experiments. X. S. : English revision, grammar correction.

Data Availability -Amazon review dataset : Availabile at https://cseweb.ucsd.edu/jmcauley/datasets/amazon/links.html-ML-100K dataset : Availabile at https://grouplens.org/datasets/movielens/100k/-BLIP2: Availabile at https://huggingface.co/Salesforce/blip2-opt-2.7b

Code Availability -Availabile at https://github.com/wangyuxiang123/MLLM4Rec.git

Declarations

Competing interests The authors declare no conflict of interest and no financial support or external funding.

References

Achiam, J., Adler, S., & Agarwal, S., et al. (2023). Gpt-4 technical report. arXiv preprint https://doi.org/10.48550/arXiv.2303.08774

Bao, K., Zhang, J., & Zhang, Y.a. (2023). Tallrec: An effective and efficient tuning framework to align large language model with recommendation. In: *Proceedings of the 17th ACM Conference on Recommender Systems*, pp. 1007–1014. https://doi.org/10.1145/3604915.3608857

Boz, A., Zorgdrager, W., & Kotti, Z., et al. (2024). Improving sequential recommendations with llms. arXiv preprint https://doi.org/10.48550/arXiv.2402.01339

Devlin, J., Chang, M.-W., & Lee, K., et al. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint https://doi.org/10.48550/arXiv.1810.04805

Ding, H., Ma, Y., & Deoras, A., et al. (2021). Zero-shot recommender systems. arXiv preprint https://doi.org/10.48550/arXiv.2105.08318

Du, Z., Qian, Y., & Liu, X., et al. (2021). Glm: General language model pretraining with autoregressive blank infilling. arXiv preprint https://doi.org/10.48550/arXiv.2103.10360

Fan, Z., Liu, Z., Wang, Y., et al. (2022). Sequential recommendation via stochastic self-attention. *Proceedings of the ACM Web Conference, 2022*, 2036–2047. https://doi.org/10.1145/3485447.3512077

Gao, A., Qin, J., Ma, C., et al. (2024). Bmdf-sr: bidirectional multi-sequence decoupling fusion method for sequential recommendation. *Journal of Intelligent Information Systems., 62*(2), 485–507. https://doi.org/10.1007/s10844-023-00825-w

Geng, S., Liu, S., & Fu, Z., et al. (2022). Recommendation as language processing (rlp): A unified pretrain, personalized prompt & predict paradigm (p5). In: *Proceedings of the 16th ACM Conference on Recommender Systems*, pp. 299–315. https://doi.org/10.1145/3523227.3546767

He, R., & McAuley, J. (2016). Fusing similarity models with markov chains for sparse sequential recommendation. In: *2016 IEEE 16th International Conference On Data Mining (ICDM)*, pp. 191–200. IEEE. https://doi.org/10.1109/ICDM.2016.0030

He, K., Zhang, X., & Ren, S., et al. (2016). Deep residual learning for image recognition. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778 . https://doi.org/10.48550/arXiv.1512.03385

Hidasi, B., & Karatzoglou, A. (2018). Recurrent neural networks with top-k gains for session-based recommendations. In: *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, pp. 843–852. https://doi.org/10.1145/3269206.3271761

Hidasi, B., Karatzoglou, A., & Baltrunas, L., et al. (2015). Session-based recommendations with recurrent neural networks. arXiv preprint. https://doi.org/10.48550/arXiv.1511.06939

Hidasi, B., Karatzoglou, A., Baltrunas, L., & Tikk, D. (2015). Session-based recommendations with recurrent neural networks. arXiv preprint https://doi.org/10.48550/arXiv.1511.06939

Hou, Y., Mu, S., Zhao, & W.X., et al. (2022). Towards universal sequence representation learning for recommender systems. In: *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 585–593. https://doi.org/10.1145/3534678.3539381

Hou, Y., Zhang, J., & Lin, Z., et al. (2024). Large language models are zero-shot rankers for recommender systems. In: *European Conference on Information Retrieval*, Springer, pp. 364–381. https://doi.org/10.1007/978-3-031-56060-6_24

Hou, Y., He, Z., McAuley, J., et al. (2023). Learning vector-quantized item representation for transferable sequential recommenders. *Proceedings of the ACM Web Conference, 2023*, 1162–1171. https://doi.org/10.1145/3543507.3583434

Ji, J., Li, Z., & Xu, S., et al. (2024). Genrec: Large language model for generative recommendation. In: *European Conference on Information Retrieval*, Springer, pp. 494–502. https://doi.org/10.1007/978-3-031-56063-7_42

Kang, W.-C., & McAuley, J. (2018). Self-attentive sequential recommendation. In: *2018 IEEE International Conference on Data Mining (ICDM)*, IEEE, pp. 197–206. https://doi.org/10.1109/ICDM.2018.00035

Lei, J., Li, Y., Yang, S., et al. (2022). Two-stage sequential recommendation for side information fusion and long-term and short-term preferences modeling. *Journal of Intelligent Information Systems., 59*(3), 657–677. https://doi.org/10.1007/s10844-022-00723-7

Li, J., Ren, P., & Chen, Z., *et al.* (2017). Neural attentive session-based recommendation. In: *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, pp. 1419–1428. https://doi.org/10.1145/3132847.3132926

Li, J., Wang, M., & Li, J., et al. (2023). Text is all you need: Learning language representations for sequential recommendation. In: *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 1258–1267. https://doi.org/10.1145/3580305.3599519

Lin, J., Dai, X., & Xi, Y., et al. (2023). How can recommender systems benefit from large language models: A survey. arXiv preprint. https://doi.org/10.48550/arXiv.2306.05817

Liu, Q., Chen, N., & Sakai, T., et al. (2023). A first look at llm-powered generative news recommendation. arXiv preprint https://doi.org/10.48550/arXiv.2305.06566

Meng, L., Feng, F., & He, X., et al. (2020). Heterogeneous fusion of semantic and collaborative information for visually-aware food recommendation. In: *Proceedings of the 28th ACM International Conference on Multimedia*, pp. 3460–3468. https://doi.org/10.1145/3394171.3413598

Mu, S., Hou, Y., Zhao, & W.X., et al. (2022). Id-agnostic user behavior pre-training for sequential recommendation. In: *China Conference on Information Retrieval*, Springer, pp. 16–27. https://doi.org/10.1007/978-3-031-24755-2_2

Mysore, S., McCallum, A., & Zamani, H. (2023). Large language model augmented narrative driven recommendations. In: *Proceedings of the 17th ACM Conference on Recommender Systems*, pp. 777–783. https://doi.org/10.1145/3604915.3608829

Petrov, A.V., & Macdonald, C. (2023). Generative sequential recommendation with gptrec. arXiv preprint https://doi.org/10.48550/arXiv.2306.11114

Quadrana, M., Karatzoglou, A., & Hidasi, B., et al. (2017). Personalizing session-based recommendations with hierarchical recurrent neural networks. In: *Proceedings of the Eleventh ACM Conference on Recommender Systems*, pp. 130–137. https://doi.org/10.1145/3109859.3109896

Radford, A., & Narasimhan, K. (2018). Improving language understanding by generative pre-training. https://api.semanticscholar.org/CorpusID:49313245

Rendle, S., Freudenthaler, C., & Schmidt-Thieme, L. (2010). Factorizing personalized markov chains for next-basket recommendation. In: *Proceedings of the 19th International Conference on World Wide Web*, pp. 811–820. https://doi.org/10.1145/1772690.1772773

Shi, X., Xue, S., & Wang, K., et al. (2024). Language models can improve event prediction by few-shot abductive reasoning. *Advances in Neural Information Processing Systems 36*. https://doi.org/10.48550/arXiv.2305.16646

Song, K., Sun, Q., & Xu, C., et al. (2023). Self-supervised multi-modal sequential recommendation. arXiv preprint https://doi.org/10.48550/arXiv.2304.13277

Sun, F., Liu, J., & Wu, J., et al. (2019). Bert4rec: Sequential recommendation with bidirectional encoder representations from transformer. In: *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, pp. 1441–1450. https://doi.org/10.1145/3357384.3357895

Tang, J., & Wang, K. (2018). Personalized top-n sequential recommendation via convolutional sequence embedding. In: *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*, pp. 565–573. https://doi.org/10.1145/3159652.3159656

Taori, R., Gulrajani, I., & Zhang, T., et al. (2023) Stanford Alpaca: An Instruction-following LLaMA model. GitHub. https://github.com/tatsu-lab/stanford_alpaca

Touvron, H., Lavril, T., & Izacard, G., et al. (2023). Llama: Open and efficient foundation language models. arXiv preprint https://doi.org/10.48550/arXiv.2302.13971

Touvron, H., Martin, L., & Stone, K., et al. (2023). Llama 2: Open foundation and fine-tuned chat models. arXiv preprint arXiv:2307.09288. https://doi.org/10.48550/arXiv.2307.09288

Vaswani, A., Shazeer, N., Parmar, N., et al. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, *30*. https://doi.org/10.5555/3295222.3295349

Wang, Y., Kordi, Y., & Mishra, S., et al. (2022). Self-instruct: Aligning language models with self-generated instructions. arXiv preprint https://doi.org/10.48550/arXiv.2212.10560

Wang, X., Zhou, K., & Wen, J.-R., et al. (2022). Towards unified conversational recommender systems via knowledge-enhanced prompt learning. In: *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 1929–1937. https://doi.org/10.1145/3534678.3539382

Wang, S., Xu, X., Zhang, X., et al. (2022). Veracity-aware and event-driven personalized news recommendation for fake news mitigation. *Proceedings of the ACM Web Conference, 2022*, 3673–3684. https://doi.org/10.1145/3485447.3512263

Wang, S., Zhang, X., Wang, Y., et al. (2022). Trustworthy recommender systems. *ACM Transactions on Intelligent Systems and Technology*. https://doi.org/10.1145/3627826

Wei, J., Bosma, M., & Zhao, V.Y., et al. (2021). Finetuned language models are zero-shot learners. arXiv preprint https://doi.org/10.48550/arXiv.2109.01652

Wei, Y., Wang, X., & Nie, L., et al. (2019). Mmgcn: Multi-modal graph convolution network for personalized recommendation of micro-video. In: *Proceedings of the 27th ACM International Conference on Multimedia*, pp. 1437–1445. https://doi.org/10.1145/3343031.3351034

Wu, L., Zheng, Z., & Qiu, Z., et al. (2023). A survey on large language models for recommendation. arXiv preprint https://doi.org/10.1007/s11280-024-01291-2

Xi, Y., Liu, W., & Lin, J., et al. (2024). Towards open-world recommendation with knowledge augmentation from large language models. In: *Proceedings of the 18th ACM Conference on Recommender Systems*, pp. 12–22. https://doi.org/10.1145/3640457.3688104

Xie, Y., Gao, J., & Zhou, P., et al. (2023). Rethinking multi-interest learning for candidate matching in recommender systems. In: *Proceedings of the 17th ACM Conference on Recommender Systems*, pp. 283–293. https://doi.org/10.1145/3604915.3608766

Xu, S., Hua, W., & Zhang, Y. (2023). Openp5: An open-source platform for developing, training, and evaluating llm-based recommender systems. arXiv preprint https://doi.org/10.1145/3626772.3657883

Yang, Z., Wu, J., & Luo, Y., et al. (2023). Large language model can interpret latent space of sequential recommender. arXiv preprint https://doi.org/10.48550/arXiv.2310.20487

Yuan, F., Karatzoglou, A., & Arapakis, I., et al. (2019). A simple convolutional generative network for next item recommendation. In: *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*, pp. 582–590. https://doi.org/10.1145/3289600.3290975

Yue, Z., Rabhi, S., & Moreira, G.d.S.P., et al. (2023). Llamarec: Two-stage recommendation using large language models for ranking. arXiv preprint https://doi.org/10.48550/arXiv.2311.02089

Yue, Z., Wang, Y., & He, Z., et al. (2024). Linear recurrent units for sequential recommendation. In: *Proceedings of the 17th ACM International Conference on Web Search and Data Mining*, pp. 930–938. https://doi.org/10.1145/3616855.3635760

Zeng, A., Liu, X., & Du, Z., et al. (2022). Glm-130b: An open bilingual pre-trained model. arXiv preprint https://doi.org/10.48550/arXiv.2210.02414

Zhou, P., Gao, J., & Xie, Y., et al. (2023). Equivariant contrastive learning for sequential recommendation. In: *Proceedings of the 17th ACM Conference on Recommender Systems*, pp. 129–140. https://doi.org/10.1145/3604915.3608786