A Survey on Large Language Models for Recommendation

Likang Wu[1,2*] , **Zhi Zheng**[1,2*] , **Zhaopeng Qiu**[2*] , **Hao Wang**[1†] , **Hongchao Gu**[1] , **Tingjia Shen**[1] , **Chuan Qin**[2] , **Chen Zhu**[2] , **Hengshu Zhu**[2†] , **Qi Liu**[1] , **Hui Xiong**[3†] , **Enhong Chen**[1†]
[1]University of Science and Technology of China, [2]Career Science Lab, BOSS Zhipin, [3]Hong Kong University of Science and Technology (Guangzhou)
{wulk,zhengzhi97,hcgu,jts_stj}@mail.ustc.edu.cn,
{zhpengqiu,chuanqin0426,zc3930155,zhuhengshu}@gmail.com,
{wanghao3,qiliuql,cheneh}@ustc.edu.cn, xionghui@ust.hk

Abstract

Large Language Models (LLMs) have emerged as powerful tools in the field of Natural Language Processing (NLP) and have recently gained significant attention in the domain of Recommendation Systems (RS). These models, trained on massive amounts of data using self-supervised learning, have demonstrated remarkable success in learning universal representations and have the potential to enhance various aspects of recommendation systems by some effective transfer techniques such as fine-tuning and prompt tuning, and so on. The crucial aspect of harnessing the power of language models in enhancing recommendation quality is the utilization of their high-quality representations of textual features and their extensive coverage of external knowledge to establish correlations between items and users. To provide a comprehensive understanding of the existing LLM-based recommendation systems, this survey presents a taxonomy that categorizes these models into two major paradigms, respectively Discriminative LLM for Recommendation (DLLM4Rec) and Generative LLM for Recommendation (GLLM4Rec), with the latter being systematically sorted out for the first time. Furthermore, we systematically review and analyze existing LLM-based recommendation systems within each paradigm, providing insights into their methodologies, techniques, and performance. Additionally, we identify key challenges and several valuable findings to provide researchers and practitioners with inspiration. We have also created a GitHub repository to index relevant papers on LLMs for recommendation[§].

1 Introduction

Recommendation systems play a critical role in assisting users in finding relevant and personalized items or content. With the emergence of Large Language Models (LLMs) in Natural Language Processing (NLP), there has been a growing interest in harnessing the power of these models to enhance recommendation systems.

The key advantage of incorporating LLMs into recommendation systems lies in their ability to extract high-quality representations of textual features and leverage the extensive external knowledge encoded within them [Liu *et al.*, 2023b]. And this survey views LLM as the Transformer-based model with a large number of parameters, trained on massive datasets using self/semi-supervised learning techniques, e.g., BERT, GPT series, PaLM series, etc[‡]. Different from traditional recommendation systems, the LLM-based models excel in capturing contextual information, comprehending user queries, item descriptions, and other textual data more effectively [Geng *et al.*, 2022]. By understanding the context, LLM-based RS can improve the accuracy and relevance of recommendations, leading to enhanced user satisfaction. Meanwhile, facing the common data sparsity issue of limited historical interactions [Da'u and Salim, 2020], LLMs also bring new possibilities to recommendation systems through zero/few-shot recommendation capabilities [Sileo *et al.*, 2022]. These models can generalize to unseen candidates due to the extensive pre-training with factual information, domain expertise, and common-sense reasoning, enabling them to provide reasonable recommendations even without prior exposure to specific items or users.

The aforementioned strategies are already well-applied in discriminative models. However, with the evolution of AI learning paradigms, generative language models have started to gain prominence [Zhao *et al.*, 2023]. A prime example of this is the emergence of ChatGPT and other comparable models, which have significantly disrupted human life and work patterns. Furthermore, the fusion of generative models with recommendation systems offers the potential for even more innovative and practical applications. For instance, the interpretability of recommendations can be improved, as LLM-based systems are able to provide explanations based on their language generation capabilities [Gao *et al.*, 2023], helping users understand the factors influencing the recommendations. Moreover, generative language models enable more personalized and context-aware recommendations, such

---

*Equal Contribution.
†Corresponding Author.

‡https://en.wikipedia.org/wiki/Large_language_model
§https://github.com/WLiK/LLM4Rec

as users' customizable prompts [Li *et al.*, 2023b] in the chat-based recommendation system, enhancing user engagement and satisfaction with the diversity of results.

Motivated by the remarkable effectiveness of the aforementioned paradigms in solving data sparsity and efficiency issues, the adaptation of language modeling paradigms for recommendation has emerged as a promising direction in both academia and industry, significantly advancing the state-of-the-art in the research of recommendation systems. So far, there are a few studies that review relevant papers in this domain [Zeng *et al.*, 2021; Liu *et al.*, 2023b]. Zeng *et al.* (2021) summarizes some research on the pre-training of recommendation models and discusses knowledge transfer methods between different domains. Liu *et al.* (2023b) proposes an orthogonal taxonomy to divide existing pre-trained language model-based recommendation systems w.r.t. their training strategies and objectives, analyzes and summarizes the connection between pre-trained language model-based training paradigms and different input data types. However, both of these surveys primarily focus on the transfer of training techniques and strategies in pretraining language models, rather than exploring the potential of language models and their capabilities, i.e., LLM-based way. Additionally, they lack a comprehensive overview of the recent advancements and systematic introductions of generative large language models in the recommendation field. To address this issue, we delve into LLM-based recommendation systems, categorizing them into discriminative LLMs for recommendation and generative LLMs for recommendation, and the focus of our review is on the latter. To the best of our knowledge, our survey is the first work that concludes an up-to-date and comprehensive review of generative large language models for recommendation systems. The main contributions of our survey are summarized as follows:

- We present a systematic survey of the current state of LLM-based recommendation systems, focusing on expanding the capacity of language models. By analyzing the existing methods, we provide a systematic overview of related advancements and applications.
- To the best of our knowledge, our survey is the first comprehensive and up-to-date review specifically dedicated to generative large language models for recommendation systems.
- From the perspective of modeling paradigms, we categorize the current studies of large language model recommendations into three distinct schools of thought. Any existing method can be fittingly placed within these categories, thereby providing a clear and organized overview of this burgeoning field.
- Our survey critically analyzes the advantages, disadvantages, and limitations of existing methods. We identify key challenges faced by LLM-based recommendation systems and propose valuable findings that can inspire further research in this potential field.

2 Modeling Paradigms and Taxonomy

The basic framework of all large language models is composed of several transformer blocks, e.g., GPT, PaLM, LLaMA, etc. The input of this architecture is generally composed of token embeddings or position embeddings and so on, while the expected output embedding or tokens can be obtained at the output module. Here, both the input and output data types are textual sequences. As shown in (1)-(3) in Figure 1, for the adaption of language models in recommendations, i.e., the modeling paradigm, existing work can be roughly divided into the following three categories:

(1) **LLM Embeddings + RS**. This modeling paradigm views the language model as a feature extractor, which feeds the features of items and users into LLMs and outputs corresponding embeddings. A traditional RS model can utilize knowledge-aware embeddings for various recommendation tasks.

(2) **LLM Tokens + RS**. Similar to the former method, this approach generates tokens based on the inputted items' and users' features. The generated tokens capture potential preferences through semantic mining, which can be integrated into the decision-making process of a recommendation system.

(3) **LLM as RS**. Different from (1) and (2), this paradigm aims to directly transfer pre-trained LLM into a powerful recommendation system. The input sequence usually consists of the profile description, behavior prompt, and task instruction. The output sequence is expected to offer a reasonable recommendation result.

In practical applications, the choice of language model significantly influences the design of modeling paradigms in recommendation systems. As shown in Figure 2, in this paper, we categorize existing works into two main categories, respectively discriminative LLMs and generative LLMs for recommendation. The taxonomy of LLMs for recommendation can be further subdivided based on the training manner, and the distinction among different manners is illustrated in Figure 3. Generally, discriminative language models are well-suited for embedding within the paradigm (1), while the response generation capability of generative language models further supports paradigms (2) or (3).

3 Discriminative LLMs for Recommendation

Indeed, so-called discriminative language models in the recommendation area mainly refer to those models of BERT series [Devlin *et al.*, 2019]. Due to the expertise of discriminative language models in natural language understanding tasks, they are often considered as embedding backbones for downstream tasks. This holds true for recommendation systems as well. Most existing works align the representations of pre-trained models like BERT with the domain-specific data through fine-tuning. Additionally, some research explores training strategies like prompt tuning. The representative approaches and common-used datasets are listed in Table 1 and Table 2.

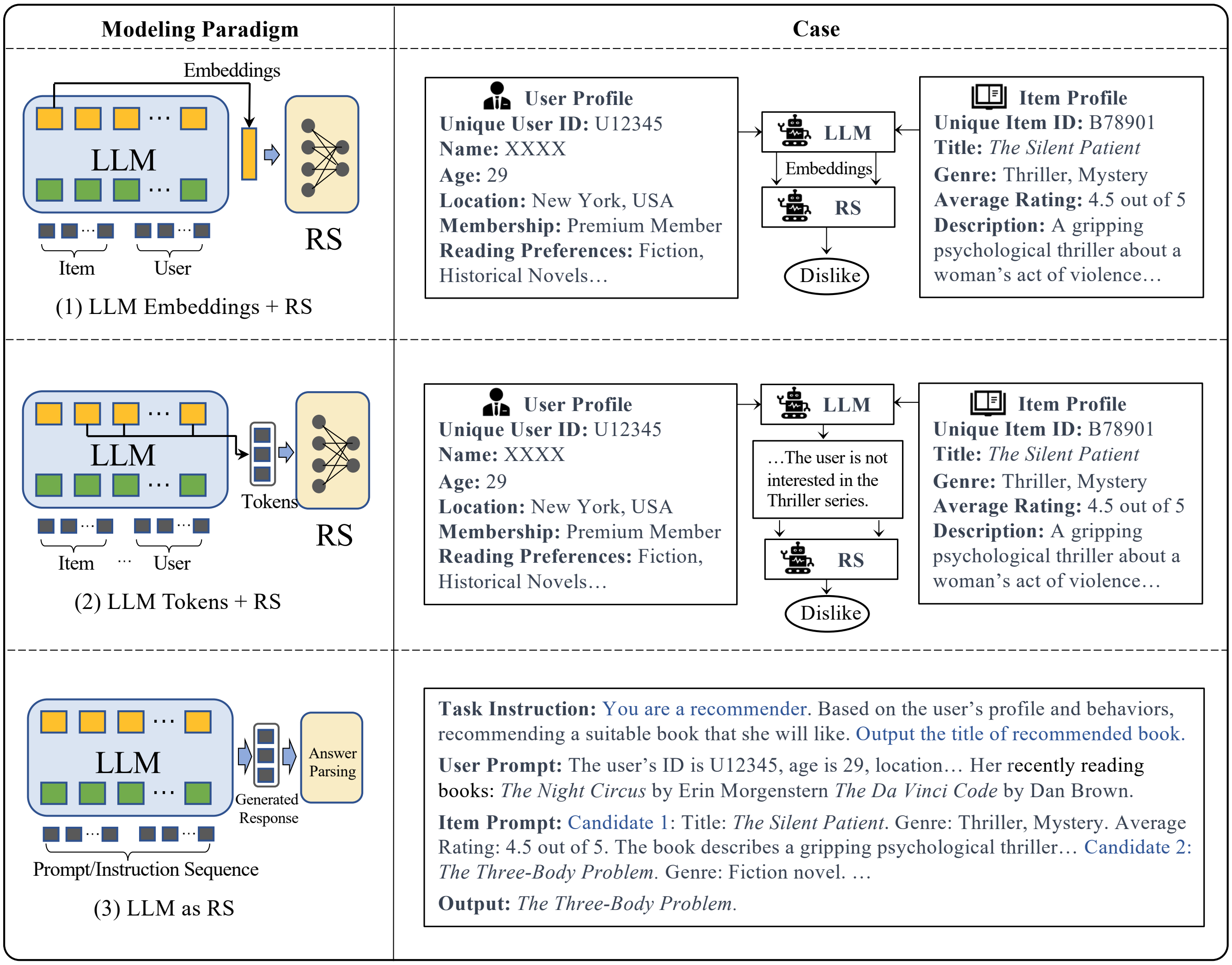

Figure 1: Three modeling paradigms of the research for large language models on recommendation systems.

3.1 Fine-tuning

Fine-tuning pre-trained language models is a universal technique that has gained significant attention in various natural language processing (NLP) tasks, including recommendation systems. The idea behind fine-tuning is to take a language model, which has already learned rich linguistic representations from large-scale text data, and adapt it to a specific task or domain by further training it on task-specific data.

The process of fine-tuning involves initializing the pre-trained language model with its learned parameters and then training it on a recommendation-specific dataset. This dataset typically includes user-item interactions, textual descriptions of items, user profiles, and other relevant contextual information. During fine-tuning, the model's parameters are updated based on the task-specific data, allowing it to adapt and specialize for recommendation tasks. The learning objectives can be different in the pre-training and fine-tuning stages.

Since the fine-tuning strategy is flexible, most bert-enhanced recommendation methods can be summarized into this track. For the basic representation task, Qiu *et al.* (2021) proposed a novel pre-training and fine-tuning-based approach U-BERT to learn users' representation, which leveraged content-rich domains to complement those users' feature with insufficient behavior data. A review co-matching layer is designed to capture implicit semantic interactions between the reviews of users and items. Similarly, in UserBERT [Wu *et al.*, 2021b], two self-supervision tasks are incorporated for user model pre-training on unlabeled behavior data to empower user modeling. This model utilizes medium-hard contrastive learning, masked behavior prediction, and behavior sequence matching to train accurate user representation via captured inherent user interests and relatedness.

The pre-trained BERT achieved outstanding breakthroughs in the ranking task as well. BECR [Yang *et al.*, 2022] proposed a lightweight composite re-ranking scheme that combined deep contextual token interactions and traditional lex-

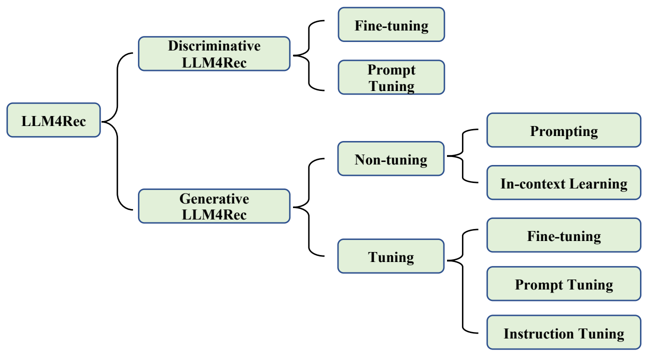

Figure 2: A taxonomy of the research for large language models on recommendation systems.

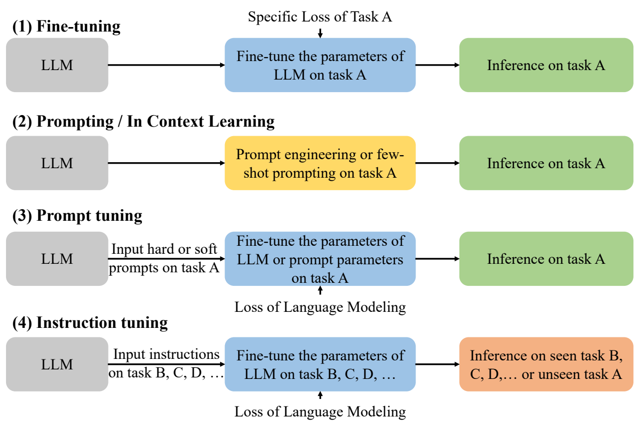

Figure 3: Detailed explanation of five different training (domain adaption) manners for LLM-based recommendations.

ical term-matching features at the same time. With a novel composite token encoding, BECR effectively approximates the query representations using pre-computable token embeddings based on uni-grams and skip-n-grams, allowing for a reasonable tradeoff between ad-hoc ranking relevance and efficiency. Besides, Wu *et al.* (2022) proposed an end-to-end multi-task learning framework for product ranking with fine-tuned domain-specific BERT to address the issue of vocabulary mismatch between queries and products. The authors utilized the mixture-of-experts layer and probability transfer between tasks to harness the abundant engagement data.

There are also many related studies in other specific tasks or scenarios, e.g., group recommendation [Zhang *et al.*, 2022a], search/matching [Yao *et al.*, 2022], CTR prediction [Muhamed *et al.*, 2021]. Especially, the "pre-train, fine-tuning" mechanism played an important role in several sequential or session-based recommendation systems, such as BERT4Rec [Sun *et al.*, 2019], RESETBERT4Rec [Zhao, 2022]. However, the above models only leveraged the advantages of the training strategy rather than expanding the large language model into the recommendation field, so it was not the focus of our discussion. The sequence representation learning model UniSRec [Hou *et al.*, 2022] developed a BERT-fine-tuned framework, which associated description text of items to learn transferable representations across different recommendation scenarios. For the content-based recommendation, especially news recommendation, NRMS [Wu *et al.*, 2021a], Tiny-NewsRec [Yu *et al.*, 2022], PREC [Liu *et al.*, 2022], exploited large language models to empower news recommendation via handling known domain shift problems or reducing transfer cost. Specifically, to answer the crucial question that *Can a purely modality-based recommendation model (MoRec) outperforms or matches a pure ID-based model (IDRec) by replacing the itemID embedding with a SOTA modality encoder?*, Yuan *et al.* (2023) conducted large-scale experiments and found that modern MoRec could already perform on par or better than IDRec with the typical recommendation architecture (i.e., Transformer backbone) even in the non-cold-start item recommendation setting with the SOTA and E2E-trained Modality Encoder. The subsequent exploration [Li *et al.*, 2023c] based on larger-scale language model encoders, e.g. OPT [Zhang *et al.*, 2022b], further validated the viewpoint.

In summary, the integration of BERT fine-tuning into recommendation systems fuses the powerful external knowledge and personalized user preference, which primarily aims to promote recommendation accuracy and simultaneously obtains a little cold-start handling capability for new items with limited historical data.

3.2 Prompt Tuning

Instead of adapting LLMs to different downstream recommendation tasks by designing specific objective functions, prompt tuning [Lester *et al.*, 2021] tries to align the tuning object of recommendation with pre-trained loss through hard/soft prompts and label word verbalizer. For example, Penha and Hauff (2020) leveraged BERT's Masked Language Modeling (MLM) head to uncover its understanding of item genres using cloze-style prompts. They further utilized BERT's Next Sentence Prediction (NSP) head and similarity (SIM) of representations to compare relevant and non-relevant search and recommendation query-document inputs. The experiment told that BERT, without any fine-tuning, can prioritize relevant items in the ranking process. Yang *et al.* (2021) developed a conversational recommendation system with prompts, where a BERT-based item encoder directly mapped the metadata of each item to an embedding. Recently, Prompt4NR [Zhang and Wang, 2023] pioneered the application of the prompt learning paradigm for news recommendation. This framework redefined the objective of predicting user clicks on candidate news as a cloze-style mask-prediction task. The experiments found that the performance of recommendation systems is noticeably enhanced through the utilization of multi-prompt ensembling, surpassing the results achieved with a single prompt on discrete and continuous templates. This highlights the effectiveness of prompt ensembling in combining multiple prompts to make more informed decisions.

4 Generative LLMs for Recommendation

Compared to discriminative models, generative models have better natural language generation capabilities. Therefore, unlike most discriminative model-based approaches that align the representation learned by LLMs to the recommendation domain, most generative model-based work translates recommendation tasks as natural language tasks, and then applies techniques such as in-context learning, prompt tuning, and instruction tuning to adapt LLMs to directly generate the recommendation results. Moreover, with the impressive capabilities demonstrated by ChatGPT, this type of work has received increased attention recently.

As shown in Figure 2, according to whether tuning parameters, these generative LLM-based approaches can be further subdivided into two paradigms: *non-tuning paradigm* and *tuning paradigm*. **Here the tuning/non-tuning target denotes the used LLM module in the following methods.** The following two sub-sections will address their details, respectively. The representative approaches and common-used datasets are also listed in Table 1 and Table 2.

4.1 Non-tuning Paradigm

The LLMs have shown strong zero/few-shot abilities in many unseen tasks [Brown *et al.*, 2020; Ouyang *et al.*, 2022]. Hence, some recent works assume LLMs already have the recommendation abilities, and attempt to trigger these abilities by introducing specific prompts. They employ the recent practice of Instruction and In-Context Learning [Brown *et al.*, 2020] to adopt the LLMs to recommendation tasks without tuning model parameters. According to whether the prompt includes the demonstration examples, the studies in this paradigm mainly belong to the following two categories: *prompting* and *in-context learning*.

Prompting

This category of works aims to design more suitable instructions and prompts to help LLMs better understand and solve the recommendation tasks. Liu *et al.* (2023a) systematically evaluated the performance of ChatGPT on five common recommendation tasks, i.e., *rating prediction*, *Sequential Recommendation*, *direct recommendation*, *explanation generation*, and *review summarization*. They proposed a general recommendation prompt construction framework, which consists of: (1) task description, adapting recommendation tasks to natural language processing tasks; (2) behavior injection, incorporating user-item interaction to aid LLMs in capturing user preferences and needs; (3) format indicator, constraining the output format and making the recommendation results more comprehensible and assessable. Similarly, Dai *et al.* (2023) conducted an empirical analysis of ChatGPT's recommendation abilities on three common information retrieval tasks, including point-wise, pair-wise, and list-wise ranking. They proposed different prompts for different kinds of tasks and introduced the *role* instructions (such as *You are a news recommendation system now.*) at the beginning of the prompts to enhance the domain adaption ability of ChatGPT.

To evaluate the enhancement of different prompting inputs, Sanner *et al.* (2023) designed three prompt templates for the case of Items only (the attribute of items), Language only (the description of user's preference), and combined Language+Items in their experiments. After analyzing the performance of language models, they have discovered that zero-shot and few-shot strategies are highly effective for making recommendations based solely on language-based preferences (without considering item preferences). In fact, these strategies have proven to be remarkably competitive in comparison to item-based collaborative filtering methods, particularly in near cold-start scenarios. Meanwhile, to summarize the user's intention by prompt based on their interaction data, MINT [Mysore *et al.*, 2023] employed 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 on both the synthetic queries and user items. The results indicate that the resulting models outperformed several strong baseline models and ablated models. In a 1-shot setup, these models matched or even outperformed a 175B LLM that was directly used for narrative-driven recommendation. However, these methods have not considered decomposing the topics in a textual description, which would result in noisy and target-unclear prompts. KAR [Xi *et al.*, 2023] solved this issue by introducing factorization prompting to elicit accurate reasoning on user preferences and factual knowledge.

Instead of proposing a general framework, some works focus on designing effective prompts for specific recommendation tasks. Sileo *et al.* (2022) mined the movie recommendation prompts from the pre-training corpus of GPT-2. Hou *et al.* (2023) introduced two prompting methods to improve the sequential recommendation ability of LLMs: *recency-focused sequential prompting*, enabling LLMs to perceive the sequential information in the user interaction history, and *bootstrapping*, shuffling the candidate item list multiple times and taking the average scores for ranking to alleviate the position bias problem. Due to the limited number of input tokens allowed for the LLMs, it's hard to input a long candidate list in the prompt. To solve this problem, Sun *et al.* (2023) proposed a sliding window prompt strategy, which only ranks the candidates in the window each time, then slides the window in back-to-first order, and finally repeat this process multiple times to obtain the overall ranking results.

In addition to taking LLMs as recommendation systems, some studies also utilize LLMs to construct model features. GENRE [Liu *et al.*, 2023c] introduced three prompts to employ LLMs to conduct three feature enhancement sub-tasks for news recommendation. Specifically, it used ChatGPT to refine the news titles according to the abstract, extract profile keywords from the user reading history, and generate synthetic news to enrich user historical interactions. By incorporating these features constructed by LLMs, the traditional news recommendation models can be improved significantly. Similarly, NIR [Wang and Lim, 2023] designed two prompts to generate user preference keywords and extract representative movies from user interaction history to improve the movie recommendation.

In practice, in addition to the ranking model, the whole recommendation system generally consists of multiple import components, such as content database, candidate retrieval model, etc. Hence, another line of using LLMs for recommendation is taking them as the controllers of the whole system. ChatREC [Gao *et al.*, 2023] designed an interactive recommendation framework around ChatGPT, which understands user requirements through multi-turn dialogues, and calls existing recommendation systems to provide results. In addition, ChatGPT can control the database to retrieve relevant content to supplement the prompt and address the cold-start item problem. GeneRec [Wang *et al.*, 2023b] proposed a generative recommendation framework and used LLMs to control when to recommend existing items or to generate new items by AIGC models. What's more, RecAgent [Wang *et al.*, 2023a] further utilized LLM as intelligent simulator to develop a virtual recommendation environment. The simulator consists of two main modules: the user module and the recommender module. The user module enables browsing the recommendation site, interaction with other users, and posting on social media. The recommender module offers tailored search and recommendation lists, supporting various model designs for recommendation. Users in the environment take actions based on LLMs and can evolve organically, mirroring real-world behaviors. This project shows potential utilization across several applications, such as simulating the feedback for RL-based recommendations and tracking information dissemination process among the users on social media.

In summary, these studies utilize natural language prompts to activate the zero-shot capability of LLM in recommendation tasks, providing a low-cost and practical solution.

In-context Learning

In-context learning is a technique used by GPT-3 and other LLMs to quickly adapt to new tasks and information. With a few demonstration input-label pairs, they can predict the label for an unseen input without additional parameter updates [Dai *et al.*, 2022]. Hence, some works attempt to add demonstration examples in the prompt to make LLMs better understand the recommendation tasks. For sequential recommendation, Hou *et al.* (2023) introduced demonstration examples by augmenting the input interaction sequence itself. In detail, they paired the prefix of the input interaction sequence and the corresponding successor as examples. Liu *et al.* (2023a) and Dai *et al.* (2023) designed the demonstration example templates for various recommendation tasks and the experimental results also showed the in-context learning method will improve the recommendation abilities of LLMs on most tasks. In addition, a suitable demonstration can be used to control the output format and content of the LLM [Wang *et al.*, 2023c], which can improve the regular evaluation metric. This is crucial for developing a stable and robust recommender system.

However, in comparison to prompting, only a few studies have explored the use of In-context Learning of Language Models (LLMs) in recommendation tasks. Numerous open questions remain, including the selection of demonstration examples and the influence of the number of demonstration examples on recommendation performance.

4.2 Tuning Paradigm

As we mentioned above, LLMs have strong zero/few-shot abilities, and their recommendation performance can significantly surpass random guessing with appropriate prompt design. However, it is not surprising that recommendation systems constructed in this manner fail to surpass the performance of recommendation models trained specifically for a given task on specific data. Therefore, many researchers aim to enhance the recommendation ability of LLMs by further fine-tuning or prompt learning. In this paper, we categorize the paradigm of the tuning methods into three different types, respectively fine-tuning, prompt tuning, and instruction tuning. Specifically, in the fine-tuning paradigm, the usage methods for discriminative and generative large language models are notably similar. The LLMs mainly serve as encoders to extract representations of users or items, and the parameters of the LLMs are subsequently fine-tuned on the specific loss functions of downstream recommendation tasks. Meanwhile, in the prompt tuning and instruction tuning paradigms, the output of the large models is consistently textual, and their parameters are trained using the loss of language modeling. The primary distinction between the prompt tuning and instruction tuning training paradigms is that prompt tuning predominantly focuses on a specific task, e.g., rating prediction, while the LLMs are trained for multiple tasks with different types of instructions under the instruction tuning paradigm. Therefore, the LLMs can get better zero-shot abilities by instruction tuning. In the subsequent sections, we will delve into representative works of these three paradigms in detail.

Fine-tuning

Since under the fine-tuning paradigm, the utilization and training methodologies of generative LLMs are fundamentally similar to the discriminative LLMs discussed in Section 3.1, therefore, we will only introduce a few representative works in this subsection. For example, Petrov and Macdonald (2023) proposed GPTRec, which is a generative sequential recommendation model based GPT-2. In contrast with BERT4Rec, which is based on discriminative LLM, GPTRec is based on generative LLM, uses SVD Tokenisation for memory efficiency, and more flexible using the Next-K generation strategy. Kang *et al.* (2023) proposed to format the user historical interactions as prompts, where each interaction is represented by information about the item, and formulated the rating prediction task as two different tasks, respectively multi-class classification and regression. Kang *et al.* (2023) further investigated various LLMs in different sizes, ranging from 250M to 540B parameters and evaluate their performance in zero-shot, few-shot, and fine-tuning scenarios, and found that FLAN-T5-XXL (11B) model with fine-tuning can achieve the best result. Li *et al.* (2023c) studied the influence of LLMs, such as GPT-3 with 175-billion parameters, on text-based collaborative filtering (TCF). Li *et al.* found that using more powerful LLMs as text encoders can result in higher recommendation accuracy. However, an extremely large LM may not result in a universal representation of users and items, and the simple ID-based collaborative filtering still remains a highly competitive approach in the warm item recommendation setting.

Prompt Tuning

In this paradigm, LLMs typically take the user/item information as input, and output the user preference (e.g., like or unlike, ratings) for the items, or output items that the user may be interested in. For example, Bao *et al.* (2023) proposed TALLRec which is trained by two tuning stages. Specifically, TALLRec is first fine-tuned based on the self-instruct data by Alpaca [Taori *et al.*, 2023]. Then, TALLRec is further fine-tuned by recommendation tuning, where the input is the historical sequence of users and the output is the "yes or no" feedback. Ji *et al.* (2023) presented an LLM-based generative recommendation method named GenRec that utilized the generation ability of generative LLM to directly generate the target item to recommend. Specifically, Ji *et al.* proposed to use input generation function to convert items into prompts, and use LLMs to generated the next item. Chen (2023) proposed a multi-step approach to harness the potential of LLMs for recommendation. Specifically, Chen first proposed to leverage LLMs to generate a summary of a user's preferences. For example, by analyzing a user's music and TV viewing history, the LLM can generate a summary like "pop music" and "fantasy movies." Then, a retrieval module is utilized to get a much smaller candidate pool. Finally, the interaction history, natural language user profile and retrieved candidates are utilized to construct a natural language prompt that can be fed into the LLM for recommendation.

The aforementioned methods are recommendations for general tasks using large language models. However, as previously mentioned, a significant advantage of large language models is their ability to efficiently align model parameters with specific domains. Currently, the domain where this has been most extensively explored is online recruitment scenarios. Specifically, within the realm of job-resume matching, the generative recommendation model GIRL [Zheng *et al.*, 2023] pioneers the use of LLM to generate potential job descriptions (JDs), enhancing the explainability and appropriateness of recommendations. GLRec [Wu *et al.*, 2023] introduced the meta-path prompt constructor, a novel approach that employed LLM recommenders to interpret behavior graphs. This method also incorporated a path augmentation module to mitigate prompt bias. Subsequently, an LLM-based framework was introduced to align unpaired low-quality resumes with high-quality generated ones using Generative Adversarial Networks (GANs). This alignment process refined resume representations, leading to improved recommendation outcomes [Du *et al.*, 2023].

Expect the above works in online recruitment scenarios, there are also several other works about how to leverage the potent generative capabilities of large models to accomplish specific tasks. For example, Jin *et al.* (2023) proposed to generate the title of the next product of interest for the user with the help of LLMs. They fine-tune a mT5 model using a generative objective defined on their dataset. However, a simple heuristic method which takes the last product title as the result, surpasses the performance of the fine-tuned language model. Friedman *et al.* (2023) proposed RecLLM, which contains a dialogue management module uses an LLM to converse with the user, a ranker module uses an LLM to match the user preferences, and a controllable LLM-based user simulator to generate synthetic conversations for tuning system modules. Li *et al.* (2023e) proposed PBNR, which can describe user behaviors and news textually in the designed prompts. Specifically, the personalized prompts are created by designing input-target templates, wherein the relevant fields in the prompts are replaced with corresponding information from the raw data. To enhance the performance of LLMs on the recommendation task, PBNR incorporates the ranking loss and the language generation loss throughout the training. Li *et al.* (2023a) proposed to regard the recommendation task as a query generation + searching problem. They further utilized the LLMs to produce diverse and interpretable user interests representations, i.e., the queries.

In addition to directly fine-tuning the LLMs, some studies also proposed to utilize prompt learning to achieve better performance. For example, Wang *et al.* (2022) designed a unified conversational recommendation system named UniCRS based on knowledge-enhanced prompt learning. In this paper, the authors proposed to freeze the parameters of LLMs, and trained the soft prompts for response generation and item recommendation by prompt learning. Li *et al.* (2023b) proposed to provide user-understandable explanations based on the generation ability of LLMs. The authors tried both discrete prompt learning and continuous prompt learning, and further proposed two training strategies, respectively sequential tuning and recommendation as regularization.

Instruction Tuning

In this paradigm, LLMs are fine-tuned for multiple tasks with different types of instructions. In this way, LLMs can better align with human intent and achieve better zero-shot ability. For example, Geng *et al.* (2022) proposed to fine-tune a T5 model on five different types of instructions, respectively sequential recommendation, rating prediction, explanation generation, review summarization, and direct recommendation. After the multitask instruction tuning on recommendation datasets, the model can achieve the capability of zero-shot generalization to unseen personalized prompts and new items. Similarly, Cui *et al.* (2022) proposed to fine-tune an M6 model on three types of tasks, respectively scoring tasks, generation tasks and retrieval tasks. Zhang *et al.* (2023b) first designed a general instruction format from three types of key aspects, respectively preference, intention and task form. Then, Zhang *et al.* (2023b) manually designed 39 instruction templates and automatically generated a large amount of user-personalized instruction data for instruction tuning on a 3B FLAN-T5-XL model. The experiment results demonstrated that this approach can outperform several competitive baselines including GPT-3.5.

5 Findings

In this survey, we systematically reviewed the application paradigms and adaptation strategies of large language models in recommendation systems, especially for generative language models. We have identified their potential to improve the performance of traditional recommendation models in specific tasks. However, it is necessary to note that the overall exploration in this field is still in the early stage. Researchers may find it challenging to determine the most worthwhile problems and pain points to investigate. To address this, we have summarized the common findings presented by numerous studies on large-scale model recommendations. These findings highlight certain technical challenges and present potential opportunities for further advancements in the field.

Table 1: A list of representative LLM-based recommendation methods and their features. Note that, here the target of tuning/non-tuning denotes the used LLM module in the following methods.

Adaption Way	Paper	Base Model	Recommendation Task	Modeling Paradigm	Source Code
Discriminative LLMs for Recommendation					
Fine-tuning	[Wu *et al.*, 2021a]	BERT/RoBERTa/UniLM	News Recommendation	LLM Embeddings + RS	https://shorturl.at/ciny7
	[Qiu *et al.*, 2021]	BERT	User Representation	LLM Embeddings + RS	N/A
	[Zhang *et al.*, 2022a]	BERT	Group Recommendation	LLM as RS	N/A
	[Yao *et al.*, 2022]	BERT	Search/Matching	LLM Embeddings + RS	https://shorturl.at/suJ69
	[Muhamed *et al.*, 2021]	BERT	CTR Prediction	LLM Embeddings + RS	N/A
	[Xiao *et al.*, 2022]	BERT/RoBERTa	Conversational RS	LLM Embeddings + RS	https://shorturl.at/vSUZ8
Prompt Tuning	[Zhang and Wang, 2023]	BERT	Sequential Recommendation	LLM as RS	https://shorturl.at/ehOT0
	[Yang *et al.*, 2021]	DistilBERT/GPT-2	Conversational RS	LLM as RS	https://shorturl.at/gkuxz
	[Penha and Hauff, 2020]	BERT	Conversational RS	LLM as RS	https://shorturl.at/mqzEY
Generative LLMs for Recommendation					
Non-tuning	[Liu *et al.*, 2023c]	ChatGPT	News Recommendation	LLM Tokens + RS	https://shorturl.at/jkFST
	[Wang *et al.*, 2022]	DialoGPT/RoBERTa	Converational RS	LLM Tokens + RS / LLM as RS	https://shorturl.at/isEU8
	[Sileo *et al.*, 2022]	GPT-2	Sequential Recommendation	LLM as RS	https://shorturl.at/EJK29
	[Wang and Lim, 2023]	GPT-3.5	Sequential Recommendation	LLM Tokens + RS / LLM as RS	https://shorturl.at/qKU38
	[Gao *et al.*, 2023]	ChatGPT/GPT-3.5	Sequential Recommendation	LLM as RS	N/A
	[Wang *et al.*, 2023b]	ChatGPT	Generative Recommendation	LLM as RS	https://shorturl.at/dBEP5
	[Hou *et al.*, 2023]	ChatGPT	Sequential Recommendation	LLM as RS	https://shorturl.at/KM056
	[Sun *et al.*, 2023]	ChatGPT/GPT-3.5	Passage Reranking	LLM as RS	https://shorturl.at/eAFY8
	[Qin *et al.*, 2023]	T5/GPT-3.5/GPT-4	Passage Reranking	LLM as RS	N/A
	[Liu *et al.*, 2023a]	ChatGPT	Five Tasks	LLM as RS	N/A
	[Dai *et al.*, 2023]	ChatGPT/GPT-3.5	Sequential Recommendation	LLM as RS	https://shorturl.at/iqtE3
	[Xi *et al.*, 2023]	ChatGLM	CTR Prediction	LLM Tokens + RS	https://shorturl.at/dghEX
	[Wang *et al.*, 2023a]	ChatGPT	Recommendation Agent	LLM Tokens + RS	https://shorturl.at/lqGY1
Tuning	[Zhang *et al.*, 2023b]	FLAN-T5	Three Tasks	LLM as RS	N/A
	[Kang *et al.*, 2023]	FLAN-T5/ChatGPT	Rating Prediction	LLM as RS	N/A
	[Bao *et al.*, 2023]	LLaMA-7B	Movie/Book RS	LLM as RS	https://shorturl.at/coEL1
	[Li *et al.*, 2023b]	GPT-2	Explainable RS	LLM as RS	https://shorturl.at/adT09
	[Geng *et al.*, 2022]	T5	Five Tasks	LLM as RS	https://shorturl.at/CRY19
	[Cui *et al.*, 2022]	M6	Five Tasks	LLM as RS	N/A
	[Wu *et al.*, 2023]	BELLE	Job Recommendation	LLM as RS	N/A
	[Zheng *et al.*, 2023]	BELLE	Generative Recommendation	LLM Tokens +RS	N/A
	[Mao *et al.*, 2023]	UniTRecAU	Text-based Recommendation	LLM as RS	https://shorturl.at/knBNP
	[Li *et al.*, 2023c]	OPT	Text-based Recommendation	LLM Embeddings +RS	N/A
	[Li *et al.*, 2023d]	RoBERTa/GLM	CTR Prediction	LLM Embeddings +RS	N/A

Table 2: A list of common datasets used in existing LLM-based recommendation methods.

Name	Scene	Tasks	Information	URL
Amazon Review	Commerce	Seq Rec/CF Rec	This is a large crawl of product reviews from Amazon. Ratings: 82.83 million, Users: 20.98 million, Items: 9.35 million, Timespan: May 1996 - July 2014	http://jmcauley.ucsd.edu/data/amazon/
Amazon-M2	Commerce	Seq Rec/CF Rec	A large dataset of anonymized user sessions with their interacted products collected from multiple language sources at Amazon. It includes 3,606,249 train sessions, 361,659 test sessions, and 1,410,675 products.	https://arxiv.org/abs/2307.09688
Steam	Game	Seq Rec/CF Rec	Reviews represent a great opportunity to break down the satisfaction and dissatisfaction factors around games. Reviews: 7,793,069, Users: 2,567,538, Items: 15,474, Bundles: 615	https://cseweb.ucsd.edu/~jmcauley/datasets.html#steam_data
MovieLens	Movie	General	The dataset consists of 4 sub-datasets, which describe users' ratings to moives and free-text tagging activities from MovieLens, a movie recommendation service.	https://grouplens.org/datasets/movielens/
Yelp	Commerce	General	There are 6,990,280 reviews, 150,346 businesses, 200,100 pictures, 11 metropolitan areas, 908,915 tips by 1,987,897 users. Over 1.2 million business attributes like hours, parking, availability, etc.	https://www.yelp.com/dataset
Douban	Movie, Music, Book	Seq Rec/CF Rec	This dataset includes three domains, i.e., movie, music, and book, and different kinds of raw information, i.e., ratings, reviews, item details, user profiles, tags (labels), and date.	https://paperswithcode.com/dataset/douban
MIND	News	General	MIND contains about 160k English news articles and more than 15 million impression logs generated by 1 million users. Every news contains textual content including title, abstract, body, category, and entities.	https://msnews.github.io/assets/doc/ACL2020_MIND.pdf
U-NEED	Commerce	Conversation Rec	U-NEED consists of 7,698 fine-grained annotated pre-sales dialogues, 333,879 user behaviors, and 332,148 product knowledge tuples.	https://github.com/LeeeeoLiu/U-NEED

5.1 Model Bias

Position Bias. In the generative language modeling paradigm of recommendation systems, various information such as user behavior sequences and recommended candidates are input to the language model in the form of textual sequential descriptions, which can introduce some position biases inherent in the language model itself [Lu *et al.*, 2021]. For example, the order of candidates affects the ranking results of LLM-based recommendation models, i.e., LLM often prioritizes the items in the top order. And the model usually cannot capture the behavior order of the sequence well. Hou *et al.* (2022) used the random sampling-based bootstrapping to alleviate the position bias of candidates and emphasized the recently interacted items to enhance behavior order. However, these solutions are not adaptive enough, and more robust learning strategies are needed in the future.

Popularity Bias. The ranking results of LLMs are influenced by the popularity levels of the candidates. Popular items, which are often extensively discussed and mentioned in the pre-training corpora of LLMs, tend to be ranked higher. Addressing this issue is challenging as it is closely tied to the composition of the pre-trained corpus.

Fairness Bias. Pre-trained language models have exhibited fairness issues related to sensitive attributes [Zhang *et al.*, 2023a], which are influenced by the training data or the demographics of the individuals involved in certain task annotations [Ferrara, 2023]. These fairness concerns can result in models making recommendations that assume users belong to a specific group, potentially leading to controversial issues when deployed commercially. One example is the bias in recommendation results caused by gender or race. Addressing these fairness issues is crucial to ensure equitable and unbiased recommendations.

5.2 Recommendation Prompt Designing

User/Item Representation. In practice, recommendation systems typically utilize a large number of discrete and continuous features to represent users and items. However, most existing LLM-based work only uses the name to represent items, and a list of item names to represent users, which is insufficient for modeling users and items accurately. Additionally, it is critical to translate a user's heterogeneous behavior sequence (such as clicks, adding to cart, and purchases in the e-commerce domain) into natural language for preference modeling. ID-like features have been proven effective in traditional recommendation models, but incorporating them into prompts to improve personalized recommendation performance is also challenging.

Limited Context Length. The context length limitation of LLMs will constrain the length of users' behavioral sequences and the number of candidate items, resulting in suboptimal performance [Zhang *et al.*, 2023b]. Existing work has proposed some techniques to alleviate this problem, such as selecting representative items from user behavior sequence [Wang and Lim, 2023] and sliding window strategy for candidate list [Sun *et al.*, 2023].

5.3 Promising Ability

Zero/Few-shot Recommendation Ability. The experimental results on multiple domain datasets indicate that LLMs possess impressive zero/few-shot abilities in various recommendation tasks [Hou *et al.*, 2023; Liu *et al.*, 2023a; Dai *et al.*, 2023]. It is worth noting that few-shot learning, which is equivalent to in-context learning, does not change the parameters of LLMs. This suggests LLMs have the potential to mitigate the cold-start problem with limited data. However, there are still some open questions, such as the need for clearer guidance in selecting representative and effective demonstration examples for few-shot learning, as well as the need for experimental results across more domains to further support the conclusion regarding the zero/few-shot recommendation abilities.

Explainable Ability. Generative LLMs exhibit a remarkable ability for natural language generation. Thus, A natural thought is using LLMs to conduct explainable recommendation via text generation manner. Liu *et al.* (2023a) conduct a comparison experiment among ChatGPT and some baselines on explanation generation task. The results demonstrate that even without fine-tuning and under the in-context learning setting, ChatGPT still performs better than some supervised traditional methods. Moreover, according to human evaluation, ChatGPT's explanations are deemed even clearer and more reasonable than the ground truth. Encouraged by these exciting preliminary experimental results, the performance of fine-tuned LLMs in explainable recommendation is expected to be promising.

5.4 Evaluation Issues

Generation Controlling. As we mentioned before, many studies have employed large-scale models as recommendation systems by providing carefully designed instructions. For these LLMs, the output should strictly adhere to the given instruction format, such as providing binary responses (yes or no) or generating a ranked list. However, in practical applications, the output of LLMs may deviate from the desired output format. For instance, the model may produce responses in incorrect formats or even refuse to provide an answer [Dai *et al.*, 2023]. And, generative models struggle to perform well in list-wise recommendation tasks due to their training data and autoregressive training mode, which make them less capable of handling ranking problems with multiple items. This issue cannot be resolved through fine-tuning, as there is no ground truth for ranking multiple items in a sequence in real-world scenarios. Therefore, it is difficult to apply autoregressive training logic based on sequence. PRP (Pairwise Ranking Prompting) [Qin *et al.*, 2023] proposes pairwise ranking for listwise tasks with LLM, which enumerates all pairs and performs a global aggregation to generate a score for each item. However, this logic is time consuming in the inference process. Therefore, addressing the challenge of ensuring better control over the output of LLMs is a pressing issue that needs to be resolved.

Evaluation Criteria. If the task performed by LLMs are standard recommendation tasks, such as rating prediction or item ranking, we can employ existing evaluation metrics for evaluation, e.g., NDCG, MSE, etc. However, LLMs also have strong generative capabilities, making them suitable for generative recommendation tasks [Wang *et al.*, 2023b]. Following the generative recommendation paradigm, LLMs can generate items that have never appeared in the historical data and recommend them to users. In this scenario, evaluating the generative recommendation capability of LLMs remains an open question.

Datasets. Currently, most of the research in this area primarily tests the recommendation capability and zero/few-shot capability of LLMs using datasets like MovieLens, Amazon Books, and similar benchmarks. However, this may bring the following two potential issues. First, compared to real-world industrial datasets, these datasets are relatively small in scale and may not fully reflect the recommendation capability of LLMs. Second, the items in these datasets, such as movies and books, may have related information that appeared in the pre-training data of LLMs. This could introduce bias in evaluating the few-zero-shot learning capability of LLMs. Currently, we still lack a suitable benchmark for conducting a more comprehensive evaluation.

In addition to the aforementioned prominent findings, there are also some limitations associated with the capabilities of large language models. For example, the challenge of knowledge forgetting may arise when training models for specific domain tasks or updating model knowledge [Jang *et al.*, 2022]. Another issue is the distinct performances caused by varying sizes of language model parameters, where using excessively large models would result in excessive computational costs for research and deployment in recommendation systems [Hou *et al.*, 2023]. These challenges also present valuable research opportunities in the field.

6 Conclusion

In this paper, we reviewed the research area of large language models (LLMs) for recommendation systems. We classified existing work into discriminative models and generative models, and then illustrated them in detail by the domain adaption manner. And in order to prevent conceptual confusion, we provided the definition and distinction of fine-tuning, prompting, prompt tuning, and instruction tuning in the LLM-based recommendation. To the best of our knowledge, our survey is the first systematic and up-to-date review specifically dedicated to generative LLMs for recommendation systems, which further summarized the common findings and challenges presented by numerous related studies. Therefore, this survey provided researchers with a valuable resource for gaining a comprehensive understanding of LLM recommendations and exploring potential research directions.

Looking to the future, as computational capabilities continue to advance and the realm of artificial intelligence expands, we anticipate even more sophisticated applications of LLMs in recommendation systems. There's a promising horizon where the adaptability and precision of these models will be harnessed in more diverse domains, possibly leading to real-time, personalized recommendations that consider multi-modal inputs. Moreover, as ethical considerations gain prominence, future LLM-based recommendation systems might also integrate fairness, accountability, and transparency more intrinsically.

In conclusion, while we have made substantial strides in understanding and implementing LLMs in recommendation systems, the journey ahead is replete with opportunities for innovation and refinement. Our survey, we hope, will serve as a foundational stepping stone for the next wave of discoveries in this dynamic and ever-evolving field.

References

[Bao *et al.*, 2023] Keqin Bao, Jizhi Zhang, Yang Zhang, Wenjie Wang, Fuli Feng, and Xiangnan He. Tallrec: An effective and efficient tuning framework to align large language model with recommendation. *CoRR*, abs/2305.00447, 2023.

[Brown *et al.*, 2020] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In *NeurIPS*, 2020.

[Chen, 2023] Zheng Chen. Palr: Personalization aware llms for recommendation. *arXiv preprint arXiv:2305.07622*, 2023.

[Cui *et al.*, 2022] Zeyu Cui, Jianxin Ma, Chang Zhou, Jingren Zhou, and Hongxia Yang. M6-rec: Generative pretrained language models are open-ended recommender systems. *CoRR*, abs/2205.08084, 2022.

[Dai *et al.*, 2022] Damai Dai, Yutao Sun, Li Dong, Yaru Hao, Zhifang Sui, and Furu Wei. Why can GPT learn in-context? language models secretly perform gradient descent as meta-optimizers. *CoRR*, abs/2212.10559, 2022.

[Dai *et al.*, 2023] Sunhao Dai, Ninglu Shao, Haiyuan Zhao, Weijie Yu, Zihua Si, Chen Xu, Zhongxiang Sun, Xiao Zhang, and Jun Xu. Uncovering chatgpt's capabilities in recommender systems. *CoRR*, abs/2305.02182, 2023.

[Da'u and Salim, 2020] Aminu Da'u and Naomie Salim. Recommendation system based on deep learning methods: a systematic review and new directions. *Artificial Intelligence Review*, 53(4):2709–2748, 2020.

[Devlin *et al.*, 2019] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT (1)*, pages 4171–4186. Association for Computational Linguistics, 2019.

[Du *et al.*, 2023] Yingpeng Du, Di Luo, Rui Yan, Hongzhi Liu, Yang Song, Hengshu Zhu, and Jie Zhang. Enhancing job recommendation through llm-based generative adversarial networks. *arXiv preprint arXiv:2307.10747*, 2023.

[Ferrara, 2023] Emilio Ferrara. Should chatgpt be biased? challenges and risks of bias in large language models. *arXiv preprint arXiv:2304.03738*, 2023.

[Friedman *et al.*, 2023] Luke Friedman, Sameer Ahuja, David Allen, Terry Tan, Hakim Sidahmed, Changbo Long, Jun Xie, Gabriel Schubiner, Ajay Patel, Harsh Lara, et al. Leveraging large language models in conversational recommender systems. *arXiv preprint arXiv:2305.07961*, 2023.

[Gao *et al.*, 2023] Yunfan Gao, Tao Sheng, Youlin Xiang, Yun Xiong, Haofen Wang, and Jiawei Zhang. Chat-rec: Towards interactive and explainable llms-augmented recommender system. *CoRR*, abs/2303.14524, 2023.

[Geng *et al.*, 2022] Shijie Geng, Shuchang Liu, Zuohui Fu, Yingqiang Ge, and Yongfeng Zhang. Recommendation as language processing (RLP): A unified pretrain, personalized prompt & predict paradigm (P5). In *RecSys*, pages 299–315. ACM, 2022.

[Hou *et al.*, 2022] Yupeng Hou, Shanlei Mu, Wayne Xin Zhao, Yaliang Li, Bolin Ding, and Ji-Rong Wen. Towards universal sequence representation learning for recommender systems. In *KDD*, pages 585–593. ACM, 2022.

[Hou *et al.*, 2023] Yupeng Hou, Junjie Zhang, Zihan Lin, Hongyu Lu, Ruobing Xie, Julian J. McAuley, and Wayne Xin Zhao. Large language models are zero-shot rankers for recommender systems. *CoRR*, abs/2305.08845, 2023.

[Jang *et al.*, 2022] Joel Jang, Seonghyeon Ye, Sohee Yang, Joongbo Shin, Janghoon Han, Gyeonghun Kim, Stanley Jungkyu Choi, and Minjoon Seo. Towards continual knowledge learning of language models. In *ICLR*, 2022.

[Ji *et al.*, 2023] Jianchao Ji, Zelong Li, Shuyuan Xu, Wenyue Hua, Yingqiang Ge, Juntao Tan, and Yongfeng Zhang. Genrec: Large language model for generative recommendation. *arXiv e-prints*, pages arXiv–2307, 2023.

[Jin *et al.*, 2023] Wei Jin, Haitao Mao, Zheng Li, Haoming Jiang, Chen Luo, Hongzhi Wen, Haoyu Han, Hanqing Lu, Zhengyang Wang, Ruirui Li, et al. Amazon-m2: A multilingual multi-locale shopping session dataset for recommendation and text generation. *arXiv preprint arXiv:2307.09688*, 2023.

[Kang *et al.*, 2023] Wang-Cheng Kang, Jianmo Ni, Nikhil Mehta, Maheswaran Sathiamoorthy, Lichan Hong, Ed H. Chi, and Derek Zhiyuan Cheng. Do llms understand user preferences? evaluating llms on user rating prediction. *CoRR*, abs/2305.06474, 2023.

[Lester *et al.*, 2021] Brian Lester, Rami Al-Rfou, and Noah Constant. The power of scale for parameter-efficient prompt tuning. *arXiv preprint arXiv:2104.08691*, 2021.

[Li *et al.*, 2023a] Jinming Li, Wentao Zhang, Tian Wang, Guanglei Xiong, Alan Lu, and Gerard Medioni. Gpt4rec:

A generative framework for personalized recommendation and user interests interpretation. *arXiv preprint arXiv:2304.03879*, 2023.

[Li *et al.*, 2023b] Lei Li, Yongfeng Zhang, and Li Chen. Personalized prompt learning for explainable recommendation. *ACM Transactions on Information Systems*, 41(4):1–26, 2023.

[Li *et al.*, 2023c] Ruyu Li, Wenhao Deng, Yu Cheng, Zheng Yuan, Jiaqi Zhang, and Fajie Yuan. Exploring the upper limits of text-based collaborative filtering using large language models: Discoveries and insights. *arXiv preprint arXiv:2305.11700*, 2023.

[Li *et al.*, 2023d] Xiangyang Li, Bo Chen, Lu Hou, and Ruiming Tang. Ctrl: Connect tabular and language model for ctr prediction. *arXiv preprint arXiv:2306.02841*, 2023.

[Li *et al.*, 2023e] Xinyi Li, Yongfeng Zhang, and Edward C Malthouse. Pbnr: Prompt-based news recommender system. *arXiv preprint arXiv:2304.07862*, 2023.

[Liu *et al.*, 2022] Qijiong Liu, Jieming Zhu, Quanyu Dai, and Xiaoming Wu. Boosting deep CTR prediction with a plug-and-play pre-trainer for news recommendation. In *COLING*, pages 2823–2833, 2022.

[Liu *et al.*, 2023a] Junling Liu, Chao Liu, Renjie Lv, Kang Zhou, and Yan Zhang. Is chatgpt a good recommender? A preliminary study. *CoRR*, abs/2304.10149, 2023.

[Liu *et al.*, 2023b] Peng Liu, Lemei Zhang, and Jon Atle Gulla. Pre-train, prompt and recommendation: A comprehensive survey of language modelling paradigm adaptations in recommender systems. *arXiv preprint arXiv:2302.03735*, 2023.

[Liu *et al.*, 2023c] Qijiong Liu, Nuo Chen, Tetsuya Sakai, and Xiao-Ming Wu. A first look at llm-powered generative news recommendation. *CoRR*, abs/2305.06566, 2023.

[Lu *et al.*, 2021] Yao Lu, Max Bartolo, Alastair Moore, Sebastian Riedel, and Pontus Stenetorp. Fantastically ordered prompts and where to find them: Overcoming few-shot prompt order sensitivity. *arXiv preprint arXiv:2104.08786*, 2021.

[Mao *et al.*, 2023] Zhiming Mao, Huimin Wang, Yiming Du, and Kam-Fai Wong. Unitrec: A unified text-to-text transformer and joint contrastive learning framework for text-based recommendation. In *Annual Meeting of the Association for Computational Linguistics*, 2023.

[Muhamed *et al.*, 2021] Aashiq Muhamed, Iman Keivanloo, Sujan Perera, James Mracek, Yi Xu, Qingjun Cui, Santosh Rajagopalan, Belinda Zeng, and Trishul Chilimbi. Ctr-bert: Cost-effective knowledge distillation for billion-parameter teacher models. In *NeurIPS Efficient Natural Language and Speech Processing Workshop*, 2021.

[Mysore *et al.*, 2023] Sheshera Mysore, Andrew McCallum, and Hamed Zamani. Large language model augmented narrative driven recommendations. *arXiv preprint arXiv:2306.02250*, 2023.

[Ouyang *et al.*, 2022] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F. Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. In *NeurIPS*, 2022.

[Penha and Hauff, 2020] Gustavo Penha and Claudia Hauff. What does BERT know about books, movies and music? probing BERT for conversational recommendation. In *RecSys*, pages 388–397. ACM, 2020.

[Petrov and Macdonald, 2023] Aleksandr V Petrov and Craig Macdonald. Generative sequential recommendation with gptrec. *arXiv preprint arXiv:2306.11114*, 2023.

[Qin *et al.*, 2023] Zhen Qin, Rolf Jagerman, Kai Hui, Honglei Zhuang, Junru Wu, Jiaming Shen, Tianqi Liu, Jialu Liu, Donald Metzler, Xuanhui Wang, et al. Large language models are effective text rankers with pairwise ranking prompting. *arXiv preprint arXiv:2306.17563*, 2023.

[Qiu *et al.*, 2021] Zhaopeng Qiu, Xian Wu, Jingyue Gao, and Wei Fan. U-BERT: pre-training user representations for improved recommendation. In *AAAI*, pages 4320–4327. AAAI Press, 2021.

[Sanner *et al.*, 2023] Scott Sanner, Krisztian Balog, Filip Radlinski, Ben Wedin, and Lucas Dixon. Large language models are competitive near cold-start recommenders for language-and item-based preferences. *arXiv preprint arXiv:2307.14225*, 2023.

[Sileo *et al.*, 2022] Damien Sileo, Wout Vossen, and Robbe Raymaekers. Zero-shot recommendation as language modeling. In *ECIR (2)*, volume 13186 of *Lecture Notes in Computer Science*, pages 223–230. Springer, 2022.

[Sun *et al.*, 2019] Fei Sun, Jun Liu, Jian Wu, Changhua Pei, Xiao Lin, Wenwu Ou, and Peng Jiang. Bert4rec: Sequential recommendation with bidirectional encoder representations from transformer. In *CIKM*, pages 1441–1450. ACM, 2019.

[Sun *et al.*, 2023] Weiwei Sun, Lingyong Yan, Xinyu Ma, Pengjie Ren, Dawei Yin, and Zhaochun Ren. Is chatgpt good at search? investigating large language models as re-ranking agent. *CoRR*, abs/2304.09542, 2023.

[Taori *et al.*, 2023] Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. Stanford alpaca: An instruction-following llama model. https://github.com/tatsu-lab/stanford_alpaca, 2023.

[Wang and Lim, 2023] Lei Wang and Ee-Peng Lim. Zero-shot next-item recommendation using large pretrained language models. *CoRR*, abs/2304.03153, 2023.

[Wang *et al.*, 2022] Xiaolei Wang, Kun Zhou, Ji-Rong Wen, and Wayne Xin Zhao. Towards unified conversational recommender systems via knowledge-enhanced prompt learning. In *KDD*, pages 1929–1937. ACM, 2022.

[Wang *et al.*, 2023a] Lei Wang, Jingsen Zhang, Xu Chen, Yankai Lin, Ruihua Song, Wayne Xin Zhao, and Ji-Rong

Wen. Recagent: A novel simulation paradigm for recommender systems. *arXiv preprint arXiv:2306.02552*, 2023.

[Wang *et al.*, 2023b] Wenjie Wang, Xinyu Lin, Fuli Feng, Xiangnan He, and Tat-Seng Chua. Generative recommendation: Towards next-generation recommender paradigm. *CoRR*, abs/2304.03516, 2023.

[Wang *et al.*, 2023c] Xiaolei Wang, Xinyu Tang, Wayne Xin Zhao, Jingyuan Wang, and Ji-Rong Wen. Rethinking the evaluation for conversational recommendation in the era of large language models. *arXiv preprint arXiv:2305.13112*, 2023.

[Wu *et al.*, 2021a] Chuhan Wu, Fangzhao Wu, Tao Qi, and Yongfeng Huang. Empowering news recommendation with pre-trained language models. In *SIGIR*, pages 1652–1656. ACM, 2021.

[Wu *et al.*, 2021b] Chuhan Wu, Fangzhao Wu, Yang Yu, Tao Qi, Yongfeng Huang, and Xing Xie. Userbert: Contrastive user model pre-training. *arXiv preprint arXiv:2109.01274*, 2021.

[Wu *et al.*, 2022] Xuyang Wu, Alessandro Magnani, Suthee Chaidaroon, Ajit Puthenputhussery, Ciya Liao, and Yi Fang. A multi-task learning framework for product ranking with BERT. In *WWW*, pages 493–501. ACM, 2022.

[Wu *et al.*, 2023] Likang Wu, Zhaopeng Qiu, Zhi Zheng, Hengshu Zhu, and Enhong Chen. Exploring large language model for graph data understanding in online job recommendations. *arXiv preprint arXiv:2307.05722*, 2023.

[Xi *et al.*, 2023] Yunjia Xi, Weiwen Liu, Jianghao Lin, Jieming Zhu, Bo Chen, Ruiming Tang, Weinan Zhang, Rui Zhang, and Yong Yu. Towards open-world recommendation with knowledge augmentation from large language models. *ArXiv*, abs/2306.10933, 2023.

[Xiao *et al.*, 2022] Shitao Xiao, Zheng Liu, Yingxia Shao, Tao Di, Bhuvan Middha, Fangzhao Wu, and Xing Xie. Training large-scale news recommenders with pretrained language models in the loop. In *KDD*, pages 4215–4225. ACM, 2022.

[Yang *et al.*, 2021] Bowen Yang, Cong Han, Yu Li, Lei Zuo, and Zhou Yu. Improving conversational recommendation systems' quality with context-aware item meta information. *arXiv preprint arXiv:2112.08140*, 2021.

[Yang *et al.*, 2022] Yingrui Yang, Yifan Qiao, Jinjin Shao, Xifeng Yan, and Tao Yang. Lightweight composite reranking for efficient keyword search with BERT. In *WSDM*, pages 1234–1244. ACM, 2022.

[Yao *et al.*, 2022] Shaowei Yao, Jiwei Tan, Xi Chen, Juhao Zhang, Xiaoyi Zeng, and Keping Yang. Reprbert: Distilling BERT to an efficient representation-based relevance model for e-commerce. In *KDD*, pages 4363–4371. ACM, 2022.

[Yu *et al.*, 2022] Yang Yu, Fangzhao Wu, Chuhan Wu, Jingwei Yi, and Qi Liu. Tiny-newsrec: Effective and efficient plm-based news recommendation. In *EMNLP*, pages 5478–5489. Association for Computational Linguistics, 2022.

[Yuan *et al.*, 2023] Zheng Yuan, Fajie Yuan, Yu Song, Youhua Li, Junchen Fu, Fei Yang, Yunzhu Pan, and Yongxin Ni. Where to go next for recommender systems? id-vs. modality-based recommender models revisited. *arXiv preprint arXiv:2303.13835*, 2023.

[Zeng *et al.*, 2021] Zheni Zeng, Chaojun Xiao, Yuan Yao, Ruobing Xie, Zhiyuan Liu, Fen Lin, Leyu Lin, and Maosong Sun. Knowledge transfer via pre-training for recommendation: A review and prospect. *Frontiers in big Data*, 4:602071, 2021.

[Zhang and Wang, 2023] Zizhuo Zhang and Bang Wang. Prompt learning for news recommendation. *arXiv preprint arXiv:2304.05263*, 2023.

[Zhang *et al.*, 2022a] Song Zhang, Nan Zheng, and Danli Wang. GBERT: pre-training user representations for ephemeral group recommendation. In *CIKM*, pages 2631–2639. ACM, 2022.

[Zhang *et al.*, 2022b] Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, et al. Opt: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068*, 2022.

[Zhang *et al.*, 2023a] Jizhi Zhang, Keqin Bao, Yang Zhang, Wenjie Wang, Fuli Feng, and Xiangnan He. Is chatgpt fair for recommendation? evaluating fairness in large language model recommendation. *CoRR*, abs/2305.07609, 2023.

[Zhang *et al.*, 2023b] Junjie Zhang, Ruobing Xie, Yupeng Hou, Wayne Xin Zhao, Leyu Lin, and Ji-Rong Wen. Recommendation as instruction following: A large language model empowered recommendation approach. *CoRR*, abs/2305.07001, 2023.

[Zhao *et al.*, 2023] Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, et al. A survey of large language models. *arXiv preprint arXiv:2303.18223*, 2023.

[Zhao, 2022] Qihang Zhao. Resetbert4rec: A pre-training model integrating time and user historical behavior for sequential recommendation. In *SIGIR*, pages 1812–1816. ACM, 2022.

[Zheng *et al.*, 2023] Zhi Zheng, Zhaopeng Qiu, Xiao Hu, Likang Wu, Hengshu Zhu, and Hui Xiong. Generative job recommendations with large language model. *arXiv preprint arXiv:2307.02157*, 2023.