Aligning Large Language Models with Recommendation Knowledge

Anonymous ACL submission

Abstract

Large language models (LLMs) have recently been used as backbones for recommender systems. However, their performance often lags behind conventional methods in standard tasks like retrieval. We attribute this to a mismatch between LLMs' knowledge and the knowledge crucial for effective recommendations. While LLMs excel at natural language reasoning, they cannot model complex user-item interactions inherent in recommendation tasks. We propose bridging the knowledge gap and equipping LLMs with recommendation-specific knowledge to address this. Operations such as Masked Item Modeling (MIM) and Bayesian Personalized Ranking (BPR) have found success in conventional recommender systems. Inspired by this, we simulate these operations through natural language to generate auxiliary-task data samples that encode item correlations and user preferences. Fine-tuning LLMs on such auxiliary-task data samples and incorporating more informative recommendation-task data samples facilitates the injection of recommendation-specific knowledge into LLMs. Extensive experiments across retrieval, ranking, and rating prediction tasks on LLMs such as FLAN-T5-Base and FLAN-T5-XL show the effectiveness of our technique in domains such as Amazon Toys & Games, Beauty, and Sports & Outdoors. Notably, our method outperforms conventional and LLM-based baselines, including the current SOTA, by significant margins in retrieval, showcasing its potential for enhancing recommendation quality.

1 Introduction

Large language models (LLMs) exhibit strong generalization abilities through zero-shot learning, in-context learning (Brown et al., 2020), fine-tuning, and instruction tuning (Wei et al., 2022). Encouraged by this, recent studies explore the use of LLMs as backbones in recommendation (Kang et al., 2023; Geng et al., 2022; Zhang et al., 2023; Bao et al., 2023). Despite their great potential, LLMs are inferior to supervised recommenders (He et al., 2017; Rendle et al., 2009) in recommendation tasks such as rating-prediction under zero-shot and few-shot in-context learning settings (Kang et al., 2023). We hypothesize that this stems from a gap between LLMs' knowledge and recommendation knowledge: LLMs are proficient at natural language reasoning, while recommendation involves modeling complex user-item interactions. In this work, we propose to mitigate this gap by fine-tuning LLMs with data samples that encode recommendation knowledge.

Recent works (Geng et al., 2022; Zhang et al., 2023; Bao et al., 2023) show that certain recommendation knowledge can be introduced into LLMs through instruction tuning. As shown in Figure 1(a), their training data samples, which we refer to as *recommendation-task data samples*, primarily help LLMs understand the recommendation tasks by providing instructions on what to do (*e.g.*, "Pick an item for the user from the following candidates."). In terms of modeling the target recommendation domain, however, they present raw user and item features for personalization (*e.g.* the user's ID or the IDs of the items they recently interacted with), which are insufficient for LLMs to fully comprehend the target domain.

Considering the aforementioned limitations of using LLMs as recommenders, we propose a novel approach to generate additional fine-tuning data samples for LLMs that effectively encode recommendation knowledge, particularly focusing on item correlations within the target domain. We refer to these generated data samples as *auxiliary-task data samples*, as they are used as auxiliary tasks in addition to the recommendations tasks. While developing the auxiliary tasks, our key inspiration comes from the classical operations that are typically used to train conventional recom-

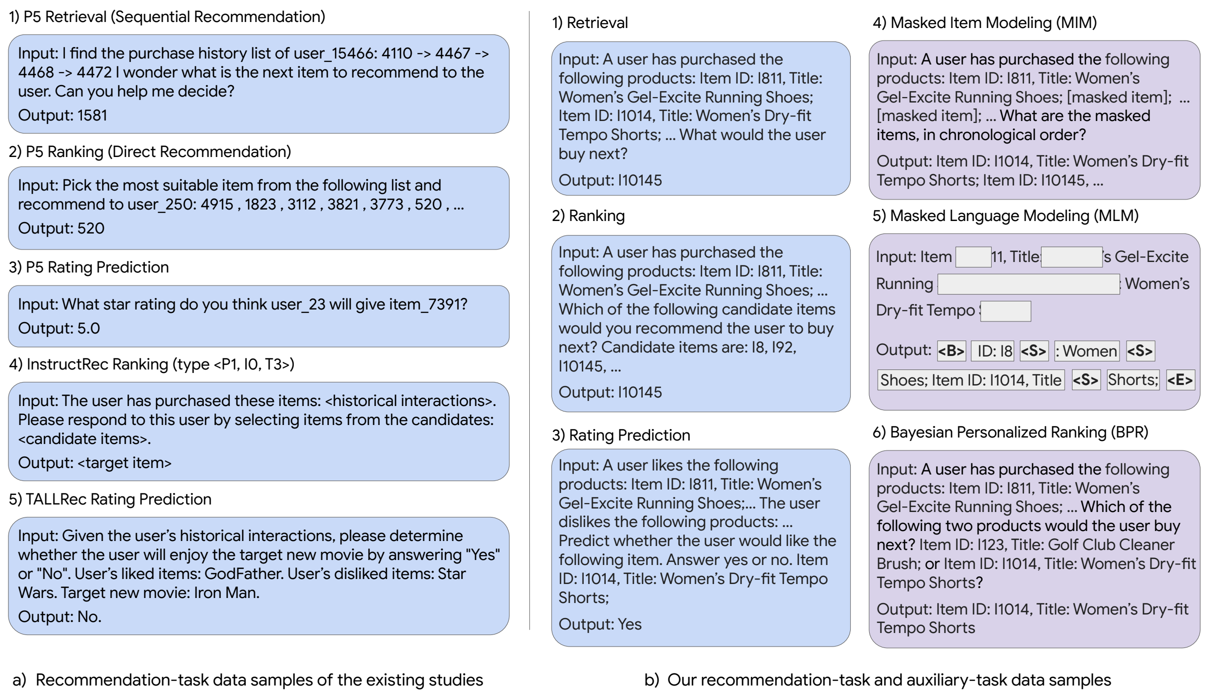

Figure 1: Data samples adopted by the existing studies and this work. (a) shows the recommendation-task data samples of the existing studies. Specifically, (a1)-(a3) demonstrate the retrieval, ranking, and rating prediction data samples of P5 (Geng et al., 2022); (a4) shows a ranking (type <P1, I0, T3>) data sample of InstructRec (Zhang et al., 2023); (a5) is a rating prediction data sample of TALLRec (Bao et al., 2023). (b) shows our recommendation-task (blue boxes) and auxiliary-task (purple boxes) data samples (we present more samples in Appendix C).

mender systems, namely, masked item modeling (MIM) (Sun et al., 2019) and Bayesian Personalized Ranking (BPR) (Rendle et al., 2009). Our key innovation lies in converting the MIM and BPR tasks into natural language tasks that can be used to train the LLMs. We also incorporate the masked language modeling (MLM) (Devlin et al., 2019) task for the user's past interactions to supplement the MIM task with fine-grained item correlations. Our contributions can be summarized as follows:

- We propose a novel method to align LLMs with new recommendation domains, *i.e.*, supplementing the fine-tuning of the LLMs with *auxiliary-task data samples* that mimic the classical operations in training conventional recommender systems with natural language prompts.
- We propose *recommendation-task data samples* that are more informative as compared to the existing work (Geng et al., 2022). Specifically, we reduce the complexity of the input/output spaces by eliminating the user IDs. We further enhance the user sequences by providing item titles.
- We fine-tune the open-source 3B FLAN-T5-XL and 223M FLAN-T5-Base with our proposed recommendation-task and auxiliary-task data samples in a simple multi-task learning framework. Experiments on various recommendation tasks, *i.e.*, retrieval, ranking, and rating-prediction, across three target domains, *i.e.*, Amazon Toys & Games, Beauty, and Sports & Outdoors, show the effectiveness of our proposed method and its components. For retrieval, our model outperforms both conventional and LLM-based baselines, including the current SOTA, by large margins.

2 Related Work

Recommender Systems. Recommender systems help users in discovering items of interest. As a practical approach, Collaborative Filtering (CF) (Mao et al., 2021) explores historical user-item interactions, assuming that users with similar behaviors have similar preferences for items. Among various CF methods, Matrix Factorization (MF) methods (Rendle et al., 2009; Mao et al., 2021) project users and items into a shared vector space and estimate a user's preference for an item through the inner product of their vectors and are widely adopted. Context-aware approaches (Cheng et al., 2016) further include additional information, such as user and contextual features, to improve recommendation quality. However, CF fails to capture the sequential patterns in users' behaviors, which leads to the rise of sequential recommendations. Sequential recommenders based on Convolutional Neural Networks (CNNs) (Tang and Wang, 2018), Gated Recurrent Units (GRUs) (Hidasi et al., 2016), and self-attention (Sun et al., 2019; Zhang et al., 2019; Kang and McAuley, 2018; Zhou et al., 2020; Rajput et al., 2023) have become prevalent in the era of deep learning. Notably, leveraging a T5-like backbone, Rajput et al. 2023 formalize recommendation as generative retrieval, *i.e.*, autoregressively decode the identifiers of the target items, and achieve the current SOTA. While structurally resembling LLMs, it lacks their pre-training knowledge and the accompanying natural language reasoning potential. Our proposed approach adopts self-attention for sequential recommendation, specifically harnessing LLMs as backbones. We compare against various baselines from all the classes discussed above.

LLMs for Recommendation. LLMs have recently been explored for recommendation tasks due to their ability to understand, generate, and reason with natural language. Several studies focus on incorporating LLMs' natural language capabilities into existing recommendation techniques. *E.g.*, Hou et al. 2022 and Cao et al. 2023 encode item contents (title, description, etc.) with BERT (Devlin et al., 2019), which enables learning semantically informed embeddings even for zero-shot items. Moreover, pre-trained LLM backbones have also been used for recommendation through zero-shot learning (Kang et al., 2023), in-context learning (Kang et al., 2023), fine tuning (Cui et al., 2022; Kang et al., 2023), and instruction tuning (Geng et al., 2022; Zhang et al., 2023; Bao et al., 2023). Besides helping classic recommendation tasks, LLMs also enable novel recommendation use cases. Geng et al. 2022 leverage LLMs to explain the recommendation results. Gao et al. 2023; Wang and Lim 2023 utilize GPT-3 (Brown et al., 2020) for conversational recommendation. Christakopoulou et al. 2023 extract persistent user interests with LLMs for deeper user understanding. Carranza et al. 2023 generate private synthetic representations of the original data with LLMs for privacy-preserving recommendation.

Recommendation as Instruction-following. The success of instruction tuning, *i.e.*, fine-tune on data described via instructions (Mishra et al., 2022; Wei et al., 2022), has inspired attempts that instruction-tune LLM backbones for recommendation tasks. Geng et al. 2022 formalize various recommendation tasks as natural language instructions and fine-tune a unified recommender with T5 (Raffel et al., 2020) backbone. Zhang et al. 2023 further supplement the tuning data with user preferences/intentions deduced by GPT-3.5[1] to accommodate instructions of free forms. Bao et al. 2023 explore instruction tuning LLMs with limited data.

In contrast to the existing studies, our work focuses on introducing new recommendation knowledge into LLMs, which we believe is the key for improving recommenders with LLM backbones. We create auxiliary tasks that improve the recommendation tasks, including retrieval, ranking, and rating prediction. Our proposed recommendation-task and auxiliary-task data samples include raw user purchase sequences in addition to natural language instructions. These data samples supplement each other in encoding the target recommendation domain knowledge. We experiment under restricted settings. Compared to the previous studies (Zhang et al., 2023), we consider larger candidate pools (*e.g.*, our retrieval and ranking experiments consider the entire dataset and 99 hard negatives, respectively). Unlike Bao et al. 2023, we fully train all models to maximize their performances.

3 Methodology

We propose designing data samples that encode recommendation knowledge to align LLMs with the target recommendation domain. Sections 3.1 and 3.2 discuss our auxiliary-task and recommendation-task data, respectively. Section 3.3 introduces a simple multi-task learning framework that we use to fine-tune LLMs.

3.1 Auxiliary-task Data Generation

Conventional recommenders acquire recommendation knowledge via classic operations such as masked item modeling (Sun et al., 2019) and BPR loss reduction (Rendle et al., 2009). We mimic these operations with natural language prompts. In addition, we sample sub-sequences of the raw user purchase sequences. The resulting data, which we refer to as auxiliary-task data samples, encode item correlations contained in users' preferences [2].

[1] https://platform.openai.com/docs/models/overview

[2] As a side note, we also explored encoding item correlations contained in item contents (categories, descriptions, etc.). Observing no noticeable performance increase, we present our approach and results in Appendix D

3.1.1 Masked Item Modeling (MIM)

Conventional sequential recommenders (Sun et al., 2019) learn item correlations from users' interaction sequences. Specifically, they predict randomly masked items in the sequences by jointly conditioning on the unmasked items. We mimic this process, which we refer to as masked item modeling (MIM), with natural language prompts.

MIM applies a Cloze objective (Sun et al., 2019). At each training step, random items in the input user sequence are replaced with a special token "[mask]", and the model learns to recover the masked items based on its surrounding context. An example of the masking process:

$$\begin{aligned}\textbf{Input: } & [i_1, i_2, i_3, i_4, i_5] \xrightarrow{\text{random masking}} \\ & [i_1, [\text{mask}]_1, i_3, [\text{mask}]_2, i_5] \\ \textbf{Label: } & [\text{mask}]_1 = i_2,\ [\text{mask}]_2 = i_4\end{aligned} \tag{1}$$

The MIM loss is computed as follows in conventional sequential recommenders:

$$\mathcal{L}_{\text{MIM}} = \frac{1}{|\mathcal{S}_u^m|} \sum_{i_m \in \mathcal{S}_u^m} -\log P(i_m | \mathcal{S}_u^{'}), \tag{2}$$

where $\mathcal{S}_u^{'}$ is the masked version of user sequence $\mathcal{S}_u$, $\mathcal{S}_u^m$ stands for the masked items in $\mathcal{S}_u$. $P(\cdot)$, the probability of observing i_m given $\mathcal{S}_u^{'}$, is calculated from deep bidirectional self-attention (Devlin et al., 2019).

Our natural language imitation of MIM loss (Equation 2) is described in Figure 1(b4). Given purchase sequence: $[i_1, i_2, i_3, i_4, i_5]$, we generate prompts, *e.g.*, Input: "A user has purchased the following products: Item ID: $[\text{ID}]_{i_1}$, Title: $[\text{Title}]_{i_1}$; [masked item]; Item ID: $[\text{ID}]_{i_3}$, Title: $[\text{Title}]_{i_3}$; [masked item]; Item ID: $[\text{ID}]_{i_5}$, Title: $[\text{Title}]_{i_5}$. What are the masked items, in chronological order?", and Output: "Item ID: $[\text{ID}]_{i_2}$, Title: $[\text{Title}]_{i_2}$; Item ID: $[\text{ID}]_{i_4}$, Title: $[\text{Title}]_{i_4}$;". To accommodate long sequences, we introduce a sliding window w and each prompt considers one sub-sequence: $[i_k, i_{k+1} ..., i_{k+w-1}]$, where $1 \leq k \leq \max\big(1, (L\text{-}w\text{+}1)\big)$ and L is the total length of the user sequence. The resulting MIM data samples encodes the correlations between the masked items and the rest of the sequences.

3.1.2 Masked Language Modeling (MLM)

In addition to MIM that considers a single item for each mask, we also mask out and recover a consecutive span of *tokens* to encode fine-grained item correlations contained in the users' purchase sequences. This process resembles masked language modeling (MLM) (Devlin et al., 2019).

As shown in Figure 1(b5), given a user sequence, we sample a sub-sequence by randomly deciding a starting item and a sub-sequence length L_s, where $2 \leq L_s \leq w$ and w is the sliding window for accommodating long sequences. These sub-sequences, referred to as MLM data samples, supplement the MIM data samples: through span corruption (Raffel et al., 2020), *i.e.*, masking and recovering consecutive spans of tokens, LLMs learn to model more fine-grained correlations across multiple continuous items from the MLM data samples.

3.1.3 Bayesian Personalized Ranking (BPR)

Besides correlating similar items, we explore contrasting dissimilar items. BPR loss (Rendle et al., 2009) is adopted by conventional recommenders (Rendle and Freudenthaler, 2014; Koren et al., 2009; Cheng et al., 2016) for personalized ranking, *i.e.*, learning users' preferences for some items over the others. Inspired by this, we imitate BPR loss reduction with natural language prompts for training LLMs.

The objective of BPR loss reduction in conventional recommenders is:

$$\mathcal{L}_{\text{BPR}} = \mathop{\mathbb{E}}_{(u,i^+)\sim p_{\text{pos}}} -\log \sigma(s(u,i^+) - s(u,i^-)), \tag{3}$$

where (u, i^+) is a pair of a user u and an item i^+ sampled from the distribution of positive pairs p_{pos}, *i.e.*, u interacted with i^+. i^- is a randomly sampled negative item that u has not interacted with. The similarity between u and i^+, denoted by $s(u, i^+)$, is calculated by taking the dot product of their representations. $\sigma(\cdot)$ is the Sigmoid function.

Figure 1(b6) shows our natural language imitation. We elicit user preferences by generating prompts with binary choices that contrast a positive item and a negative item. Each prompt takes the form of a binary decision, *e.g.*, Input: "A user has purchased ... Which of the following two products would the user buy next? Item ID: $[\text{ID}]_{i^-}$, Title: $[\text{Title}]_{i^-}$; Item ID: $[\text{ID}]_{i^+}$, Title: $[\text{Title}]_{i^+}$.", and Output: "Item ID: $[\text{ID}]_{i^+}$, Title: $[\text{Title}]_{i^+}$". Following Section 3.1.1, we adopt a sliding window w to accommodate long user sequences and the positive item is always the one next to the sliding window. These BPR data samples encode dissimilarities between the purchased items and the rest of the items in the dataset.

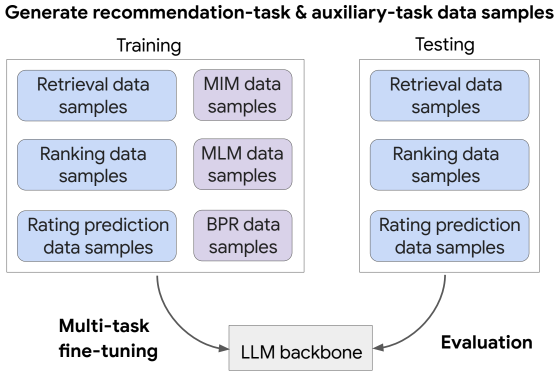

Figure 2: Fine-tuning and evaluation framework.

3.2 Recommendation-task Data Generation

As shown in Figure 1(a), the existing recommenders with LLM backbones adopt prompts that primarily convey the recommendation tasks by providing directions on how to perform them. Such information is essential, yet insufficient for representing the target recommendation domain.

We propose prompts that help LLMs comprehend the target recommendation domain in addition to the recommendation tasks. Specifically, we reduce the complexity of the input/output spaces. In contrast to Geng et al. 2022, we eliminate user IDs and represent the users by their historical purchases. Consequently, we relieve LLMs from memorizing a substantial volume of user IDs (*e.g.*, Amazon Sports & Outdoors has 35,598 users). Moreover, compared to Geng et al. 2022 that represent user sequences solely by item IDs, we include both the IDs and the titles of the items, which makes it easier for LLMs to recognize the items. Notably, ranking candidates and items in the output are represented solely by their IDs to reduce the length of the prompts and maintain a smaller output space. Figures 1(b1)-(b3) show examples of our retrieval, ranking, and rating prediction recommendation-task data samples. The raw item IDs (*e.g.*, '0000031852') are mapped into shorter ones (*e.g.*, 'I123') [3] to reduce input/output space complexity. To fully present the users' historical purchases to LLMs, we adopt a sliding window w similar to Section 3.1.1.

3.3 Fine-tuning and Evaluation Framework

As shown in Figure 2, we adopt a simple framework to fine-tune the LLM backbones and evaluate the resulting model. We first generate recommendation-task and auxiliary-task data samples using the training set. Next, we tune the LLM backbone with these data samples in a multi-task learning manner. Finally, we evaluate the recommendation tasks using the recommendation-task data samples generated from the test set.

4 Experiments

We evaluate the proposed method and compare it with conventional as well as LLM-based recommenders. We aim to answer the following research questions: **RQ1.** Can our method introduce knowledge into LLMs from new recommendation domains? **RQ2.** How does our model perform compared to the conventional as well as LLM-based recommenders in retrieval, ranking, and rating prediction? **RQ3.** How beneficial are the individual proposed tasks? **RQ4.** What's the effect of varying the size of the backbone LLM?

4.1 Experimental Setting

Datasets. We experiment on three real-world datasets: Amazon Toys & Games, Beauty, and Sports & Outdoors [4]. Following Zhou et al. 2020; Geng et al. 2022, we keep 5-core data and apply leave-one-out evaluation, *i.e.*, for each user purchase sequence (where the interactions are sorted by timestamp in ascending order), the last, the second to the last, and the prior interactions are used for testing, validation, and training, respectively. We present data statistics in Appendix B.

Recommendation Tasks. We evaluate on three established recommendation tasks: **retrieval**, which retrieves the ground truth item that a user interacted with from the entire dataset; **ranking**, which chooses the ground truth item that a user interacted with from a candidate pool of size 100 (1 positive item and 99 negative items sampled based on popularity); **rating prediction**, which classifies an interaction as either "like" or "dislike" (interactions with ratings > 3 are considered as "like"). We leave the exploration and evaluation of novel recommendation tasks (*e.g.*, explanation generation) to the future, due to a lack of ground-truth data.

Evaluation Metrics. For retrieval and ranking, we report top-k Hit Ratio (HR@k) and Normalized Discounted Cumulative Gain (NDCG@k), where k is set to 5/10 and 1/5/10, respectively. For rating prediction, we report Area Under the Receiver

[3] We adopt random mapping, *i.e.*, similar-looking IDs may not imply any connection or semantic similarity. We acknowledge that using semantic-rich IDs (Rajput et al., 2023) could enhance performance and leave the exploration to the future.

[4] https://nijianmo.github.io/amazon/

Methods	Toys & Games				Beauty				Sports & Outdoors			
	NDCG @5	NDCG @10	HR @5	HR @10	NDCG @5	NDCG @10	HR @5	HR @10	NDCG @5	NDCG @10	HR @5	HR @10
Caser[1]	0.0107	0.0141	0.0166	0.0270	0.0131	0.0176	0.0205	0.0347	0.0072	0.0097	0.0116	0.0194
HGN[1]	0.0221	0.0277	0.0321	0.0497	0.0206	0.0266	0.0325	0.0512	0.0120	0.0159	0.0189	0.0313
GRU4Rec[1]	0.0059	0.0084	0.0097	0.0176	0.0099	0.0137	0.0164	0.0283	0.0086	0.0110	0.0129	0.0204
BERT4Rec[1]	0.0071	0.0099	0.0116	0.0203	0.0124	0.0170	0.0203	0.0347	0.0075	0.0099	0.0115	0.0191
FDSA[1]	0.0140	0.0189	0.0228	0.0381	0.0163	0.0208	0.0267	0.0407	0.0122	0.0156	0.0182	0.0288
SASRec[1]	0.0306	0.0374	0.0463	0.0675	0.0249	0.0318	0.0387	0.0605	0.0154	0.0192	0.0233	0.0350
S^3-Rec[1]	0.0294	0.0376	0.0443	0.0700	0.0244	0.0327	0.0387	0.0647	0.0161	0.0204	0.0251	0.0385
TIGER[2]	0.0371	0.0432	0.0521	0.0712	0.0321	0.0384	0.0454	0.0648	0.0181	0.0225	0.0264	0.0400
P5[2]	0.0050	0.0066	0.0070	0.0121	0.0107	0.0136	0.0163	0.0254	0.0041	0.0052	0.0061	0.0095
P5-XL	0.0023	0.0031	0.0035	0.0061	0.0036	0.0050	0.0063	0.0104	0.0029	0.0035	0.0040	0.0060
FLAN-T5-Base	0.0000	2e−5	0.0000	5e−5	0.0000	0.0000	0.0000	0.0000	0.0000	9e−6	0.0000	3e−5
FLAN-T5-XL	2e−5	2e−5	5e−5	5e−5	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
ReAT [**Ours**]	0.0390	**0.0461**	0.0558	**0.0776**	**0.0382**	**0.0442**	**0.0535**	**0.0722**	**0.0188**	**0.0232**	**0.0285**	**0.0422**
UT [**Ours**]	0.0166	0.0202	0.0252	0.0362	0.0188	0.0231	0.0292	0.0425	0.0079	0.0101	0.0118	0.0187
UT+AT [**Ours**]	**0.0392**	0.0459	**0.0563**	0.0772	0.0329	0.0397	0.0482	0.0693	0.0178	0.0219	0.0268	0.0393
Δ (%)	+5.66	+6.71	+8.06	+8.99	+19.00	+15.10	+17.84	+11.42	+3.87	+3.11	+7.95	+5.50

Table 1: Retrieval results. [1] marks results from Zhou et al. 2020; [2] marks results from Rajput et al. 2023. Δ compares the best [**Ours**] with the best baseline.

Methods	Toys & Games					Beauty					Sports & Outdoors				
	NDCG @5	NDCG @10	HR @1	HR @5	HR @10	NDCG @5	NDCG @10	HR @1	HR @5	HR @10	NDCG @5	NDCG @10	HR @1	HR @5	HR @10
BPR-MF[1]	0.0641	0.0940	0.0233	0.1066	0.2003	0.0857	0.1224	0.0311	0.1426	0.2573	0.0848	0.1220	0.0314	0.1404	0.2563
BPR-MLP[1]	0.0688	0.0988	0.0252	0.1142	0.2077	0.0848	0.1215	0.0317	0.1392	0.2542	0.0927	0.1296	0.0351	0.1520	0.2671
SimpleX[1]	0.1244	0.1469	0.0268	0.1958	0.2662	**0.1441**	0.1711	0.0325	**0.2247**	0.3090	**0.1505**	**0.1800**	0.0331	**0.2362**	**0.3290**
P5-XL	0.0290	0.0444	0.0097	0.0494	0.0977	0.0298	0.0456	0.0110	0.0498	0.0992	0.0286	0.0436	0.0097	0.0486	0.0957
FLAN-T5-Base	0.0107	0.0127	0.0057	0.0156	0.0217	0.0097	0.0113	0.0052	0.0137	0.0189	0.0069	0.0082	0.0035	0.0102	0.0144
FLAN-T5-XL	0.0160	0.0312	0.0026	0.0315	0.0793	0.0152	0.0296	0.0022	0.0301	0.0753	0.0097	0.0193	0.0014	0.0192	0.0491
RaAT [**Ours**]	**0.1714**	0.2034	**0.0956**	**0.2464**	0.3453	0.1376	0.1691	0.0702	0.2036	0.3013	0.0933	0.1199	0.0424	0.1448	0.2272
UT [**Ours**]	0.1536	0.1867	0.0831	0.2233	0.3259	0.1236	0.1537	0.0609	0.1863	0.2798	0.0867	0.1137	0.0381	0.1362	0.2202
UT+AT [**Ours**]	0.1703	**0.2064**	0.0938	0.2443	**0.3562**	**0.1441**	**0.1758**	**0.0742**	0.2126	**0.3112**	0.0997	0.1281	**0.0468**	0.1526	0.2404
Δ (%)	+37.78	+40.50	+256.72	+25.84	+33.81	0.00	+2.75	+128.31	-5.38	+0.71	-33.75	-28.83	+33.33	-35.39	-26.93

Table 2: Ranking results. [1] marks results from Geng et al. 2022. Δ compares the best [**Ours**] with the best baseline.

Methods	Toys & Games	Beauty	Sports & Outdoors
History	66.59	64.80	62.78
DMF	51.82	51.23	51.38
Wide&Deep	70.93	67.10	**67.60**
P5-XL	51.04	50.63	50.36
FLAN-T5-Base	57.85	56.04	55.00
FLAN-T5-XL	55.23	53.77	52.01
RpAT [**Ours**]	**71.16**	**68.27**	65.87
UT [**Ours**]	70.79	67.45	65.35
UT+AT [**Ours**]	71.08	67.55	65.18
Δ (%)	+0.32	+1.74	-2.56

Table 3: Rating prediction AUC-ROC. Δ compares the best [**Ours**] with the best baseline.

Operating Characteristic Curve (AUC-ROC).

Models. We compare to non LLM-based recommenders. For retrieval, we consider sequential recommenders including **Caser** (Tang and Wang, 2018), which leverages CNNs, **HGN** (Ma et al., 2019), which adopts hierarchical gating networks, **GRU4Rec** (Hidasi et al., 2016), which leverages GRUs (Cho et al., 2014), **BERT4Rec** (Sun et al., 2019), **FDSA** (Zhang et al., 2019), **SASRec** (Kang and McAuley, 2018), **S^3-Rec** (Zhou et al., 2020), and **TIGER** (Rajput et al., 2023), which leverage self-attention, with TIGER being the current SOTA. For ranking, we consider **BPR-MF** (Rendle et al., 2009), **BPR-MLP** (Cheng et al., 2016), and **SimpleX** (Mao et al., 2021), which are collaborative filtering-based method. For rating prediction, we consider **History**, a naive method that always predicts based on how likely a user likes the training items they purchased, **DMF** (Xue et al., 2017), a neural matrix factorization model, and **Wide&Deep** (Cheng et al., 2016), a context-aware method. Beside, we also consider LLM-based methods including **P5** (Geng et al., 2022), which fine-tunes T5 (Raffel et al., 2020) with multi-task recommendation prompts, **P5-XL**, which fine-tunes FLAN-T5-XL with P5 prompts, **FLAN-T5-Base/XL** (Wei et al., 2022), which make zero-shot predictions with FLAN-T5-Base or FLAN-T5-XL (we query them with our proposed recommendation-task data samples generated from the test set). **ReAT/ RaAT/ RpAT**, which fine-tune FLAN-T5-XL with our proposed retrieval (Re), ranking (Ra), or rating prediction (Rp) task data samples along with the auxiliary-task (AT) data samples [5], **unified training (UT)**, which fine-tunes FLAN-T5-XL with a combination of our proposed Re, Ra, Rp data samples, **unified training w/ auxiliary tasks (UT+AT)**, which fine-tunes FLAN-T5-XL with a combination of our proposed Re, Ra, Rp, MIM, MLM data samples.

Implementation Details. We adopt the 3B FLAN-T5-XL (Wei et al., 2022) as the backbone (note that we use the 223M FLAN-T5-Base for the ablation studies in Section 4.3). We set the sliding window size w to 20. For the BPR data samples, we sample the negative items based on popularity. For the ranking and BPR data samples, the position of the positive item in the candidate pool is always determined randomly. For the MIM and MLM data samples, we adopt a masking ratio of 20%. To fully fine-tune the LLM backbone, we apply dynamic sampling for the BPR and MIM/MLM data samples (we present details about the dynamic sampling and the statistics of our data samples in Appendix C). To reduce cost, we validate on 3,000 users. Meanwhile, testing is performed on all users. We fine-tune FLAN-T5-XL and FLAN-T5-Base for $70,000$ and $10,000$ steps, with batch sizes 16 and 64, respectively. We set the learning rate to 0.001 and warm-up steps to 1,000. During prediction, we set the width of the beam search for retrieval and ranking to 20. For unified models, *i.e.*, UT and UT+AT, model selections are based on retrieval validation performance. We present the detailed settings of P5-XL experiments in Appendix A. We cite the results of some baseline models from Zhou et al. 2020; Geng et al. 2022, and Rajput et al. 2023. We implement DMF and Wide&Deep with RecBole [6]. We adopt the default configurations, except the data split, mapping (ratings to "like"s or "dislike"s), and metric are adjusted to follow our experiment settings as reported earlier.

4.2 Overall Performance (RQ1 & RQ2)

Tables 1, 2, and 3 show the results of retrieval, ranking, and rating prediction, respectively. FLAN-T5-Base/XL exhibit suboptimal performance on retrieval and ranking. For retrieval, they show near zero NDCGs and HRs. For ranking, they are significantly inferior to the conventional baselines. For rating prediction, they perform much higher than random guessing (50.00), outperforming DMF, but still fall behind History and Wide&Deep. This shows that FLAN-T5 models lack recommendation knowledge. Moreover, we find that our proposed method effectively aligns LLMs with new recommendation domains (RQ1). In particular, by fine-tuning FLAN-T5-XL with our proposed data samples, our models significantly outperform FLAN-T5-XL on all three tasks across the datasets.

When compared to the baselines, our models show remarkable performance, especially on retrieval (RQ2). For retrieval, our ReAT outperforms TIGER, the current SOTA, by large margins across datasets and metrics. Additionally, it is essential to highlight that our method possesses natural language reasoning potentials of LLMs, which are absent in TIGER. For ranking, our RaAT greatly outperforms SimpleX, the best baseline, on Toys & Games. On Beauty, RaAT performs on par with SimpleX. On Sports & Outdoors, RaAT is inferior to the conventional recommenders on metrics such as NDCG/HR@10, yet still greatly outperforms the LLM-based baselines. Notably, the @1 performance of RaAT is always much higher than the conventional recommenders. For rating prediction, our RpAT outperforms Wide&Deep, the best baseline, on Toys & Games and Beauty while lags slightly behind it on Sports & Outdoors. These results verify that our method introduces substantial recommendation domain knowledge into LLMs for outperforming strong baselines.

Moreover, our UT greatly outperforms P5-XL across datasets and metrics. This shows that our proposed recommendation task prompts better preserve item correlations as compared to the P5 ones. Specifically, we enhance user sequence modeling by introducing helpful details such as item titles while excluding less informative details such as user IDs and explanation data.

We also compare our UT+AT model with our task-specific models, *i.e.*, ReAT/ RaAT/ RpAT. We show that our method allows fine-tuning a unified model that addresses all recommendation tasks without sacrificing per-task performance by much. For retrieval, UT+AT is slightly worse than ReAT but still outperforms all baselines, except that UT+AT performs comparably with TIGER on Sports & Outdoors. For ranking, UT+AT performs

[5] BPR data samples are used only by RaAT as we observe that they help ranking but not retrieval and rating prediction. MIM/ MLM data samples are used by ReAT, RaAT, and RpAT.

[6] https://recbole.io

#	Methods	NDCG @5	NDCG @10	HR @5	HR @10
1	TIGER	0.0371	0.0432	0.0521	0.0712
2	FLAN-T5-XL	2e−5	2e−5	5e−5	5e−5
3	2+retrieval	0.0182	0.0219	0.0273	0.0388
4	3+MLM	0.0306	0.0369	0.0443	0.0641
5	4+MIM	**0.0390**	**0.0461**	**0.0558**	**0.0776**
6	FLAN-T5-Base	0.0000	2e−5	0.0000	5e−5
7	6+retrieval	0.0149	0.0183	0.0219	0.0325
8	7+MLM	0.0219	0.0271	0.0334	0.0495
9	8+MIM	0.0242	0.0304	0.0376	0.0566

Table 4: Retrieval ablation study on Toys & Games. Rows 1, 2, 5 (equivalent to ReAT), and 6 are copied from Table 1.

#	Methods	NDCG @5	NDCG @10	HR @1	HR @5	HR @10
1	SimpleX	0.1244	0.1469	0.0268	0.1958	0.2662
2	FLAN-T5-XL	0.0160	0.0312	0.0026	0.0315	0.0793
3	2+ranking	0.1520	0.1864	0.0807	0.2218	0.3284
4	3+MLM	0.1580	0.1912	0.0854	0.2303	0.3333
5	4+MIM	0.1677	0.1976	0.0938	0.2391	0.3317
6	5+BPR	**0.1714**	**0.2034**	**0.0956**	**0.2464**	**0.3453**
7	FLAN-T5-Base	0.0107	0.0127	0.0057	0.0156	0.0217
8	7+ranking	0.1349	0.1654	0.0720	0.1957	0.2901
9	8+MLM	0.1481	0.1782	0.0820	0.2119	0.3051
10	9+MIM	0.1489	0.1811	0.0817	0.2141	0.3136
11	10+BPR	0.1534	0.1844	0.0844	0.2196	0.3153

Table 5: Ranking ablation study on Toys & Games. Rows 1, 2, 6 (equivalent to RaAT), and 7 are copied from Table 2.

on par with or slightly better than our task-specific RaAT model. For rating prediction, UT+AT is slightly worse than RpAT.

4.3 Ablation Studies (RQ3 & RQ4)

Tables 4, 5, and 6 show ablation studies on Toys & Games for retrieval, ranking, and rating prediction, respectively. We observe that all the proposed tasks are beneficial (RQ3). In Table 4 rows 2-5, successively adding our proposed retrieval, MLM, and MIM data samples into the fine-tuning data increases the retrieval performance. All three tasks are essential. *E.g.*, row 4, which fine-tunes FLAN-T5-XL using retrieval and MLM data samples performs on par with S^3-Rec and worse than TIGER (row 1, the current SOTA). Further adding MIM data samples (row 5) surpasses TIGER. This shows that the item-level and token-level item correlations introduced by MIM and MLM are essential and complement each other. Similarly, in Table 5 rows 2-6, the ranking performance improves as we incorporate our proposed ranking, MLM, MIM, and BPR data samples into fine tuning. Among these data samples, ranking task data samples are the most helpful. BPR data samples, which contrast the positive items with the negative ones, provide the least assistance. For rating predictions, as shown in Table 6 rows 2-5, our proposed rating prediction data samples greatly increase the performance. MLM and MIM do help, but only marginally.

We also find that our proposed method is effective regardless of the size of the backbone model (RQ4). In Tables 4, 5, and 6, we apply our method on FLAN-T5-Base and observe significant performance increases on all three recommendation tasks. In terms of overall performance, our best retrieval model with FLAN-T5-Base (Table 4 row 9) falls behind TIGER but still outperforms all baselines except TIGER, S^3-Rec, and SASRec. In Table 5, our best ranking model with FLAN-T5-Base (row 11) outperforms SimpleX by large margins, though falls behind our best ranking model with FLAN-T5-XL (row 6). In Table 6, our best rating prediction model with FLAN-T5-Base (row 7) is slightly inferior to the best model with FLAN-T5-XL (row 5) and Wide&Deep. The effectiveness of the individual tasks remains roughly consistent with the previous results with FLAN-T5-XL (except that MLM does not help rating prediction). *E.g.*, in Table 5 rows 7-11, our ranking task, MLM, MIM, and BPR data samples all contribute to the ranking performance, with the ranking task data samples being the most beneficial and BPR the least beneficial.

#	Methods	AUC-ROC
1	Wide&Deep	70.93
2	FLAN-T5-XL	55.23
3	2+rating-prediction	70.38
4	3+MLM	71.08
5	4+MIM	**71.16**
6	FLAN-T5-Base	57.85
7	6+rating-prediction	69.17
8	7+MLM	67.31
9	8+MIM	68.24

Table 6: Rating-prediction ablation study on Toys & Games. Rows 1, 2, 5 (equivalent to RpAT), and 6 are copied from Table 3.

5 Conclusion

We propose to align LLMs with the recommendation domain by fine-tuning with data samples that encode recommendation knowledge. We propose auxiliary-task data samples that encode item correlations contained in users' preferences. We further design recommendation-task data samples that are more informative than ones in existing studies. Experiments on retrieval, ranking, and rating prediction show that our method effectively introduces recommendation knowledge into FLAN-T5-Base/XL from three domains. Our method greatly outperforms both conventional and LLM-based baselines in retrieval, achieving the new SOTA.

6 Limitations

Our proposed method utilizes LLMs as the backbones. The substantial parameter size of the LLMs results in increased computational resource consumption and extended training and inference times compared to conventional recommenders. Nevertheless, adopting LLM backbones is beneficial due to their significant potential. In addition to the exceptional performance demonstrated in this study, we anticipate that future research will continue to augment existing recommendation tasks and address novel recommendation scenarios by leveraging the diverse capabilities of LLM backbones.

References

Keqin Bao, Jizhi Zhang, Yang Zhang, Wenjie Wang, Fuli Feng, and Xiangnan He. 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*.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Yuwei Cao, Liangwei Yang, Chen Wang, Zhiwei Liu, Hao Peng, Chenyu You, and Philip S Yu. 2023. Multi-task item-attribute graph pre-training for strict cold-start item recommendation. In *Proceedings of the 17th ACM Conference on Recommender Systems*.

Aldo Gael Carranza, Rezsa Farahani, Natalia Ponomareva, Alex Kurakin, Matthew Jagielski, and Milad Nasr. 2023. Privacy-preserving recommender systems with synthetic query generation using differentially private large language models. *arXiv preprint arXiv:2305.05973*.

Heng-Tze Cheng, Levent Koc, Jeremiah Harmsen, Tal Shaked, Tushar Chandra, Hrishi Aradhye, Glen Anderson, Greg Corrado, Wei Chai, Mustafa Ispir, et al. 2016. Wide & deep learning for recommender systems. In *Proceedings of the 1st workshop on deep learning for recommender systems*, pages 7–10.

Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using rnn encoder-decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*.

Konstantina Christakopoulou, Alberto Lalama, Cj Adams, Iris Qu, Yifat Amir, Samer Chucri, Pierce Vollucci, Fabio Soldo, Dina Bseiso, Sarah Scodel, et al. 2023. Large language models for user interest journeys. *arXiv preprint arXiv:2305.15498*.

Zeyu Cui, Jianxin Ma, Chang Zhou, Jingren Zhou, and Hongxia Yang. 2022. M6-rec: Generative pretrained language models are open-ended recommender systems. *arXiv preprint arXiv:2205.08084*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT 2019*, pages 4171–4186.

Yunfan Gao, Tao Sheng, Youlin Xiang, Yun Xiong, Haofen Wang, and Jiawei Zhang. 2023. Chat-rec: Towards interactive and explainable llms-augmented recommender system. *arXiv preprint arXiv:2303.14524*.

Shijie Geng, Shuchang Liu, Zuohui Fu, Yingqiang Ge, and Yongfeng Zhang. 2022. Recommendation as language processing (rlp): A unified pretrain, personalized prompt & predict paradigm (p5). In *Proceedings of the 16th ACM Conference on Recommender Systems*, pages 299–315.

Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. 2017. Neural collaborative filtering. In *Proceedings of the 26th international conference on world wide web*, pages 173–182.

Balázs Hidasi, Alexandros Karatzoglou, Linas Baltrunas, and Domonkos Tikk. 2016. Session-based recommendations with recurrent neural networks. In *Proceedings of the 4th International Conference on Learning Representations*.

Yupeng Hou, Shanlei Mu, Wayne Xin Zhao, Yaliang Li, Bolin Ding, and Ji-Rong Wen. 2022. Towards universal sequence representation learning for recommender systems. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 585–593.

Wang-Cheng Kang and Julian McAuley. 2018. Self-attentive sequential recommendation. In *2018 IEEE international conference on data mining (ICDM)*, pages 197–206. IEEE.

Wang-Cheng Kang, Jianmo Ni, Nikhil Mehta, Maheswaran Sathiamoorthy, Lichan Hong, Ed Chi, and Derek Zhiyuan Cheng. 2023. Do llms understand user preferences? evaluating llms on user rating prediction. *arXiv preprint arXiv:2305.06474*.

Yehuda Koren, Robert Bell, and Chris Volinsky. 2009. Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37.

Chen Ma, Peng Kang, and Xue Liu. 2019. Hierarchical gating networks for sequential recommendation. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 825–833.

Kelong Mao, Jieming Zhu, Jinpeng Wang, Quanyu Dai, Zhenhua Dong, Xi Xiao, and Xiuqiang He. 2021. Simplex: A simple and strong baseline for collaborative filtering. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, pages 1243–1252.

Swaroop Mishra, Daniel Khashabi, Chitta Baral, and Hannaneh Hajishirzi. 2022. Cross-task generalization via natural language crowdsourcing instructions. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 21(1):5485–5551.

Shashank Rajput, Nikhil Mehta, Anima Singh, Raghunandan H Keshavan, Trung Vu, Lukasz Heldt, Lichan Hong, Yi Tay, Vinh Q Tran, Jonah Samost, et al. 2023. Recommender systems with generative retrieval. In *Advances in Neural Information Processing Systems*.

Steffen Rendle and Christoph Freudenthaler. 2014. Improving pairwise learning for item recommendation from implicit feedback. In *Proceedings of the 7th ACM international conference on Web search and data mining*, pages 273–282.

Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. 2009. Bpr: Bayesian personalized ranking from implicit feedback. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*.

Fei Sun, Jun Liu, Jian Wu, Changhua Pei, Xiao Lin, Wenwu Ou, and Peng Jiang. 2019. Bert4rec: Sequential recommendation with bidirectional encoder representations from transformer. In *Proceedings of the 28th ACM international conference on information and knowledge management*, pages 1441–1450.

Jiaxi Tang and Ke Wang. 2018. Personalized top-n sequential recommendation via convolutional sequence embedding. In *Proceedings of the eleventh ACM international conference on web search and data mining*, pages 565–573.

Lei Wang and Ee-Peng Lim. 2023. Zero-shot next-item recommendation using large pretrained language models. *arXiv preprint arXiv:2304.03153*.

Jason Wei, Maarten Bosma, Vincent Y Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. 2022. Finetuned language models are zero-shot learners. In *Proceedings of the 10th International Conference on Learning Representations*.

Hong-Jian Xue, Xinyu Dai, Jianbing Zhang, Shujian Huang, and Jiajun Chen. 2017. Deep matrix factorization models for recommender systems. In *IJCAI*, volume 17, pages 3203–3209. Melbourne, Australia.

Junjie Zhang, Ruobing Xie, Yupeng Hou, Wayne Xin Zhao, Leyu Lin, and Ji-Rong Wen. 2023. Recommendation as instruction following: A large language model empowered recommendation approach. *arXiv preprint arXiv:2305.07001*.

Tingting Zhang, Pengpeng Zhao, Yanchi Liu, Victor S Sheng, Jiajie Xu, Deqing Wang, Guanfeng Liu, Xiaofang Zhou, et al. 2019. Feature-level deeper self-attention network for sequential recommendation. In *IJCAI*, pages 4320–4326.

Kun Zhou, Hui Wang, Wayne Xin Zhao, Yutao Zhu, Sirui Wang, Fuzheng Zhang, Zhongyuan Wang, and Ji-Rong Wen. 2020. S3-rec: Self-supervised learning for sequential recommendation with mutual information maximization. In *Proceedings of the 29th ACM international conference on information & knowledge management*, pages 1893–1902.

Dataset	# Users	# Items	# Interactions	Sparsity (%)
Toys & Games	19,412	11,924	167,597	99.93
Beauty	22,363	12,101	198,502	99.93
Sports & Outdoors	35,598	18,357	296,337	99.95

Table 7: Statistics of the datasets.

A P5-XL Experimental Setting and Additional Results

A.1 Experimental Setting

We generate P5 prompts using the source code provided by the P5 authors [7]. However, for a fair comparison, we update the data pre-processing to be consistent with our method and the other baselines. Specifically, we apply random instead of sequential indexing when mapping the item IDs. As pointed out by Rajput et al. 2023, the sequential indexing of items (*e.g.*, the purchase sequence of the first user in Toys & Games is mapped into '1, 2, 3, 4, 5, 6, 7') in the original P5 pre-processing leads to data leakage (*e.g.*, given the train items, *i.e.*, '1, 2, 3, 4, 5, 6', the LLM can easily infer the test item, *i.e.*, '7'). Therefore, we adopt random mapping (*i.e.*, consecutive or similar-looking IDs may not imply any connection), which is consistent with our method. In addition, the original P5 pre-processing adopts leave-one-out split for retrieval and ranking, while splitting the dataset by 0.8:0.1:0.1 for the training, validation, and testing of rating prediction. This could result in data leakage, as the test interactions of one task might be included in the training set of another task. We instead adopt leave-one-out data split for all three recommendation tasks, which is consistent with our proposed method as well as the other baselines.

For a fair comparison, We apply the same backbone (FLAN-T5-XL), fine-tuning steps (70,000), batch size (16), and learning rate (0.001) as adopted by our proposed method. Following the original P5 code, we fine-tune a unified model with prompts of their proposed five task families (rating, sequential recommendation, explanation, review, and direct recommendation. The sequential recommendation and direct recommendation families are weighted 5 times higher than the rest families). In Tables 1, 2, and 3, we adopt prompt templates 2-1, 2-7, and 1-4 for evaluating the retrieval, ranking, and rating prediction performance of the P5-XL model, as these templates better suit the forms of the recommendation tasks (introduced in the second subsection of Section 4.1) than the other templates.

[7] https://github.com/jeykigung/P5

A.2 Additional Results

In Table 8, we report the ranking results of P5-XL evaluated with prompt template 5-5. We can tell that P5-XL (5-5) slightly fall behind P5-XL. Our proposed UT greatly outperforms both P5-XL and P5-XL (5-5), which again verifies that our proposed recommendation task prompts are more informative than the P5 ones.

B Dataset Statistics

Table 7 presents the statistics of the Amazon datasets, *i.e.*, Toys & Games, Beauty, and Sports & Outdoors [8] that we used to evaluate our proposed method as well as all the baselines.

C Examples and Statistics of the Proposed Data Samples

C.1 Statistics of the Data Samples

Table 10 presents the statistics of our proposed recommendation-task and auxiliary-task data samples. Consider the recommendation-task data samples, the training data samples are generated by swiping a sliding window of size $w = 20$ over the training split of the user sequence. The validation data samples consider only 3,000 users for each dataset for cost-efficient validation. We test on all users, therefore the counts of the testing data samples equal to the total number of users in the datasets. The auxiliary-task data samples, on the other hand, are generated using only the training splits. Notably, during training, we apply dynamic sampling that decide the negative items in the BPR data samples as well as the masked items/tokens in the MIM/MLM data samples on the fly. Such dynamic sampling helps to fully fine-tune the LLM backbones.

C.2 Examples of the Data Samples

In Table 11, we present examples of our proposed data samples. These data samples are generated with the training data split of an Amazon - Toys & Games user whose ID is 'A12HF3UBDV34RR'. Note that to fully fine-tune the LLM backbone, we apply dynamic sampling for the BPR and MIM/MLM data samples and decide the negative items and masked items/tokens on the fly. Here, we only present the BPR, MIM, and MLM data samples resulted from a single sampling.

[8] https://nijianmo.github.io/amazon/

Methods	Toys & Games					Beauty					Sports & Outdoors				
	NDCG @5	NDCG @10	HR @1	HR @5	HR @10	NDCG @5	NDCG @10	HR @1	HR @5	HR @10	NDCG @5	NDCG @10	HR @1	HR @5	HR @10
P5-XL	0.0290	0.0444	0.0097	0.0494	0.0977	0.0298	0.0456	0.0110	0.0498	0.0992	0.0286	0.0436	0.0097	0.0486	0.0957
P5-XL (5-5)	0.0274	0.0428	0.0089	0.0467	0.0948	0.0289	0.0443	0.0093	0.0497	0.0982	0.0275	0.0426	0.0091	0.0470	0.0943
UT [**Ours**]	**0.1536**	**0.1867**	**0.0831**	**0.2233**	**0.3259**	**0.1236**	**0.1537**	**0.0609**	**0.1863**	**0.2798**	**0.0867**	**0.1137**	**0.0381**	**0.1362**	**0.2202**

Table 8: Additional P5-XL Ranking results. Rows 1 and 3 are copied from Table 2.

Methods	NDCG @5	NDCG @10	HR @5	HR @10
UT [**Ours**]	**0.0079**	**0.0101**	0.0118	**0.0187**
UT+IE [**Ours**]	0.0076	0.0097	**0.0121**	0.0185

Table 9: Retrieval results on Sports & Outdoors with (UT+IE) or without (UT) IE data samples. Row 1 is copied from Table 1.

Input: What's the title of I1014? Output: Women's Dry-fit Tempo Shorts
Input: What's the brand of I1014? Output: Nike
Input: What's the price of I1014? Output: $31.8
...

Figure 3: Item embedding (IE) data samples.

D Mimicking Item Embedding

Our proposed data samples introduced in the main paper encode item correlations encompassed in users' preferences. We also explore encoding item correlations encompassed in item contents, *i.e.*, categories, descriptions, etc.

We observe that the conventional context-aware recommenders commonly integrate item contents to help the model better understand the items and achieve enhanced performance. *E.g.*, Hou et al. 2022 embed the concatenations of item content fields with BERT (Devlin et al., 2019). The learned item embeddings, $\mathbf{X} \in \mathbb{R}^{N \times d}$, where N is the number of the items and d is the dimension of the vector space, serve as initial representations of the items.

We mimic this item embedding (IE) process with natural language prompts. As shown in Figure 3, by asking questions about the properties of an item in the input and answering them in the output, we can generate item embedding data samples such as 'Input: What's the brand of I1014? Output: Nike'. We repeat such question answering process for various available item content fields, including title, categories, brand, price, attributes, and descriptions. These data samples represent knowledge about the items, but with natural language rather than numerical vectors. We expect that tuning LLMs with IE data samples can help them to comprehend the items in the target recommendation domain and enhance their performance.

To evaluate the IE data samples, we tune a **UT+IE** model, which augments the fine-tuning data of our UT model with IE data samples (the rest experimental settings of UT+IE and UT remain the same). We present its retrieval performance on Sports & Outdoors in Table 9. We observe no noticeable performance increase when incorporate IE data samples. The reason might be, the raw item content fields are noisy. *E.g.*, the description field is long and can contain noise such as hashtags and URLs. It has been shown (Cao et al., 2023) pre-processing the raw fields to extract fine-grained features helps to enhance context-aware recommenders. Inspired by this, in the future, we plan to improve the IE data samples by refining the item content fields.

Task	Toys & Games			Beauty			Sports & Outdoors		
	# Train	# Valid	# Test	# Train	# Valid	# Test	# Train	# Valid	# Test
Retrieval	30,761	3,000	19,412	36,582	3,000	22,363	47,320	3,000	35,598
Ranking	30,761	3,000	19,412	36,582	3,000	22,363	47,320	3,000	35,598
Rating prediction	30,761	3,000	19,412	36,582	3,000	22,363	47,320	3,000	35,598
MIM	DS	0	0	DS	0	0	DS	0	0
MLM	DS	0	0	DS	0	0	DS	0	0
BPR	DS	0	0	DS	0	0	DS	0	0

Table 10: Statistics of our proposed data samples. DS stands for dynamic sampling.

Task	Data sample
Retrieval	**Input**: A user has purchased the following Amazon products (arranged in chronological order, from earliest to most recent): Item ID: I9762, Title: Winstonia's 8 Wheels Combo Set Nail Art Polymer Slices Fimo Decal Pieces Accessories - Butterflies, Bows, Animals, Fruit, Flowers, Dragonflies, Cupcakes, Hearts; Item ID: I8123, Title: MASH Rhinestones 2400 Piece 12 Color Nail Art Nailart Manicure Wheels; Item ID: I158, Title: Aveeno Clear Complexion Daily Moisturizer, 4 Ounce; Item ID: I5324, Title: Bdellium Tools Professional Antibacterial Makeup Brush Studio Line - Precision Kabuki Airbrushed Effect 957; Item ID: I7522, Title: Bdellium Tools Professional Makeup Brush Green Bambu Series Smoky Eyes 5pc. Brush Set; Item ID: I7647, Title: real Techniques Stippling Brush; Item ID: I7811, Title: Maybelline New York Color Sensational High Shine Lipcolor, Coral Lustre 840, 0.12 Ounce; Item ID: I9440, Title: Bed Head BH313 Orange Crush 1-inch Styler; Item ID: I5046, Title: Herstyler Baby Curl Curling Iron, Purple; What would the user buy next? **Output**: I3977
Ranking	**Input**: A user has purchased the following Amazon products (arranged in chronological order, from earliest to most recent): Item ID: I9762, Title: Winstonia's 8 Wheels Combo Set Nail Art Polymer Slices Fimo Decal Pieces Accessories - Butterflies, Bows, Animals, Fruit, Flowers, Dragonflies, Cupcakes, Hearts; Item ID: I8123, Title: MASH Rhinestones 2400 Piece 12 Color Nail Art Nailart Manicure Wheels; Item ID: I158, Title: Aveeno Clear Complexion Daily Moisturizer, 4 Ounce; Item ID: I5324, Title: Bdellium Tools Professional Antibacterial Makeup Brush Studio Line - Precision Kabuki Airbrushed Effect 957; Item ID: I7522, Title: Bdellium Tools Professional Makeup Brush Green Bambu Series Smoky Eyes 5pc. Brush Set; Item ID: I7647, Title: real Techniques Stippling Brush; Item ID: I7811, Title: Maybelline New York Color Sensational High Shine Lipcolor, Coral Lustre 840, 0.12 Ounce; Item ID: I9440, Title: Bed Head BH313 Orange Crush 1-inch Styler; Item ID: I5046, Title: Herstyler Baby Curl Curling Iron, Purple; Which of the following candidate items would you recommend the user to buy next? Candidate items are: I10537, I11849, I2647, I10506, I377, I8136, I3598, I2316, I114, I10379, I6767, I2801, I4687, I3446, I7222, I5925, I4608, I2226, I2279, I11708, I4376, I8771, I6502, I8650, I7006, I11350, I6716, I4690, I11303, I3446, I8704, I4001, I9816, I1498, I6896, I1598, I7653, I2086, I12019, I3235, I12052, I27, I5786, I9936, I697, I10050, I447, I10898, I2093, I2618, I2044, I2618, I6924, I2769, I8117, I10772, I9252, I4668, I6982, I2234, I9894, I9441, I6514, I5519, I8620, I710, I10212, I8654, I7648, I11054, I1419, I10958, I334, I576, I1537, I8278, I3181, I189, I3510, I7974, I6010, I11187, I6465, I9596, I9356, I311, I2313, I7117, I9249, I643, I6732, I8803, I5499, I2434, I3977, I10691, I10707, I5553, I7999, I8672. **Output**: I3977
Rating prediction	**Input**: A user likes the following Amazon products: Item ID: I7522, Title: Bdellium Tools Professional Makeup Brush Green Bambu Series Smoky Eyes 5pc. Brush Set; Item ID: I7811, Title: Maybelline New York Color Sensational High Shine Lipcolor, Coral Lustre 840, 0.12 Ounce; The user dislikes the following Amazon products: Item ID: I7647, Title: real Techniques Stippling Brush; Item ID: I9440, Title: Bed Head BH313 Orange Crush 1-inch Styler; Item ID: I5046, Title: Herstyler Baby Curl Curling Iron, Purple; Predict whether the user would like the following item. Answer yes or no. Item ID: I3977, Title: L'Oreal Paris HiP Studio Secrets Professional Color Truth Cream Eyeliner, Brown, 0.159 Ounce **Output**: no
MIM	**Input**: A user has purchased the following Amazon products (arranged in chronological order, from earliest to most recent): Item ID: I9762, Title: Winstonia's 8 Wheels Combo Set Nail Art Polymer Slices Fimo Decal Pieces Accessories - Butterflies, Bows, Animals, Fruit, Flowers, Dragonflies, Cupcakes, Hearts; [masked item]; Item ID: I158, Title: Aveeno Clear Complexion Daily Moisturizer, 4 Ounce; Item ID: I5324, Title: Bdellium Tools Professional Antibacterial Makeup Brush Studio Line - Precision Kabuki Airbrushed Effect 957; Item ID: I7522, Title: Bdellium Tools Professional Makeup Brush Green Bambu Series Smoky Eyes 5pc. Brush Set; [masked item]; Item ID: I7811, Title: Maybelline New York Color Sensational High Shine Lipcolor, Coral Lustre 840, 0.12 Ounce; Item ID: I9440, Title: Bed Head BH313 Orange Crush 1-inch Styler; Item ID: I5046, Title: Herstyler Baby Curl Curling Iron, Purple; Item ID: I3977, Title: L'Oreal Paris HiP Studio Secrets Professional Color Truth Cream Eyeliner, Brown, 0.159 Ounce; What are the masked items, in chronological order? **Output**: Item ID: I8123, Title: MASH Rhinestones 2400 Piece 12 Color Nail Art Nailart Manicure Wheels; Item ID: I7647, Title: real Techniques Stippling Brush;
MLM	**Input**: Item ID: I7811, Title: Maybelline New York Color Sensational High Shine Lipcolor, Coral Lustre 840, 0.12 Ounce; Item ID: I9440, Title: Bed Head BH313 Orange Crush 1-inch Styler;
BPR	**Input**: A user has purchased the following Amazon products (arranged in chronological order, from earliest to most recent): Item ID: I9762, Title: Winstonia's 8 Wheels Combo Set Nail Art Polymer Slices Fimo Decal Pieces Accessories - Butterflies, Bows, Animals, Fruit, Flowers, Dragonflies, Cupcakes, Hearts; Item ID: I8123, Title: MASH Rhinestones 2400 Piece 12 Color Nail Art Nailart Manicure Wheels; Item ID: I158, Title: Aveeno Clear Complexion Daily Moisturizer, 4 Ounce; Item ID: I5324, Title: Bdellium Tools Professional Antibacterial Makeup Brush Studio Line - Precision Kabuki Airbrushed Effect 957; Item ID: I7522, Title: Bdellium Tools Professional Makeup Brush Green Bambu Series Smoky Eyes 5pc. Brush Set; Item ID: I7647, Title: real Techniques Stippling Brush; Item ID: I7811, Title: Maybelline New York Color Sensational High Shine Lipcolor, Coral Lustre 840, 0.12 Ounce; Item ID: I9440, Title: Bed Head BH313 Orange Crush 1-inch Styler; Item ID: I5046, Title: Herstyler Baby Curl Curling Iron, Purple; Which of the following two items would the user buy next? Item ID: I4168, Title: Sulfur Soap with Lanolin; Item ID: I3977, Title: L'Oreal Paris HiP Studio Secrets Professional Color Truth Cream Eyeliner, Brown, 0.159 Ounce; **Output**: Item ID: I3977, Title: L'Oreal Paris HiP Studio Secrets Professional Color Truth Cream Eyeliner, Brown, 0.159 Ounce;

Table 11: Examples of our proposed data samples.