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

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

Ming Wang
Amazon, United States
mingww@amazon.com

Jin Li
Amazon, United States
jincli@amazon.com

Jinmiao Fu
Amazon, United States
jinnmiaof@amazon.com

Xin Shen
Amazon, United States
xinshen@amazon.com

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

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

KDD 2023

Code will be released upon acceptance.

Reported by Zicong Dou

Introduction

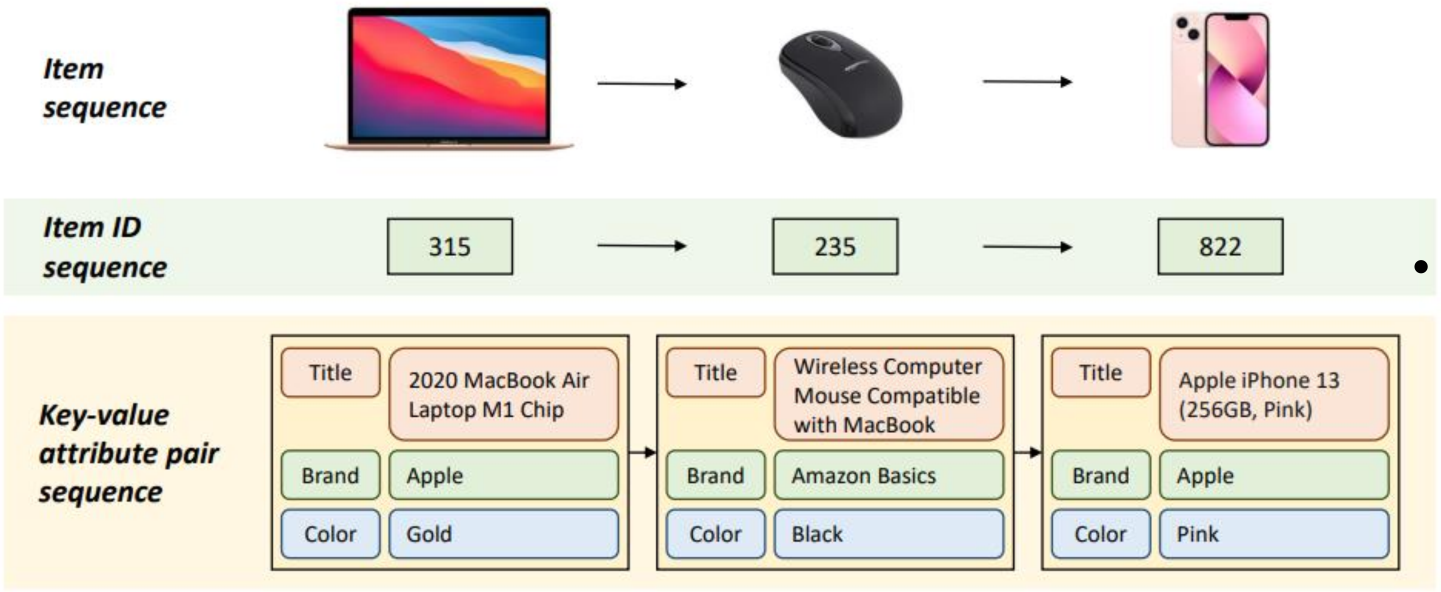


Figure 1: Input data comparison between item ID sequences for traditional sequential recommendation and key-value attribute pair sequences used in RECFORMER.

Contributions:

- We formulate items as key-value attribute pairs for the ID free sequential recommendation and propose a bidirectional Transformer structure to encode sequences of key-value pairs.

- We design the learning framework that helps the model learn users' preferences and transfer knowledge into different recommendation domains and cold-start items.

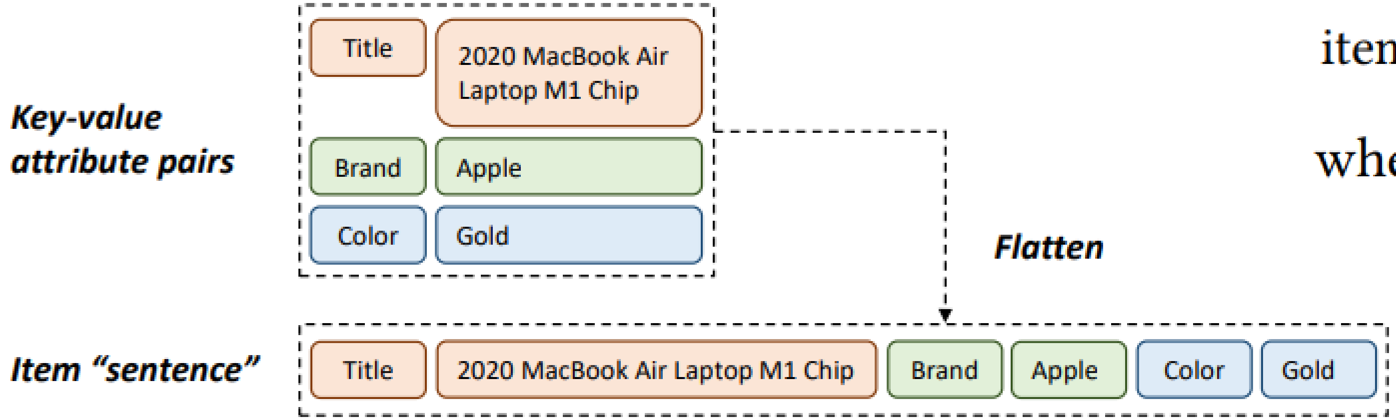


Figure 2: Model input construction. Flatten key-value attribute pairs into an item "sentence".

Problem Setup and Formulation

item set $\mathcal{I}$ $\quad s = \{i_1, i_2, \ldots, i_n\}$

where n is the length of s and $i \in \mathcal{I}$

attribute dictionary D_i

$\{(k_1, v_1), (k_2, v_2), \ldots, (k_m, v_m)\}$

$$(k, v) = \{w_1^k, \ldots, w_c^k, w_1^v, \ldots, w_c^v\}$$

$$T_i = \{k1, v1, k2, v2, \ldots, k_m, v_m\}$$

Method

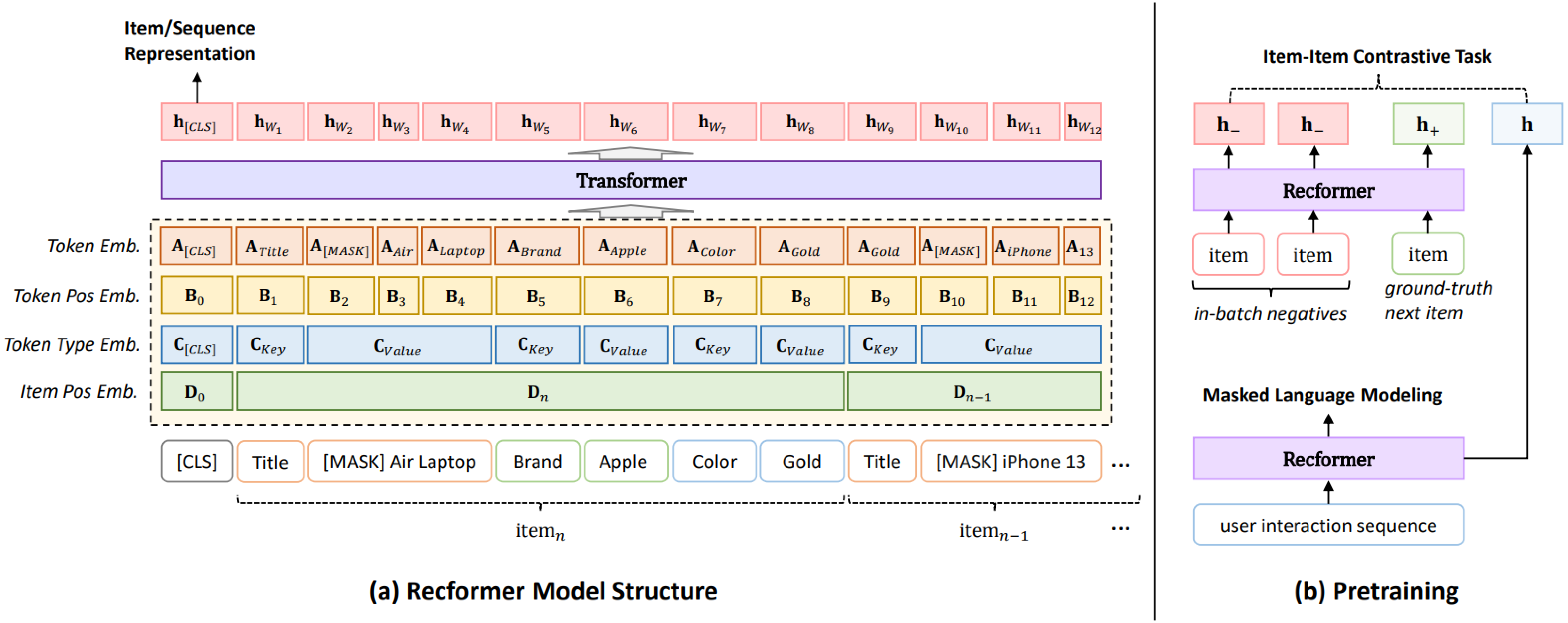


Figure 3: The overall framework of RECFORMER.

Method

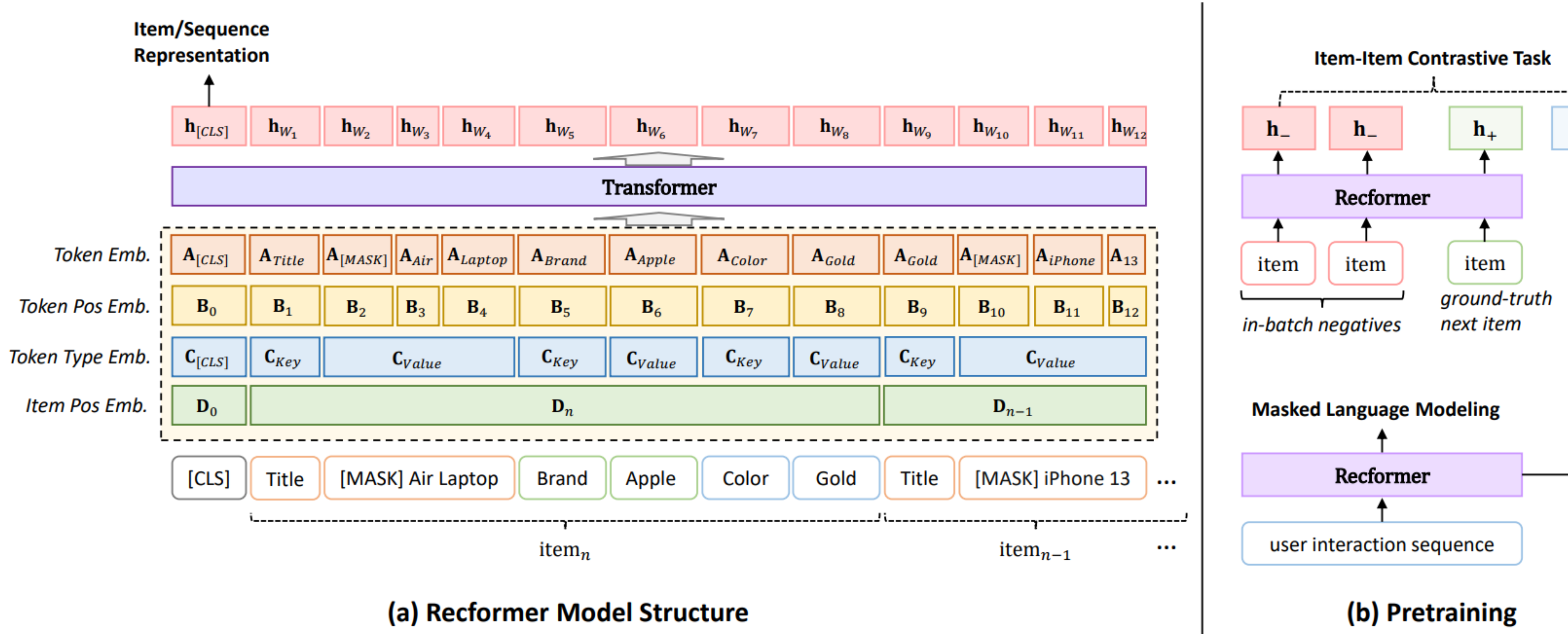


Figure 3: The overall framework of RECFORMER.

Embedding Layer

Token embedding

$$\mathbf{A} \in \mathbb{R}^{V_w \times d}$$

Token position embedding

$$\mathbf{B}_i \in \mathbb{R}^d$$

Token type embedding

$$\mathbf{C}_{\texttt{[CLS]}}, \mathbf{C}_{Key}, \mathbf{C}_{Value} \in \mathbb{R}^d$$

Item position embedding

$$\mathbf{D}_k \in \mathbb{R}^d \quad \mathbf{D} \in \mathbb{R}^{n \times d}$$

Model Inputs.

$$T_i = \{k1, v1, k2, v2, \ldots, k_m, v_m\}$$

$$(k, v) = \{w_1^k, \ldots, w_c^k, w_1^v, \ldots, w_c^v\}$$

$$s = \{i_1, i_2, \ldots, i_n\} \quad \{i_n, i_{n-1}, \ldots, i_1\}$$

$$X = \{\texttt{[CLS]}, T_n, T_{n-1}, \ldots, T_1\} \tag{1}$$

$$\mathbf{E}_w = \text{LayerNorm}(\mathbf{A}_w + \mathbf{B}_w + \mathbf{C}_w + \mathbf{D}_w) \tag{2}$$

where $\mathbf{E}_w \in \mathbb{R}^d$

$$\mathbf{E}_X = [\mathbf{E}_{\texttt{[CLS]}}, \mathbf{E}_{w_1}, \ldots, \mathbf{E}_{w_l}] \tag{3}$$

where $\mathbf{E}_X \in \mathbb{R}^{(l+1)\times d}$ and l is the maximum length of tokens in a user's interaction sequence.

Method

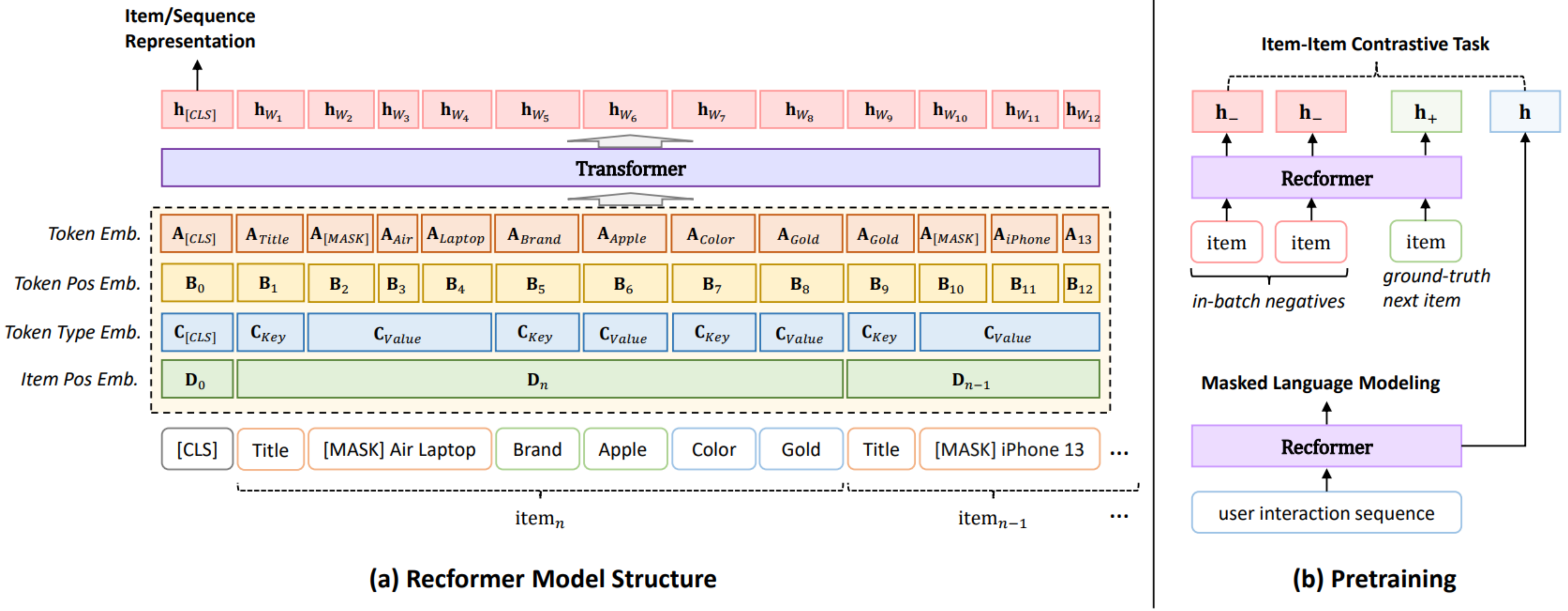


Figure 3: The overall framework of RECFORMER.

$$[\mathbf{h}_{[\mathrm{CLS}]}, \mathbf{h}_{w_1}, \ldots, \mathbf{h}_{w_l}] = \mathrm{Longformer}([\mathbf{E}_{[\mathrm{CLS}]}, \mathbf{E}_{w_1}, \ldots, \mathbf{E}_{w_l}]) \quad (4)$$

where $\mathbf{h}_w \in \mathbb{R}^d$.

$X = \{[\mathrm{CLS}], T_i\}$ $\quad \mathbf{h}_{[\mathrm{CLS}]}$ $\quad \mathbf{h}_i$

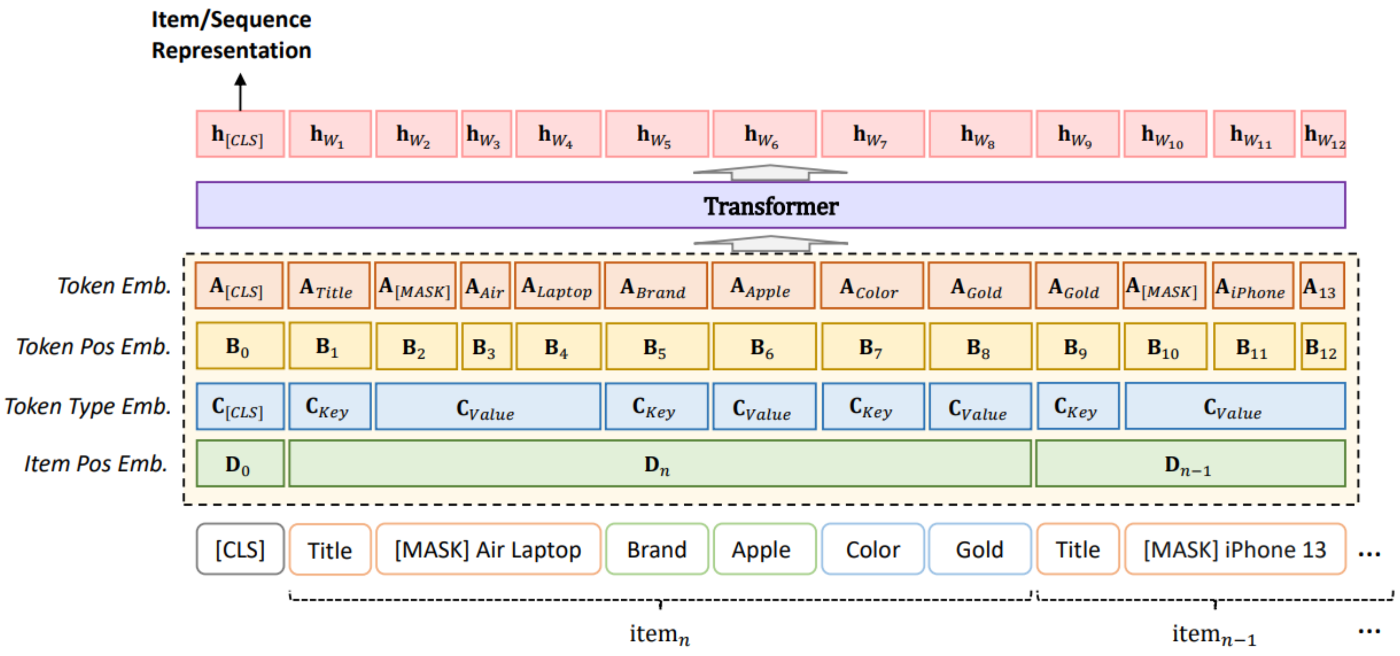
$$r_{i,s} = \frac{\mathbf{h}_i^\top \mathbf{h}_s}{\|\mathbf{h}_i\| \cdot \|\mathbf{h}_s\|} \quad (5)$$

where $r_{i,s} \in \mathbb{R}$ is the relevance of item i being the next item given s.

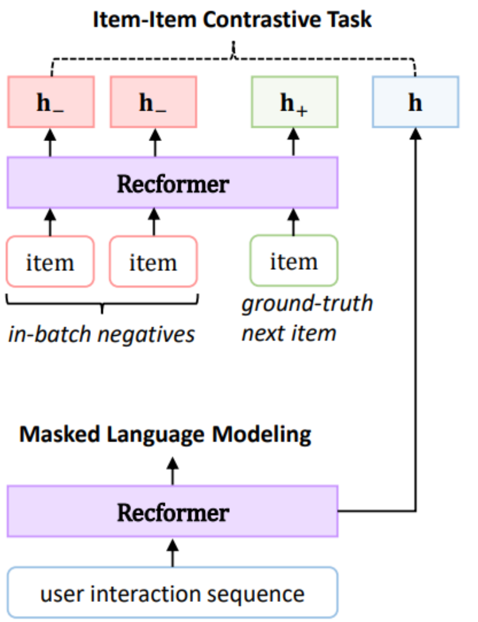
$$\hat{i}_s = \mathrm{argmax}_{i \in \mathcal{I}}(r_{i,s}) \quad (6)$$

where $\hat{i}_s$ is the predicted item given user interaction sequence s.

Method



(a) Recformer Model Structure



(b) Pretraining

we replace the token with (1) the [MASK] with probability 80%; (2) a random token with probability 10%; (3) the unchanged token with probability 10%. The MLM loss is calculated as:

$$\mathbf{m} = \mathrm{LayerNorm}(\mathrm{GELU}(\mathbf{W}_h \mathbf{h}_w + \mathbf{b}_h)) \tag{7}$$

$$p = \mathrm{Softmax}(\mathbf{W}_0 \mathbf{m} + \mathbf{b}_0) \tag{8}$$

$$\mathcal{L}_{\mathrm{MLM}} = -\sum_{i=0}^{|\mathcal{V}|} y_i \log(p_i) \tag{9}$$

where $\mathbf{W}_h \in \mathbb{R}^{d\times d}$, $\mathbf{b}_h \in \mathbb{R}^d$, $\mathbf{W}_0 \in \mathbb{R}^{|\mathcal{V}|\times d}$, $\mathbf{b}_0 \in \mathbb{R}^{|\mathcal{V}|}$, GELU is the GELU activation function [10] and $\mathcal{V}$ is the vocabulary used in the language model.

$$\mathcal{L}_{\mathrm{IIC}} = -\log \frac{e^{\mathrm{sim}(\mathbf{h}_s, \mathbf{h}_i^+)/\tau}}{\sum_{i\in\mathcal{B}} e^{\mathrm{sim}(\mathbf{h}_s, \mathbf{h}_i)/\tau}} \tag{10}$$

$\mathbf{h}_i^+$ is the representation of the ground truth next item; $\mathcal{B}$ is the ground truth item set in one batch and τ is a temperature parameter.

$$\mathcal{L}_{\mathrm{PT}} = \mathcal{L}_{\mathrm{IIC}} + \lambda \cdot \mathcal{L}_{\mathrm{MLM}} \tag{11}$$

$$\mathcal{L}_{\mathrm{FT}} = -\log \frac{e^{\mathrm{sim}(\mathbf{h}_s, \mathbf{I}_i^+)/\tau}}{\sum_{i\in\mathcal{I}} e^{\mathrm{sim}(\mathbf{h}_s, \mathbf{I}_i)/\tau}} \tag{12}$$

where $\mathbf{I}_i$ is the item feature of item i.

Experiments

Table 1: Statistics of the datasets after preprocessing. Avg. n denotes the average length of item sequences.

Datasets	#Users	#Items	#Inters.	Avg. n	Density
Pre-training	3,613,906	1,022,274	33,588,165	9.29	9.1e-6
-Training	3,501,527	954,672	32,291,280	9.22	9.0e-6
-Validation	112,379	67,602	1,296,885	11.54	1.7e-4
Scientific	11,041	5,327	76,896	6.96	1.3e-3
Instruments	27,530	10,611	231,312	8.40	7.9e-4
Arts	56,210	22,855	492,492	8.76	3.8e-4
Office	101,501	27,932	798,914	7.87	2.8e-4
Games	11,036	15,402	100,255	9.08	5.9e-4
Pet	47,569	37,970	420,662	8.84	2.3e-4

Experiments

Table 2: Performance comparison of different recommendation models. The best and the second-best performance is bold and underlined respectively. Improv. denotes the relative improvement of RECFORMER over the best baselines.

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

Experiments

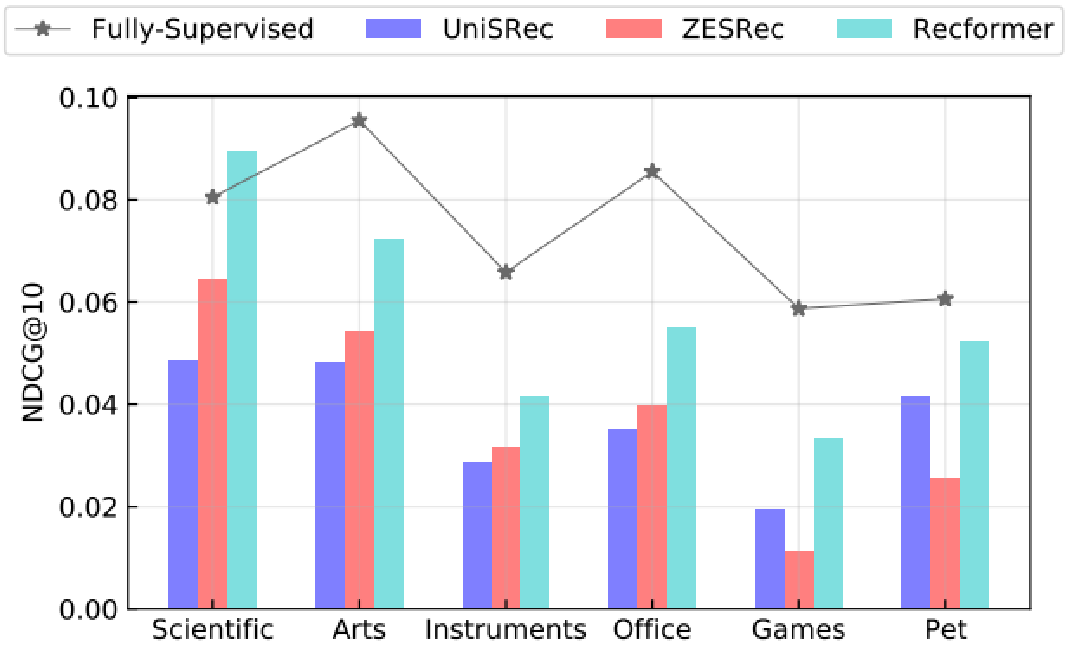


Figure 4: Performance (NDCG@10) of three Text-Only methods under the zero-shot setting. Fully-Supervised denotes the average scores of three classical ID-Only methods (i.e., SASRec, BERT4Rec, GRU4Rec) trained with all training data.

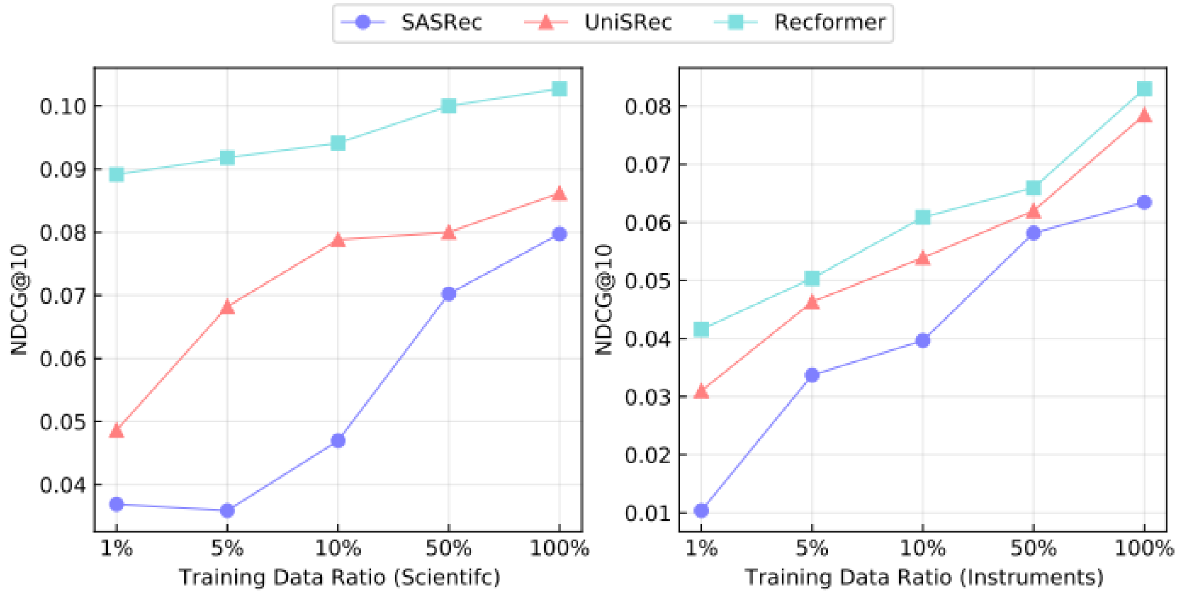


Figure 5: Performance (NDCG@10) of SASRec, UniSRec, RECFORMER over different sizes (i.e., 1%, 5%, 10%, 50%, 100%) of training data.

Experiments

Table 3: Performance of models compared between in-set items and cold-start items on four datasets. N@10 and R@10 stand for NDCG@10 and Recall@10 respectively.

Dataset	Metric	SASRec		UniSRec		RECFORMER	
		In-Set	Cold	In-Set	Cold	In-Set	Cold
Scientific	N@10	0.0775	0.0213	0.0864	0.0441	0.1042	0.0520
	R@10	0.1206	0.0384	0.1245	0.0721	0.1417	0.0897
Instruments	N@10	0.0669	0.0142	0.0715	0.0208	0.0916	0.0315
	R@10	0.1063	0.0309	0.1094	0.0319	0.1130	0.0468
Arts	N@10	0.1039	0.0071	0.1174	0.0395	0.1568	0.0406
	R@10	0.1645	0.0129	0.1736	0.0666	0.1866	0.0689
Pet	N@10	0.0597	0.0013	0.0771	0.0101	0.0994	0.0225
	R@10	0.0934	0.0019	0.1115	0.0175	0.1192	0.0400

Experiments

Table 4: Ablation study on two downstream datasets. The best and the second-best scores are bold and underlined respectively.

Variants	Scientific			Instruments		
	NDCG@10	Recall@10	MRR	NDCG@10	Recall@10	MRR
(0) Recformer	**0.1027**	**0.1448**	**0.0951**	**0.0830**	**0.1052**	**0.0807**
(1) w/o two-stage finetuning	0.1023	<u>0.1442</u>	<u>0.0948</u>	0.0728	0.1005	0.0685
(1) + (2) freezing word emb. & item emb.	<u>0.1026</u>	0.1399	0.0942	0.0728	<u>0.1015</u>	0.0682
(1) + (3) trainable word emb. & item emb.	0.0970	0.1367	0.0873	<u>0.0802</u>	<u>0.1015</u>	0.0759
(1) + (4) trainable item emb. & freezing word emb.	0.0965	0.1383	0.0856	0.0801	0.1014	<u>0.0760</u>
(5) w/o pre-training	0.0722	0.1114	0.0650	0.0598	0.0732	0.0584
(6) w/o item position emb. & token type emb.	0.1018	0.1427	0.0945	0.0518	0.0670	0.0501

Experiments

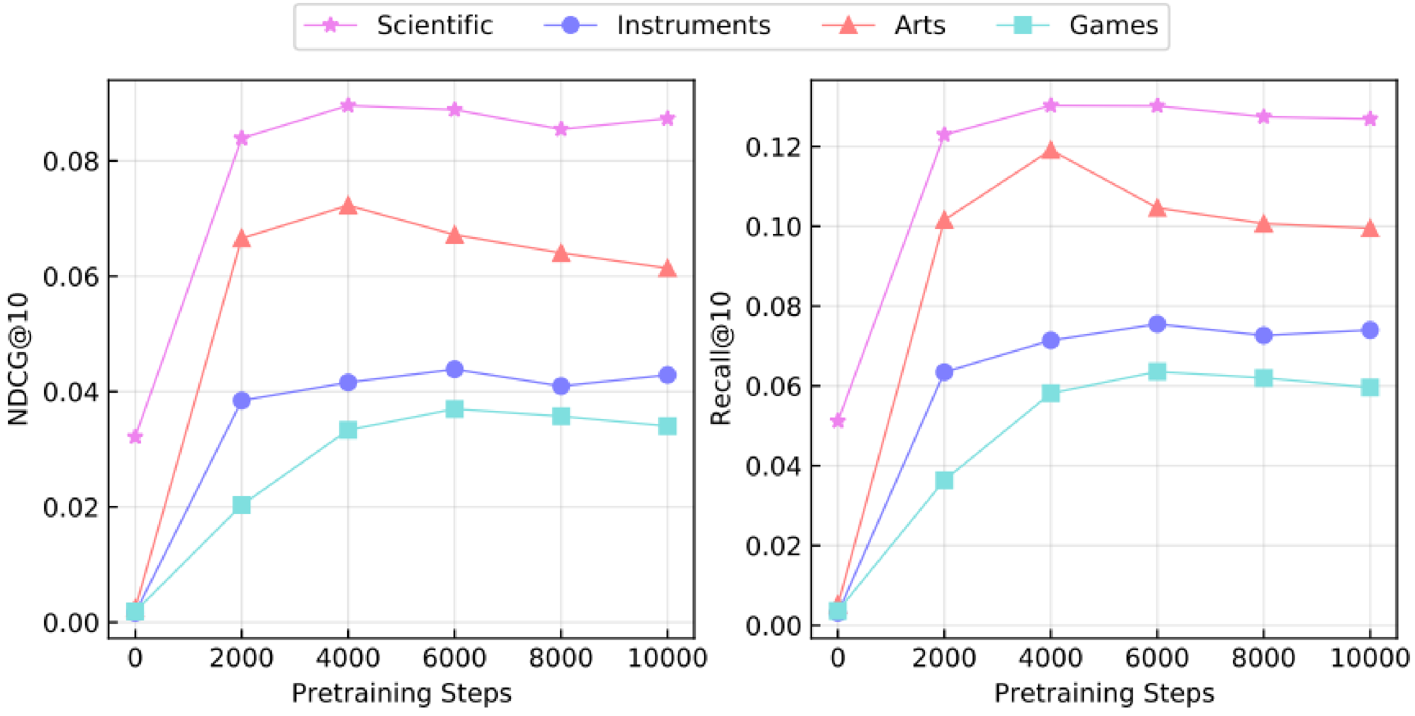


Figure 6: RECFORMER zero-shot recommendation performance (NDCG@10 and Recall@10) over different pre-training steps.

Method

Algorithm 1: Two-Stage Finetuning

Input: D_{train}, D_{valid}, $\mathcal{I}$, M
Hyper-parameters: n_{epoch}
Output: M', $\mathbf{I}'$

```
M ← initialized with pre-trained parameters
p ← metrics are initialized with 0
Stage 1
for n in n_epoch do
    I ← Encode(M, 𝓘)
    M ← Train(M, I, D_train)
    p' ← Evaluate(M, I, D_valid)
    if p' > p then
        M', I' ← M, I
        p ← p'
    end if
end for
Stage 2
M ← M'
for n in n_epoch do
    M ← Train(M, I', D_train)
    p' ← Evaluate(M, I', D_valid)
    if p' > p then
        M' ← M
        p ← p'
    end if
end for
return M', I'
```