Enhancing Sequential Recommendation with Graph Contrastive Learning

Yixin Zhang[1*], **Yong Liu**[3*], **Yonghui Xu**[2†], **Hao Xiong**[5], **Chenyi Lei**[5], **Wei He**[1],
Lizhen Cui[1,2†] and **Chunyan Miao**[3,4]
[1]School of Software, Shandong University, China
[2]Joint SDU-NTU Centre for Artificial Intelligence Research (C-FAIR), Shandong University, China
[3]Alibaba-NTU Singapore JRI & LILY Research Centre, Nanyang Technological University, Singapore
[4]School of Computer Science and Engineering, Nanyang Technological University, Singapore
[5]Alibaba Group, China
yixinzhang@mail.sdu.edu.cn, {stephenliu, ascymiao}@ntu.edu.sg, xu.yonghui@hotmail.com,
{songling.xh, chenyi.lcy}@alibaba-inc.com, {hewei, clz}@sdu.edu.cn

IJCAI 2022

Reported by Nengqiang Xiang

Introduction

- Ignoring the correlation between users with similar behavior.
- Using the item prediction task to train models, but the user behavior data is very sparse.
- Based on implicit feedback sequences, which may include noise information.

- Building a Weighted Item Transition Graph (WITG) to provide global context information.
- Neighborhood sampling on WITG is performed to build augmented graph views for each interaction sequence.
- Graph contrastive is employed to learn augmented representations for the user interaction sequence.

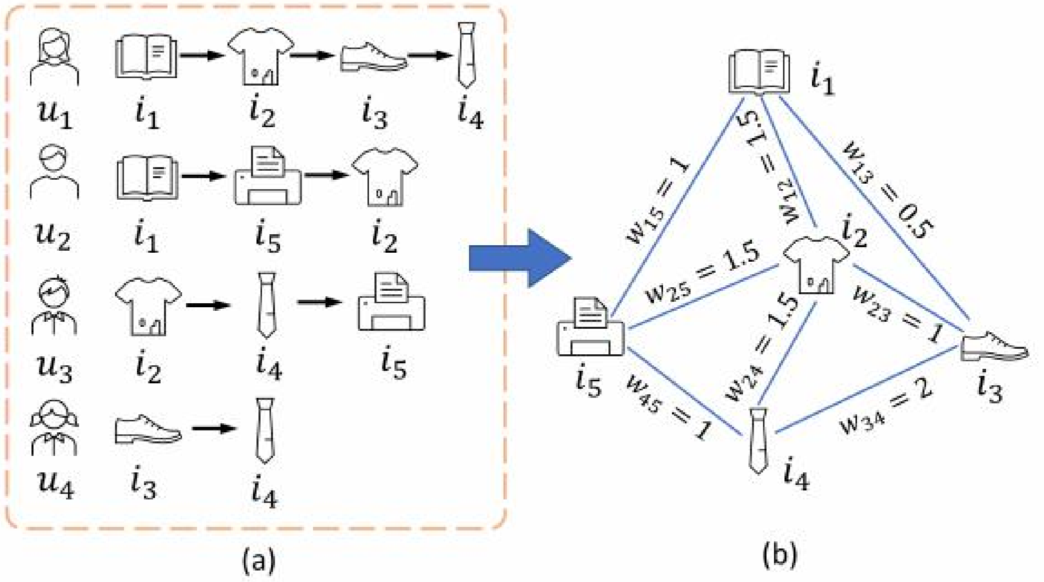


Figure 1: An example showing the transition graph construction procedure, where (a) shows the observed user behavior sequences, and (b) illuminates the weighted transition graph.

Method

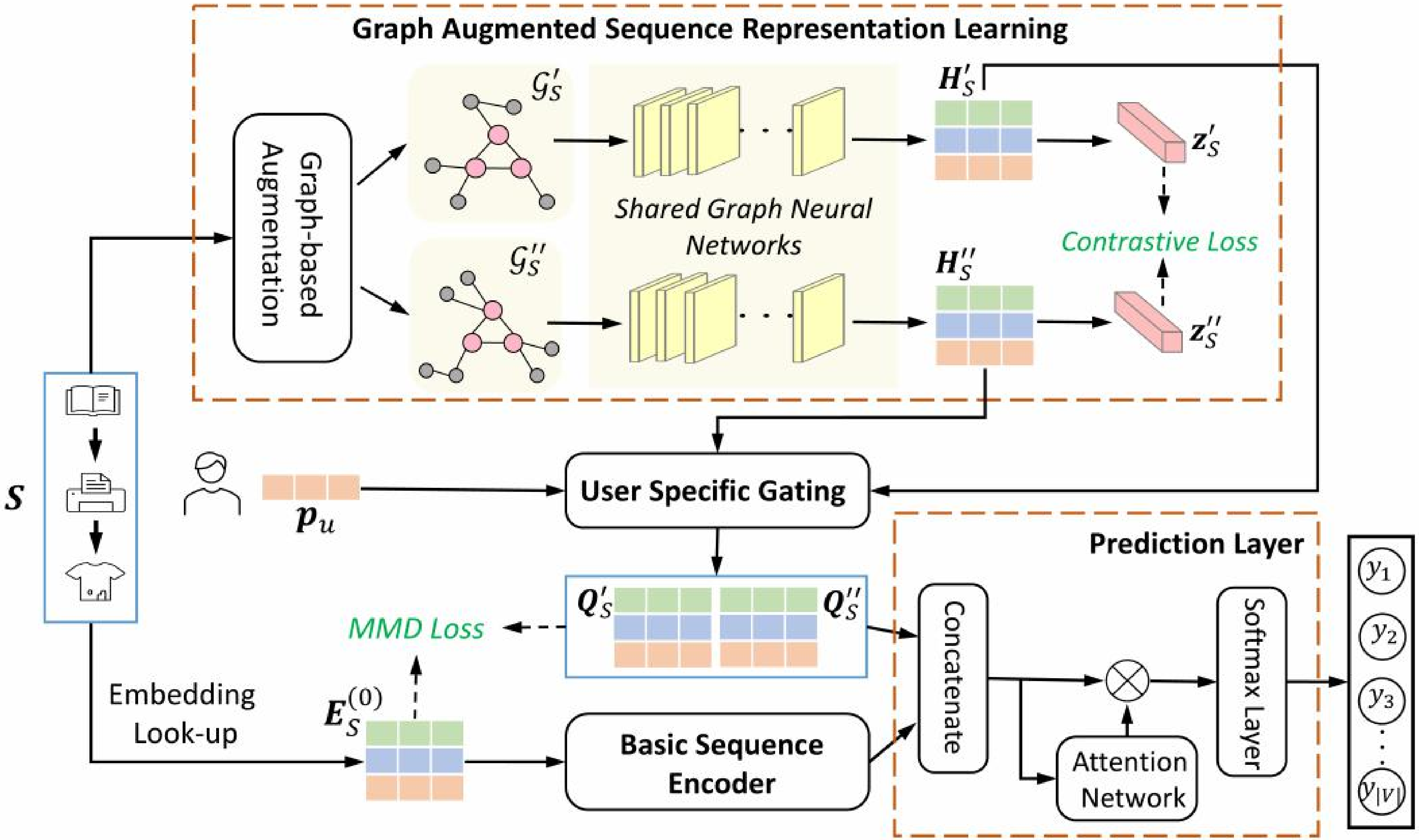
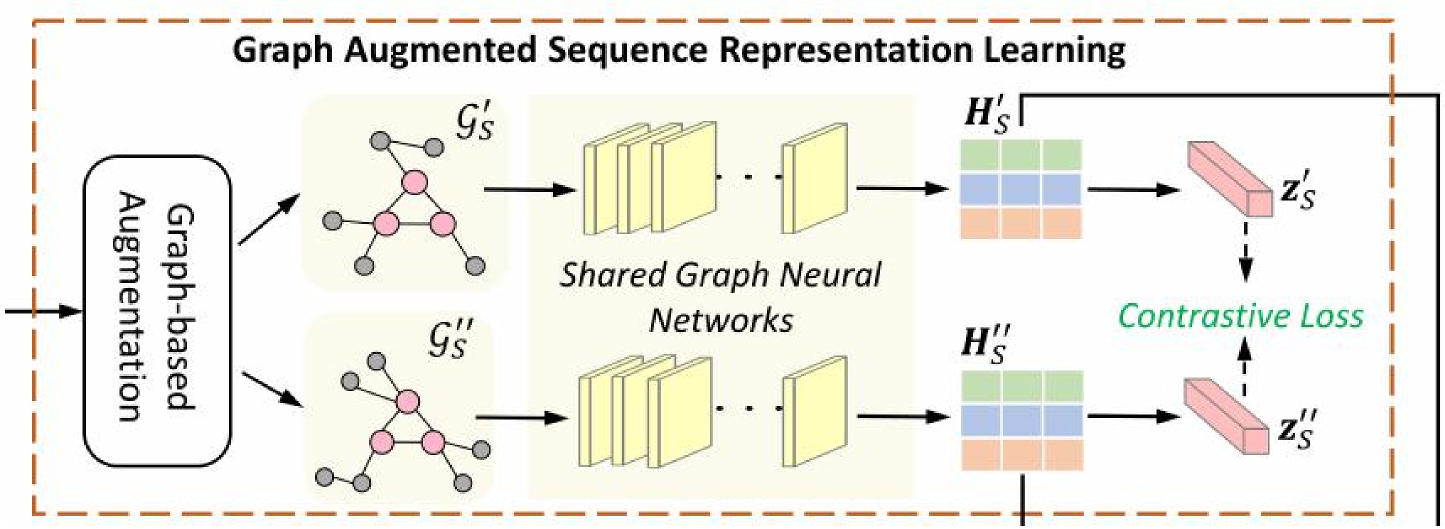


Figure 2: The framework of the proposed GCL4SR model.

Method



Graph-based Augmentation:

$$\widehat{w}(v_t, v_j) = w(v_t, v_j)\left(\frac{1}{\deg(v_i)} + \frac{1}{\deg(v_j)}\right), \quad (1)$$

Shared Graph Neural Networks:

$$\mathbf{a}_{v_i}^{(t)} = \text{Aggregate}^{(t)}\left(\{\mathbf{h}_{v_j}^{(t-1)} : v_j \in N'_{v_i}\}\right),$$
$$\mathbf{h}_{v_i}^{(t)} = \text{Combine}^{(t)}\left(\mathbf{a}_{v_i}^{(t)}, \mathbf{h}_{v_i}^{(t-1)}\right), \quad (2)$$

Graph Contrastive Learning Objective:

$$\mathcal{L}_{GCL}(S) = \sum_{S \in \mathcal{D}} -\log \frac{\exp\left(cos(\mathbf{z}'_S, \mathbf{z}''_S)/\tau\right)}{\sum_{K \in \mathcal{D}} \exp\left(cos(\mathbf{z}'_S, \mathbf{z}''_K)/\tau\right)}, \quad (3)$$

Symbol Definition:

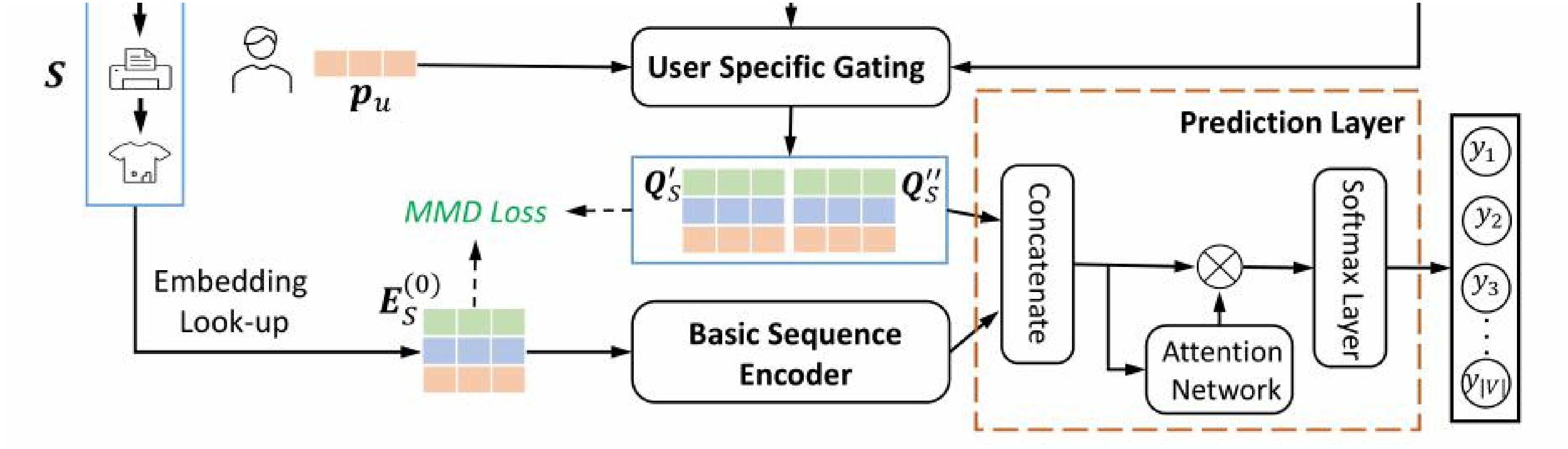
$\mathbf{p}_u \in \mathbb{R}^{1\times d}$: the user u's embedding

$\mathbf{e}_i \in \mathbb{R}^{1\times d}$: the item i's embedding

$\mathbf{E}_S^{(0)} \in \mathbb{R}^{n\times d}$: the initial embedding of the sequence S

$\mathbf{E} \in \mathbb{R}^{|V|\times d}$: the embeddings of all items

Method



User-specific Gating

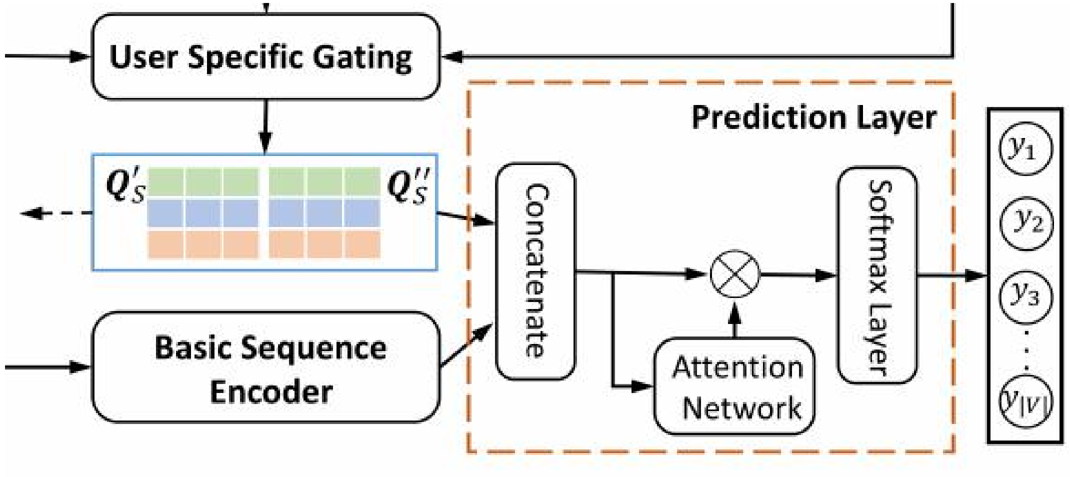
$$\mathbf{Q}_S' = \mathbf{H}_S' \otimes \sigma(\mathbf{H}_S'\mathbf{W}_{g1} + \mathbf{W}_{g2}\mathbf{p}_u^\top), \qquad (4)$$

Representation Alignment Objective:

$$MMD(\mathbf{X},\mathbf{Y}) = \frac{1}{m^2}\sum_{a=1}^{m}\sum_{b=1}^{m}\mathcal{K}(\mathbf{x}_a,\mathbf{x}_b) + \frac{1}{\widetilde{m}^2}\sum_{a=1}^{\widetilde{m}}\sum_{b=1}^{\widetilde{m}}\mathcal{K}(\mathbf{y}_a,\mathbf{y}_b) - \frac{2}{m\widetilde{m}}\sum_{a=1}^{m}\sum_{b=1}^{\widetilde{m}}\mathcal{K}(\mathbf{x}_a,\mathbf{y}_b), \qquad (5)$$

$$\mathcal{L}_{MM}(S) = MMD(\mathbf{E}_S^{(0)}, \mathbf{Q}_S') + MMD(\mathbf{E}_S^{(0)}, \mathbf{Q}_S''). \qquad (6)$$

Method



Basic Sequence Encoder

$$\mathbf{H}^{\ell} = FFN(Concat(head_1, \ldots, head_h)\mathbf{W}^h),$$
$$head_i = Attention(\mathbf{H}^{\ell-1}\mathbf{W}_i^Q, \mathbf{H}^{\ell-1}\mathbf{W}_i^K, \mathbf{H}^{\ell-1}\mathbf{W}_i^V), \quad (7)$$

$$Attention(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = softmax(\frac{\mathbf{Q}\mathbf{K}^\top}{\sqrt{d}})\mathbf{V}, \quad (8)$$

Prediction Layer

$$\mathbf{M} = AttNet(Concat(\mathbf{Q}'_S, \mathbf{Q}''_S, \mathbf{H}^{\ell})\mathbf{W}_T), \quad (9)$$

$$\hat{\mathbf{y}}^{(S)} = softmax(\mathbf{M}\mathbf{E}^\top), \quad (10)$$

Multi Task Learning

$$\mathcal{L}_{main} = -\sum_{S_u \in \mathcal{D}} \sum_{k=1}^{|S_u|-1} \log\left(\hat{\mathbf{y}}^{(S_u^{1:k})}(v_u^{k+1})\right), \quad (11)$$

$$\mathcal{L} = \mathcal{L}_{main} + \sum_{S_u \in \mathcal{D}} \sum_{k=1}^{|S_u|-1} \lambda_1 \mathcal{L}_{GCL}(S_u^{1:k}) + \lambda_2 \mathcal{L}_{MM}(S_u^{1:k}), \quad (12)$$

Experiments

	Home	Phones	Comics	Poetry
# Users	66,519	27,879	13,810	3,522
# Items	28,237	10,429	16,630	2,624
# Interactions	551,682	194,439	343,587	40,703
# Nodes of $\mathcal{G}$	28,237	10,429	16,630	2,624
# Edges of $\mathcal{G}$	1,617,638	430,940	1,310,952	122,700

Table 1: The statistics of experimental datasets.

Experiments

Datasets	Metrics	LightGCN	FPMC	GRU4Rec	Caser	SASRec	HGN	SR-GNN	GC-SAN	GCE-GNN	CL4SRec	S^3-Rec	GCL4SR
Home	HR@10	0.0160	0.0162	0.0210	0.0101	0.0228	0.0152	0.0201	0.0281	0.0259	0.0266	0.0280	**0.0313**
	HR@20	0.0250	0.0218	0.0330	0.0173	0.0316	0.0231	0.0292	0.0394	0.0359	0.0387	0.0406	**0.0422**
	N@10	0.0085	0.0097	0.0110	0.0051	0.0141	0.0083	0.0123	0.0174	0.0161	0.0160	0.0169	**0.0190**
	N@20	0.0108	0.0111	0.0140	0.0068	0.0163	0.0103	0.0146	0.0197	0.0186	0.0186	0.0196	**0.0218**
Phones	HR@10	0.0687	0.0634	0.0835	0.0435	0.0883	0.0680	0.0778	0.0881	0.0946	0.0929	0.1037	**0.1171**
	HR@20	0.1012	0.0854	0.1213	0.0647	0.1213	0.0990	0.1114	0.1232	0.1304	0.1305	0.1428	**0.1666**
	N@10	0.0370	0.0374	0.0459	0.0233	0.0511	0.0364	0.0427	0.0500	0.0543	0.0533	0.0594	**0.0665**
	N@20	0.0452	0.0430	0.0554	0.0287	0.0594	0.0442	0.0512	0.0588	0.0634	0.0627	0.0693	**0.0790**
Poetry	HR@10	0.1411	0.1275	0.1414	0.1068	0.1428	0.1034	0.1193	0.1309	0.1533	0.1496	0.1613	**0.1638**
	HR@20	0.2127	0.1851	0.2104	0.1567	0.2030	0.1545	0.1723	0.1936	0.2229	0.2164	0.2277	**0.2428**
	N@10	0.0771	0.0704	0.0783	0.0607	0.0829	0.0597	0.0686	0.0732	0.0859	0.0838	**0.0915**	0.0914
	N@20	0.0954	0.0849	0.0956	0.0732	0.0980	0.0725	0.0818	0.0891	0.1035	0.1004	0.1108	**0.1112**
Comics	HR@10	0.1106	0.1382	0.1593	0.1156	0.1709	0.1242	0.1481	0.1638	0.1722	0.1751	0.1781	**0.1829**
	HR@20	0.1672	0.1736	0.2058	0.1499	0.2100	0.1704	0.1857	0.2048	0.2232	0.2172	**0.2258**	0.2249
	N@10	0.0587	0.1019	0.1096	0.0790	0.1276	0.0743	0.1067	0.1189	0.1222	0.1235	0.1234	**0.1312**
	N@20	0.0730	0.1108	0.1213	0.0876	0.1374	0.0859	0.1161	0.1292	0.1325	0.1341	0.1354	**0.1417**

Table 2: The performance achieved by different methods. The best results are in **boldface**, and the second best results are underlined.

Experiments

Method	Poetry		Phones	
	HR@20	N@20	HR@20	N@20
GCL4SR	0.2428	**0.1112**	**0.1666**	**0.0790**
$\text{GCL4SR}_{\text{w/o G}}$	**0.2433**	0.1095	0.1607	0.0734
$\text{GCL4SR}_{\text{w/o GM}}$	0.2138	0.0958	0.1423	0.0713
$\text{GCL4SR}_{\text{w/o W}}$	0.2172	0.0979	0.1500	0.0694
SASRec	0.2030	0.0980	0.1213	0.0594

Table 3: The performance achieved by GCL4SR variants and SAS-Rec on Poetry and Phones datasets.

Experiments

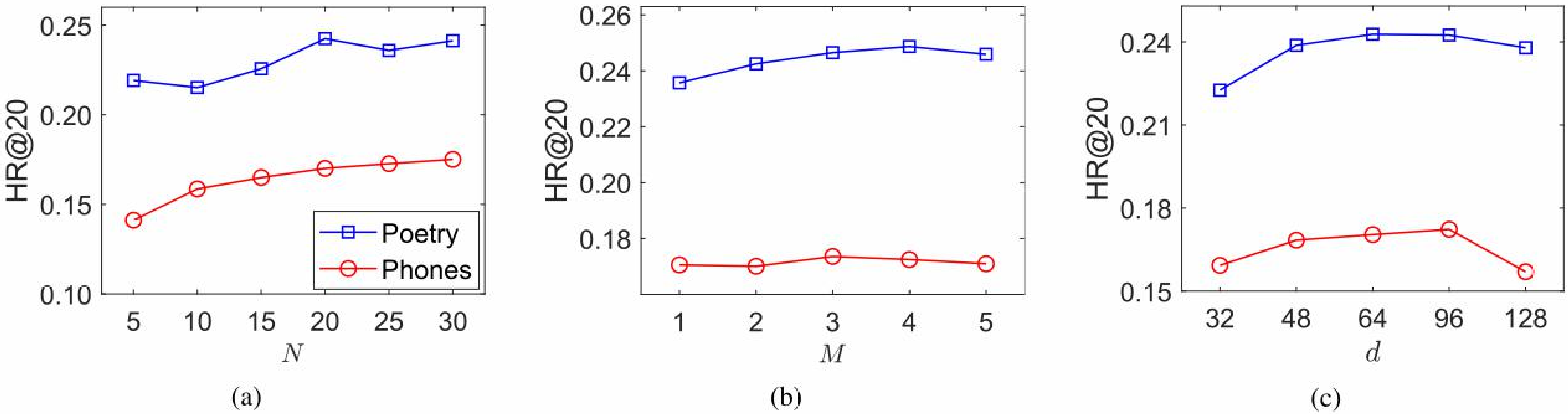


Figure 3: The performance trends of GCL4SR with respect to different settings of M, N, and d on Poetry and Phones datasets.

Experiments

Method	Poetry		Phones	
	HR@20	N@20	HR@20	N@20
HGN	0.1545	0.0725	0.0990	0.0442
GCL4SR-HGN	0.1712	0.0763	0.1064	0.0475
GRU4Rec	0.2104	0.0956	0.1213	0.0554
GCL4SR-GRU	0.2362	0.1057	0.1622	0.0763
SASRec	0.2030	0.0980	0.1213	0.0594
GCL4SR-SAS	**0.2428**	**0.1112**	**0.1666**	**0.0790**

Table 4: The performance of HGN, GRU4Rec, SASRec, and GCL4SR with different basic sequence encoders.

Thanks