Self-supervised Graph Neural Networks for Multi-behavior Recommendation

Yu Tian[1], Jianxin Chang[2], Yanan Niu[2], Yang Song[2], Chenliang Li[1†]
[1]Key Laboratory of Aerospace Information Security and Trusted Computing, Ministry of Education, School of Cyber Science and Engineering, Wuhan University, Wuhan, 430072, China
s.braylon1002@gmail.com; cllee@whu.edu.cn
[2]Kuaishou Technology Co., Ltd., Beijing, 10010, China
{changjianxin,niuyanan,yangsong}@kuaishou.com

IJCAI 2022

2022.5.29 • ChongQing

gesis
Leibniz-Institut
für Sozialwissenschaften

Reported by Gu Tang

Introduction

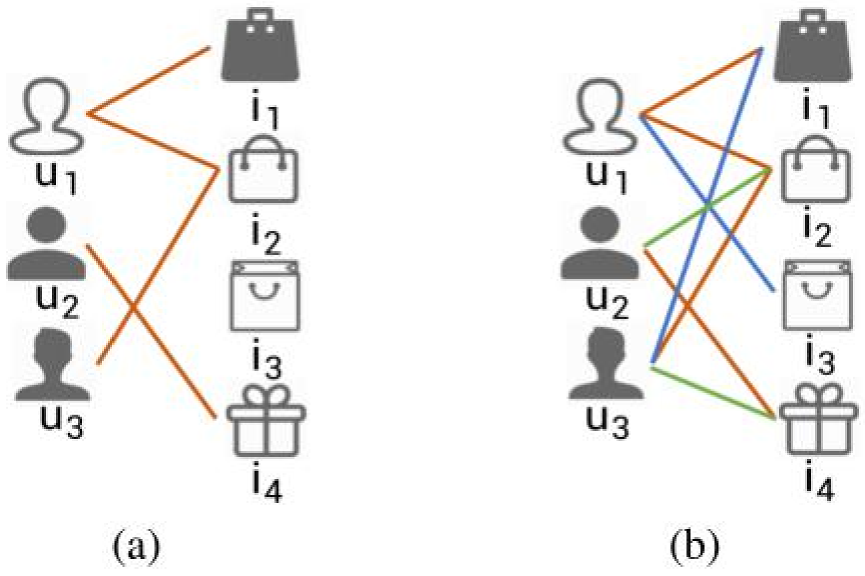


Figure 1: Examples of single-behavior and multi-behavior in e-commerce scene. (a) is *single-behavior* and (b) is *multi-behavior*. The **red** line indicates *purchase* behavior, the **blue** line indicates *click* behavior, and the **green** line indicates *add to cart* behavior.

Method

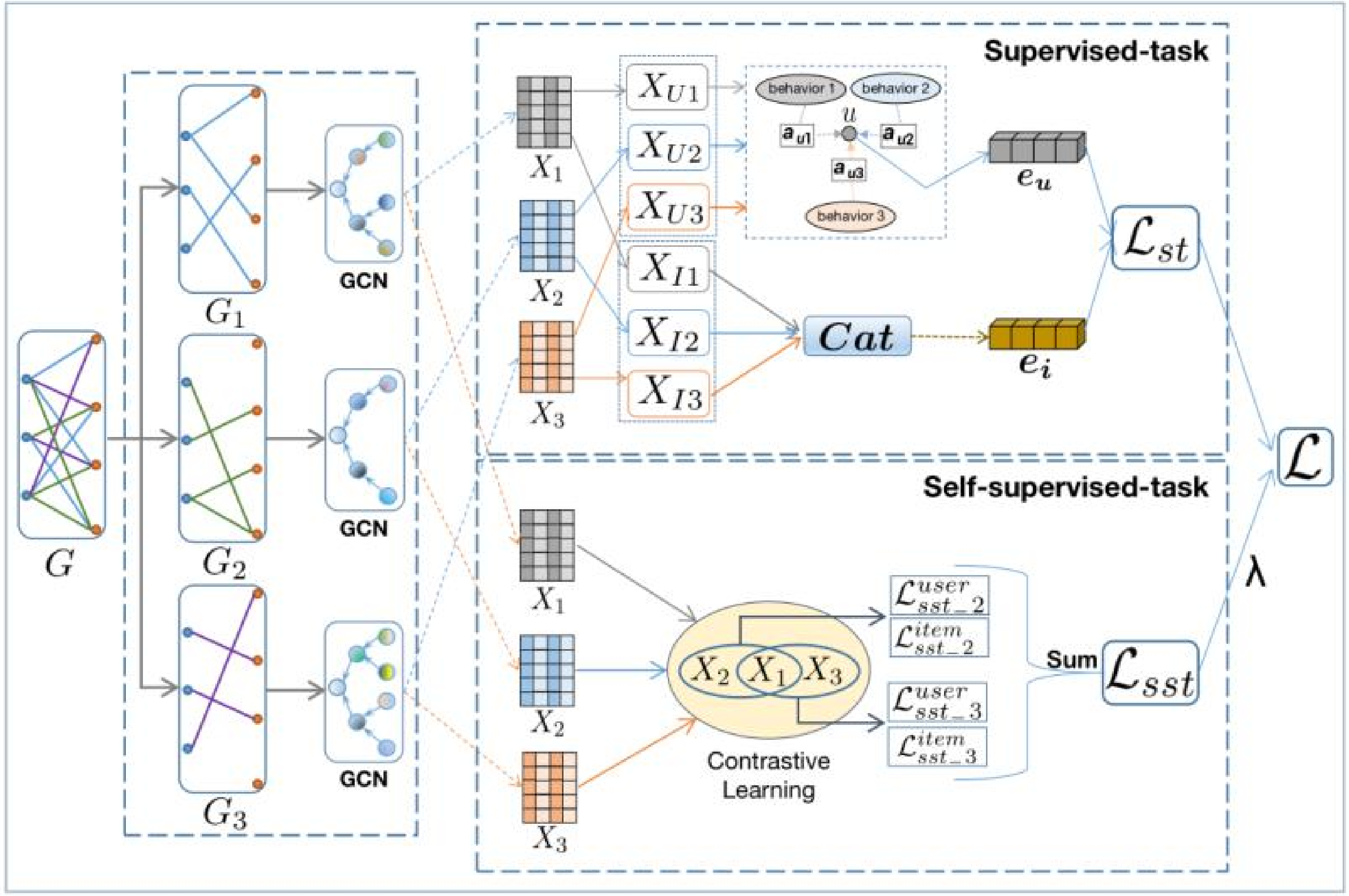


Figure 2: The model architecture of S-MBRec. (We take an example that $K = 3$, i.e., there are three kinds of behavior, in which the first is target behavior and the other two are auxiliary behaviors.)

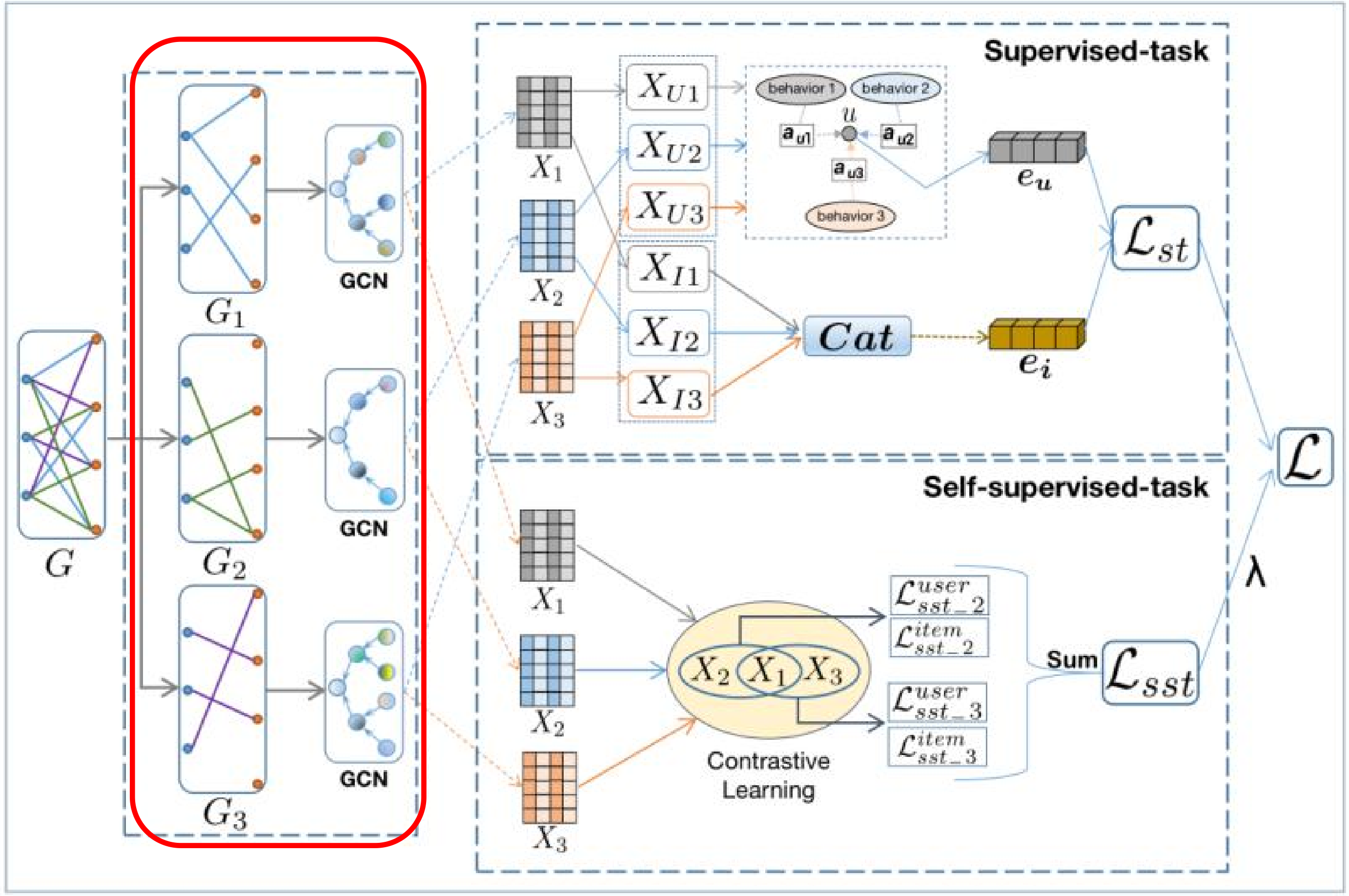


Figure 2: The model architecture of S-MBRec. (We take an example that $K = 3$, i.e., there are three kinds of behavior, in which the first is target behavior and the other two are auxiliary behaviors.)

$$\boldsymbol{A}_k = \begin{pmatrix} 0 & \boldsymbol{R}_k \\ \boldsymbol{R}_k^T & 0 \end{pmatrix}, \tag{1}$$

$$\widehat{\boldsymbol{A}_k} = \boldsymbol{D}_k^{-\frac{1}{2}}(\boldsymbol{A}_k + \boldsymbol{I}_k)\boldsymbol{D}_k^{-\frac{1}{2}}$$

$$\boldsymbol{X}_k^{(l+1)} = \sigma(\widehat{\boldsymbol{A}_k}\boldsymbol{X}_k^{(l)}\boldsymbol{W}_k), \tag{2}$$

$$\boldsymbol{X}_k = f(\boldsymbol{X}_k^{(l)}), \tag{3}$$

where $l = [0, ..., L]$. $\boldsymbol{X}_k$ consists of user embedding matrix $\boldsymbol{X}_{Uk} \in \boldsymbol{R}^{|U|*d}$ and item embedding matrix $\boldsymbol{X}_{Ik} \in \boldsymbol{R}^{|I|*d}$. The common designs of f are last layer only [Ying *et al.*, 2018], concatenation [Wang *et al.*, 2019], and weighted sum [He *et al.*, 2020], and we choose the concatenation operation in this paper.

Method

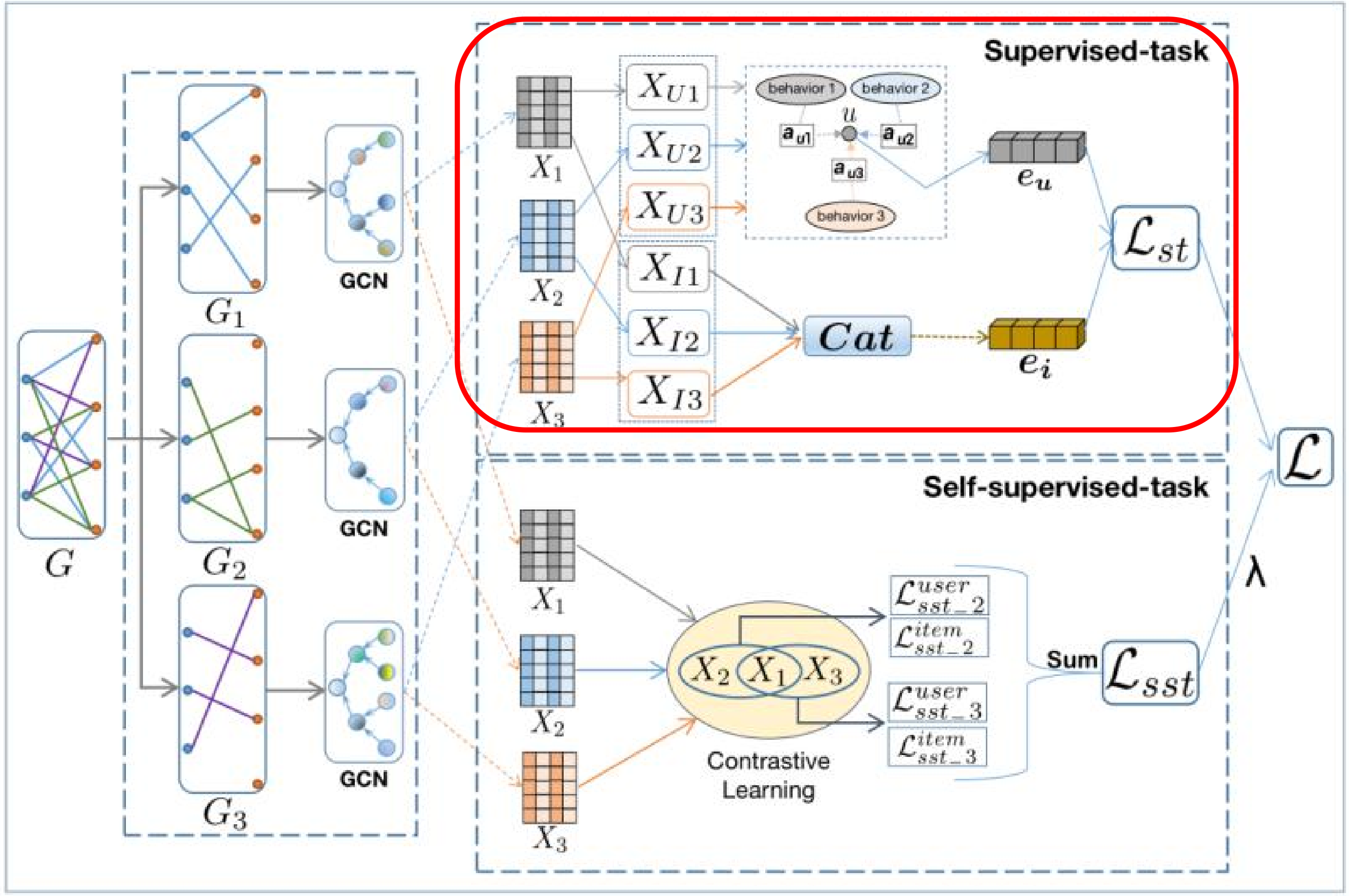


Figure 2: The model architecture of S-MBRec. (We take an example that $K = 3$, i.e., there are three kinds of behavior, in which the first is target behavior and the other two are auxiliary behaviors.)

$$a_{uk} = \frac{exp(w_k * n_{uk})}{\sum_{m=1}^{K} exp(w_m * n_{um})}, \tag{4}$$

$$\boldsymbol{e_u} = \sigma \left\{ \boldsymbol{W}\left(\sum_{m=1}^{K} a_{uk} * \boldsymbol{x_{uk}}\right) + \boldsymbol{b} \right\}, \tag{5}$$

$$\boldsymbol{e_i} = g\left\{Cat(\boldsymbol{x_{ik}})\right\}, \tag{6}$$

where $k = [1, ..., K]$, g is a Multi-Layer Perceptron (MLP), and Cat denotes the concatenation operation between K vectors.

$$\mathcal{L}_{st} = \sum_{(u,i,j)\in O} -log\left\{\sigma(\boldsymbol{e_u^T e_i} - \boldsymbol{e_u^T e_j})\right\}, \tag{7}$$

where $O = \{(u,i,j)|(u,i) \in O_+, (u,j) \in O_-\}$ is training data, and O_+ is the observed interactions. $O_- = (U \times V) - O_+$, which represents all unobserved interactions.

Method

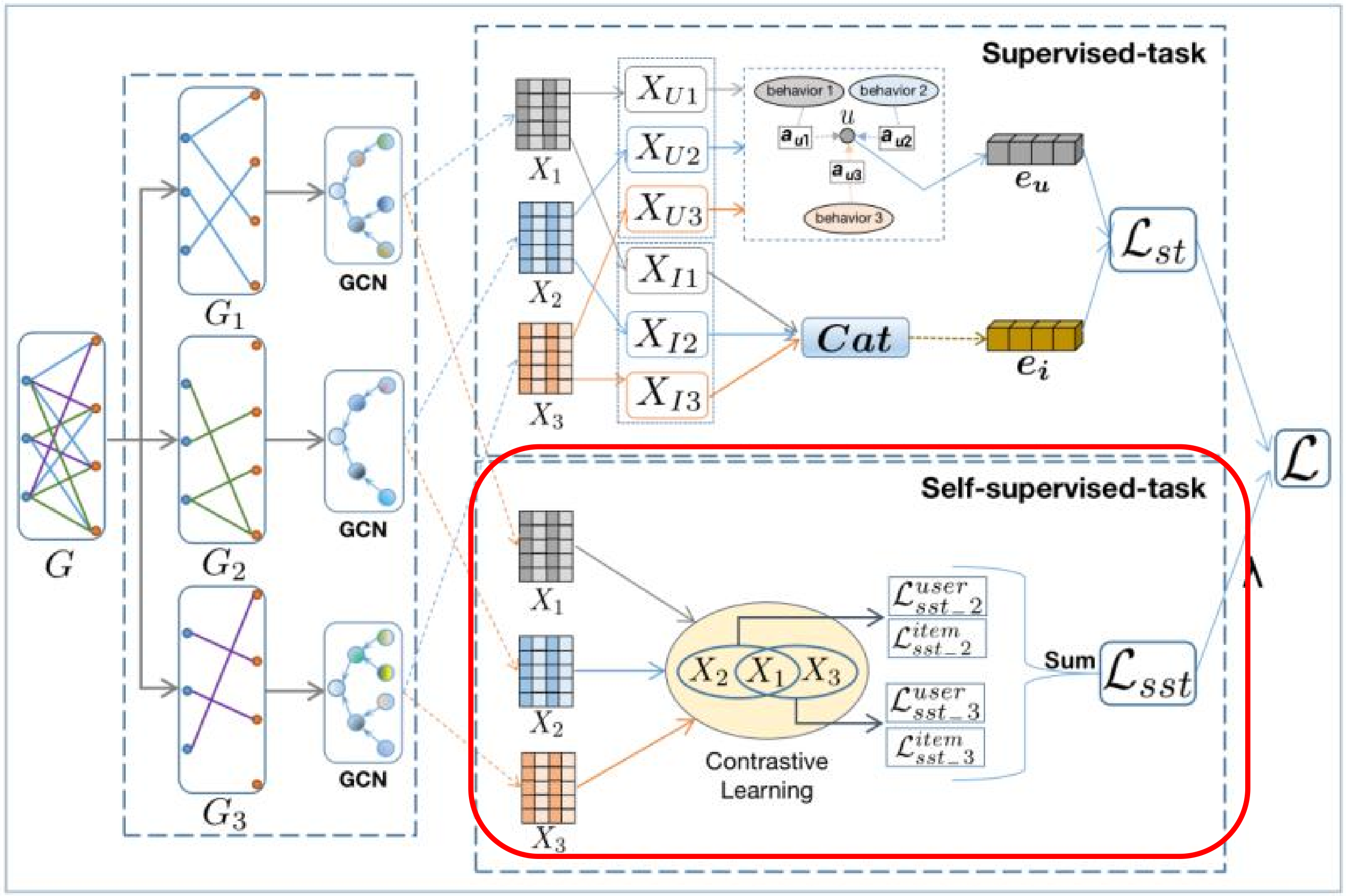


Figure 2: The model architecture of S-MBRec. (We take an example that $K = 3$, i.e., there are three kinds of behavior, in which the first is target behavior and the other two are auxiliary behaviors.)

$$PMI(u,u') = log\frac{p(u,u')}{p(u)p(u')}, \tag{8}$$

$$p(u) = \frac{|I(u)|}{|I|}, \tag{9}$$

$$p(u,u') = \frac{|I(u)\cap I(u')|}{|I|}, \tag{10}$$

$$\mathcal{L}_{sst\_k'}^{user} = \sum_{u\in U} -log\frac{\sum_{u^+\in U} exp\{(\boldsymbol{x_{uK}})^T \boldsymbol{x_{u+k'}}/\tau)\}}{\sum_{u^-\in U} exp\{(\boldsymbol{x_{uK}})^T \boldsymbol{x_{u-k'}}/\tau\}}, \tag{11}$$

where $(\boldsymbol{x_{uK}}, \boldsymbol{x_{u+k'}})$ is the the positive pair and $(\boldsymbol{x_{uK}}, \boldsymbol{x_{u-k'}})$ is the negative pair. τ is a hyper-parameter, known as the *temperature* coefficient in softmax. Analogously, we can obtain the contrastive loss $\mathcal{L}_{sst\_k'}^{item}$. By

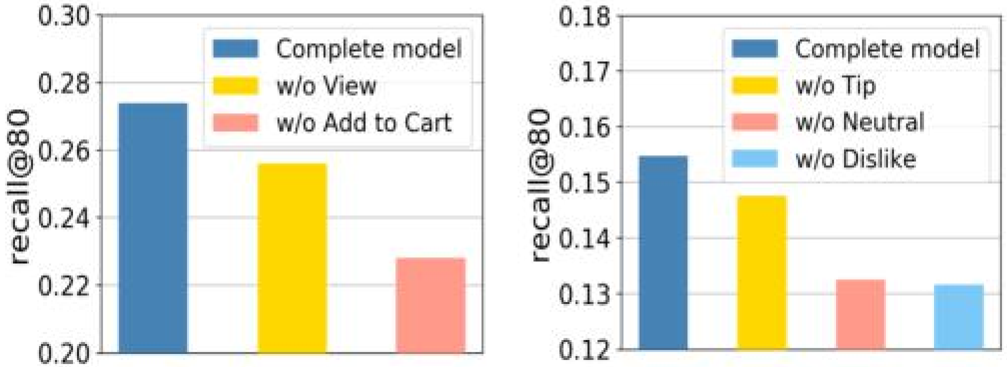
$$\mathcal{L}_{sst} = \sum_{k'=2}^{K} (\mathcal{L}_{sst\_k'}^{user} + \mathcal{L}_{sst\_k'}^{item}). \tag{12}$$

$$\mathcal{L} = \mathcal{L}_{st} + \lambda\mathcal{L}_{sst} + \mu \left\|\Theta\right\|_2^2, \tag{13}$$

		Single-Behavior Models				Multi-Behavior Models				Our Model
Dataset	**Metric**	**NCF**	**NGCF**	**ENMF**	**LightGCN**	**NMTR**	**EHCF**	**RGCN**	**MB-GMN**	**S-MBRec**
Beibei	**Recall@10**	0.0251	0.0389	0.0377	0.0452	0.0462	0.0459	0.0480	*0.0497*	**0.0529**
	Recall@40	0.0554	0.0754	0.0633	0.1211	0.1366	0.1271	0.1263	*0.1498*	**0.1647**
	Recall@80	0.0641	0.0933	0.0812	0.1939	0.1992	0.1923	0.1912	*0.2017*	**0.2740**
	NDCG@10	0.0117	0.0121	0.0109	0.0127	0.0129	0.0134	0.0123	*0.0139*	**0.0148**
	NDCG@40	0.0164	0.0154	0.0171	0.0187	0.0193	0.0214	0.0226	*0.0397*	**0.0429**
	NDCG@80	0.0228	0.0206	0.0312	0.0334	0.0423	0.0439	0.0443	*0.0465*	**0.0615**
Taobao	**Recall@10**	0.0141	0.0219	0.0198	0.3177	0.0369	0.0295	0.0372	*0.0438*	**0.0608**
	Recall@40	0.0204	0.0297	0.0224	0.0405	0.0487	0.0599	0.0706	*0.0873*	**0.1027**
	Recall@80	0.0311	0.0763	0.0459	0.0795	0.0983	0.1030	0.1527	*0.1559*	**0.1647**
	NDCG@10	0.0094	0.0105	0.0129	0.0216	0.0237	0.0284	0.0214	*0.0326*	**0.0391**
	NDCG@40	0.0141	0.0162	0.0226	0.0287	0.0305	0.0374	0.0304	*0.0398*	**0.0464**
	NDCG@80	0.0196	0.0206	0.0248	0.0265	0.0336	0.0390	0.0448	*0.0476*	**0.0583**
Yelp	**Recall@10**	0.0114	0.0175	0.0163	0.0148	0.0197	0.0186	0.0205	*0.0243*	**0.0259**
	Recall@40	0.0375	0.0398	0.0407	0.0676	0.0724	0.0705	0.0843	*0.0879*	**0.1135**
	Recall@80	0.0498	0.0604	0.0535	0.0823	0.0634	0.0980	0.1090	*0.1398*	**0.1548**
	NDCG@10	0.0044	0.0095	0.0102	0.0178	0.0190	0.0164	0.0214	*0.0273*	**0.0287**
	NDCG@40	0.0141	0.0162	0.0126	0.0187	0.0305	0.0294	0.0204	*0.0248*	**0.0337**
	NDCG@80	0.0164	0.0216	0.0227	0.0235	0.0354	0.0342	0.0398	*0.0416*	**0.0438**

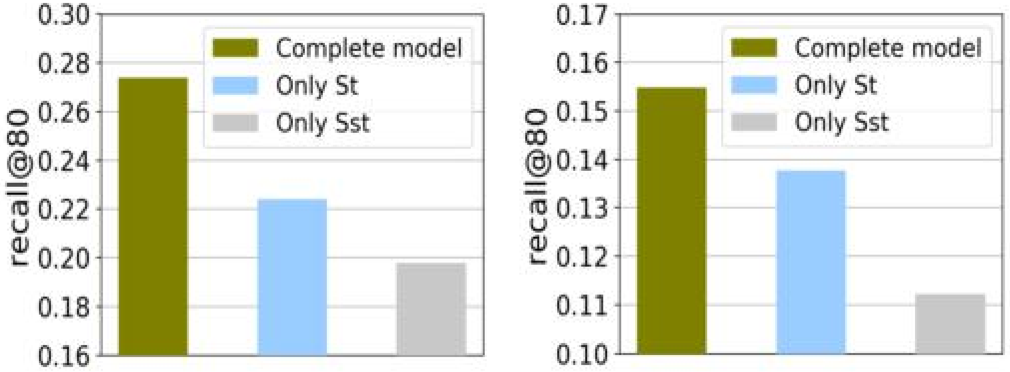
Table 2: Overall model performance on Beibei, Taobao and Yelp datasets, with the metrics of Recall@K and NDCG@K (K=10, 40, 80).

Experiment



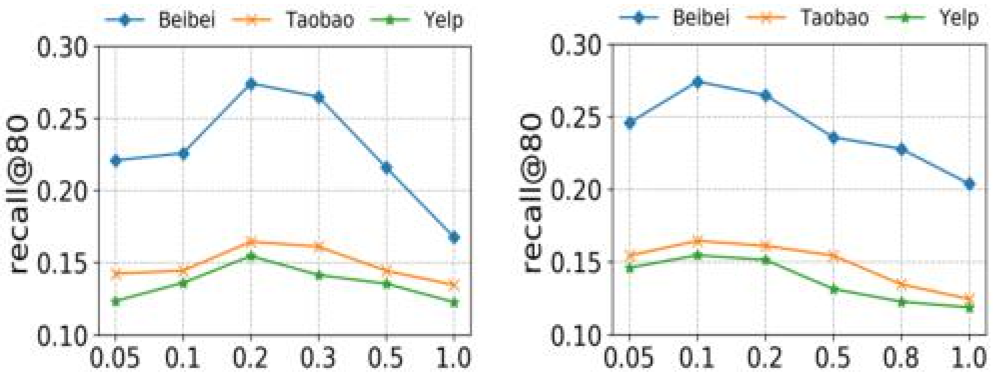
(a) Beibei (b) Yelp

Figure 3: The result comparison of removing different auxiliary behaviors. (Take Beibei and Yelp datasets as examples, and the evaluation index is $Recall@80$.)



(a) Beibei (b) Yelp

Figure 4: The result comparison of removing different tasks. (Take Beibei and Yelp datasets as examples, and the evaluation index is $Recall@80$. St represents supervised task. Sst represents self-supervised task.)



(a) λ (b) τ

Figure 5: Impact of λ and τ to our model under three datasets.

Thanks