Robust Preference-Guided Denoising for Graph based Social Recommendation

Yuhan Quan
Department of Electronic Engineering, Tsinghua University
Beijing, China

Jingtao Ding*
Department of Electronic Engineering, Tsinghua University
Beijing, China

Chen Gao
Department of Electronic Engineering, Tsinghua University
Beijing, China

Lingling Yi
Wechat, Tencent
Shenzhen, China

Depeng Jin
Department of Electronic Engineering, Tsinghua University
Beijing, China

Yong Li
Department of Electronic Engineering, Tsinghua University
Beijing, China

（WWW-2023)

code： https://github.com/tsinghua-fib-lab/Graph-Denoising-SocialRec.

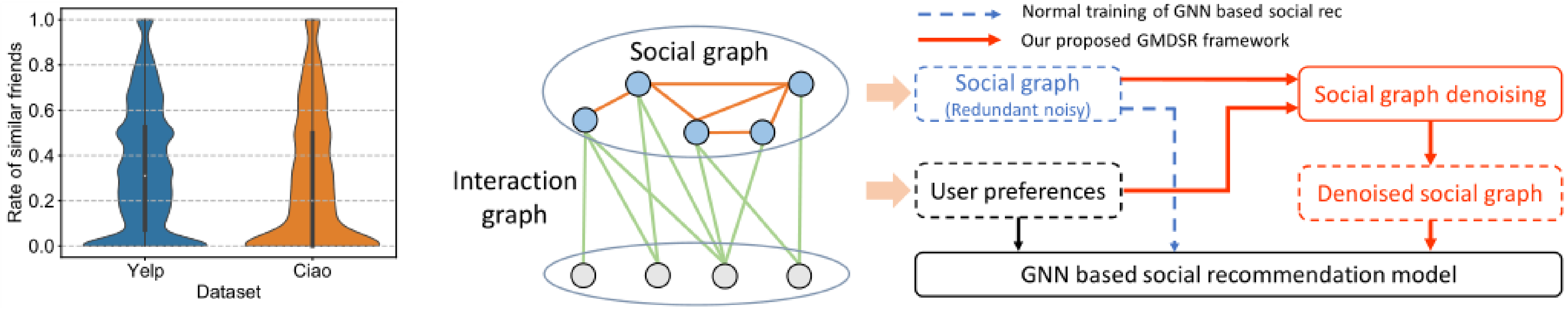
gesis Leibniz-Institut für Sozialwissenschaften

Forschungszentrum · Research Center L3S

Reported by Zhaoze Gao

1. Introduction
2. Approach
3. Experiments

Introduction



(a) Empirical data statistics

(b) Proposed framework

Figure 1: (a) Distribution plot *w.r.t.* ratio of friends having co-interactions. (b) Our proposed denoising enhanced social recommendation framework.

Approach

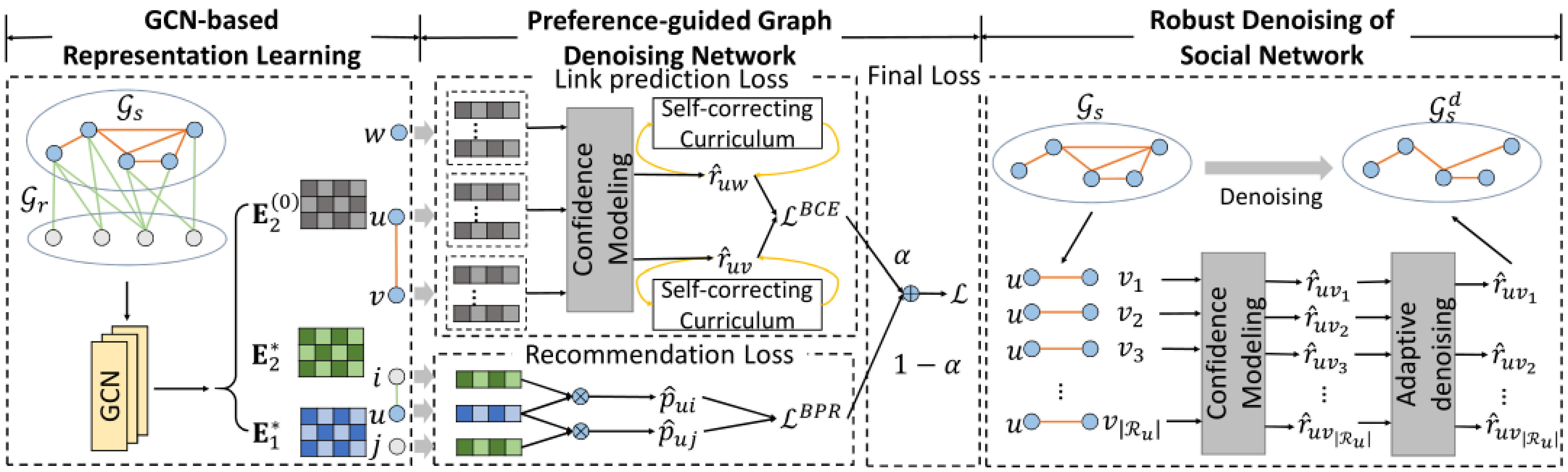


Figure 2: Details of graph denoising process in GDMSR.

Approach

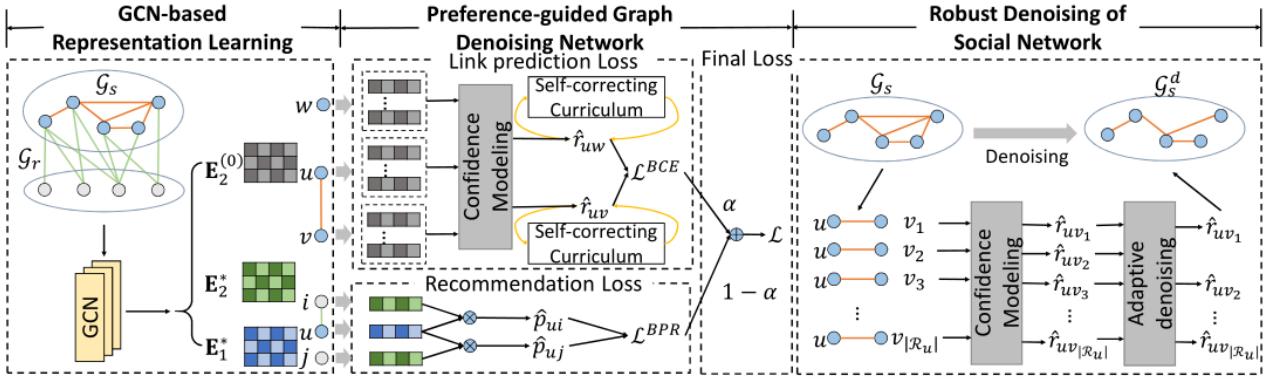


Figure 2: Details of graph denoising process in GDMSR.

$$\mathbf{E}_{1,s}^{(K)}(u) = \text{GNN}\left(\mathbf{E}_1^{(K-1)}(u), \left\{\mathbf{E}_1^{(K-1)}(v) | \forall v \in \mathcal{R}_u\right\}\right), \quad (1)$$

$$\mathbf{E}_{1,r}^{(K)}(u) = \text{GNN}\left(\mathbf{E}_1^{(K-1)}(u), \left\{\mathbf{E}_2^{(K-1)}(i) | \forall i \in \mathcal{P}_u\right\}\right), \quad (2)$$

$$\mathbf{E}_1^{(K)}(u) = \text{Combine}\left(\mathbf{E}_{1,s}^{(K)}(u), \mathbf{E}_{1,r}^{(K)}(u)\right), \quad (3)$$

$$\mathbf{E}_2^{(K)}(i) = \text{GNN}\left(\mathbf{E}_2^{(K-1)}(i), \left\{\mathbf{E}_1^{(K-1)}(u) | \forall u \in \mathcal{P}_i\right\}\right), \quad (4)$$

$$\hat{p}_{ui} = \mathbf{E}_1^*(u) \cdot \mathbf{E}_2^*(i),$$
$$\text{where } \mathbf{E}_1^*(u) = \frac{\sum_{k=0}^K \mathbf{E}_1^{(k)}(u)}{K+1}, \mathbf{E}_2^*(i) = \frac{\sum_{k=0}^K \mathbf{E}_2^{(k)}(i)}{K+1}. \quad (5)$$

$$\mathcal{L}^{BPR} = \sum_{(u,i,j)\in\overline{\mathcal{P}}} -\ln \sigma(\hat{p}_{ui} - \hat{p}_{uj}), \quad (6)$$

$\mathcal{G}_s(\mathcal{U}, \mathcal{E})$ $\mathcal{U} = \{u\}$ and $\mathcal{R} = \{(u,v) | r_{uv} = 1, \forall u, v \in \mathcal{U}\}$

$\mathcal{G}_r(\{\mathcal{U} \cup \mathcal{I}\}, \mathcal{P})$ $\mathbf{E}_1 \in \mathbb{R}^{N\times D}$ and $\mathbf{E}_2 \in \mathbb{R}^{M\times D}$

$\mathbf{E}_{1,s}$ and $\mathbf{E}_{1,r}$,

Approach

$$\mathcal{L}^{BCE} = -\sum_{(u,v)\in\mathcal{R}} \log(\sigma(\hat{r}_{uv})) - \sum_{(u,w)\notin\mathcal{R}} \log(1-\sigma(\hat{r}_{uw})). \quad (7)$$

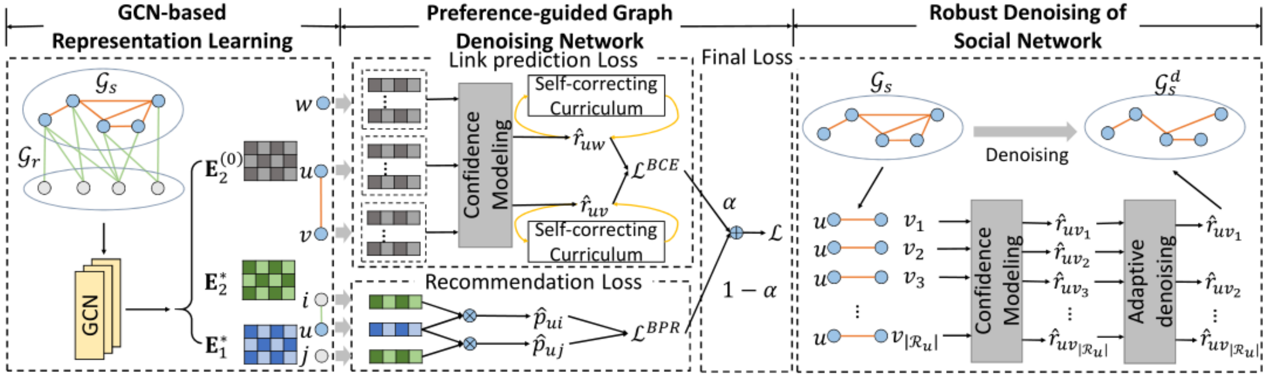


Figure 2: Details of graph denoising process in GDMSR.

$$\hat{r}_{uv} = \phi\left(\{\mathbf{E}_1^{(k)}(u)\}_{k=0}^{K}, \{\mathbf{E}_1^{(k)}(v)\}_{k=0}^{K}\right), \quad (8)$$

$$\hat{r}_{uv} = \mathrm{Trf}\left(S_L\left(\left\{\mathbf{E}_2^{(0)}(i)|\forall i \in \mathcal{P}_u\right\}\right) \oplus S_L\left(\left\{\mathbf{E}_2^{(0)}(j)|\forall j \in \mathcal{P}_v\right\}\right)\right), \quad (9)$$

$$\mathcal{L} = \alpha\mathcal{L}^{BCE} + (1-\alpha)\mathcal{L}^{BPR}, \quad (10)$$

Approach

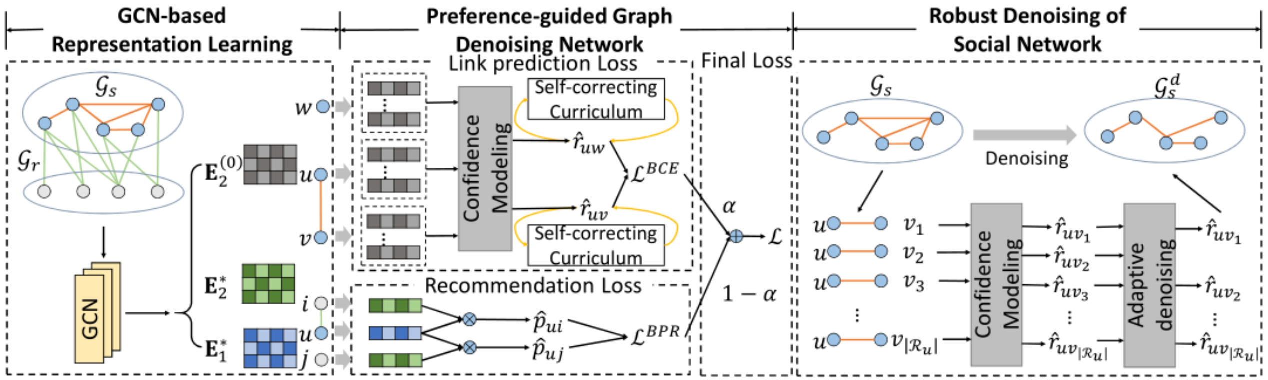


Figure 2: Details of graph denoising process in GDMSR.

$$\hat{r}_{uv}(t = kD) = \beta \cdot \hat{r}_{uv}(t = (k-1)D) + (1-\beta) \cdot \hat{r}_{uv}(t = kD), \quad (11)$$

$$\eta_u = \begin{cases} 0, & \text{if } |\mathcal{R}_u| < \epsilon, \\ \left[\lfloor \log_{10}(|\mathcal{R}_u|) \rfloor\right]^{\gamma} \times R, & \text{else,} \end{cases} \quad (12)$$

Experiments

Table 1: Basic information of datasets.

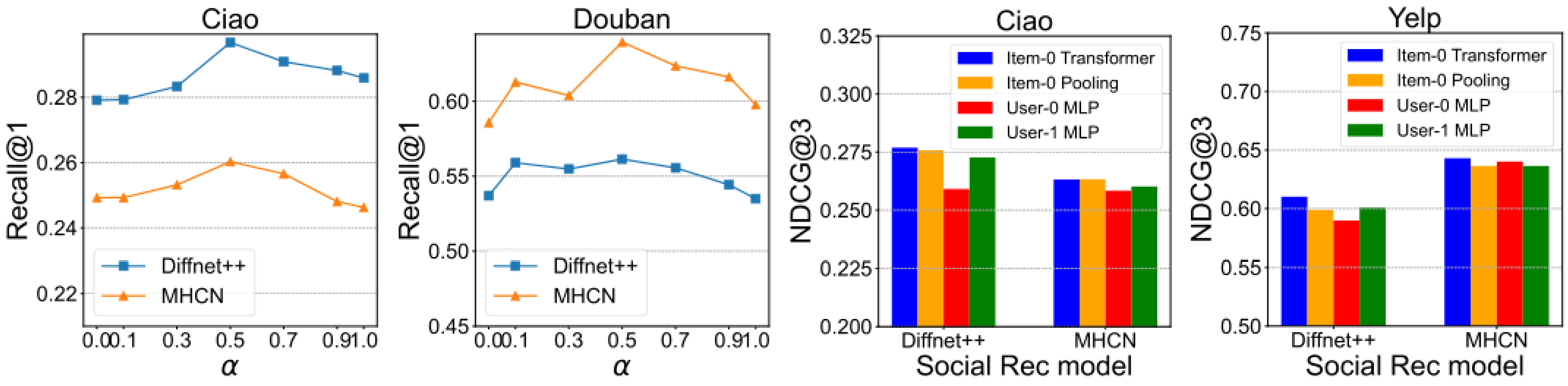
Dataset	#Users	#Items	#Interactions	#Relations	Interaction Density	Relation Density
Ciao	7,355	17,867	140,628	111,679	0.11%	0.21%
Yelp	32,827	59,972	598,121	964,510	0.03%	0.09%
Douban	2,669	15,940	535,210	32,705	1.26%	0.46%

Experiments

Table 2: Overall performance of our proposed method on different recommendation methods.

Dataset		Ciao			Yelp			Douban		
Basemodel	Method	R@1	R@3	N@3	R@1	R@3	N@3	R@1	R@3	N@3
LightGCN	-	0.2298	0.0785	0.2071	0.5861	0.2774	0.5804	0.4321	0.1696	0.4156
Diffnet++	w/o denoising	0.2742	0.1109	0.2639	0.6031	0.3072	0.5897	0.5165	0.2156	0.4988
	Rule based	0.2860	0.1123	0.2677	0.6230	0.3228	0.5996	0.5358	0.2489	0.5172
	NeuralSparse	0.2869	0.1153	0.2734	0.6383	0.3289	0.6054	0.5470	0.2226	0.5102
	ESRF	0.2864	0.1197	0.2736	0.6184	0.3124	0.5958	0.5374	0.2393	0.5194
	GDMSR	**0.3020**	**0.1244**	**0.2821**	**0.6449**	**0.3291**	**0.6102**	**0.5614**	**0.2540**	**0.5297**
	Δ	5.26%	3.93%	3.11%	1.03%	0.06%	0.79%	2.63%	2.05%	1.98%
MHCN	w/o denoising	0.2330	0.0884	0.2297	0.6991	0.3252	0.6364	0.6198	0.3167	0.5933
	Rule based	0.2301	0.0916	0.2311	0.6966	0.3234	0.6347	0.6082	0.3372	0.5980
	NeuralSparse	0.2461	0.1034	0.2540	0.7012	0.3288	0.6352	0.6206	0.3349	0.6011
	ESRF	0.2495	0.1028	0.2568	0.6927	0.3298	0.6344	0.6194	0.3244	0.5995
	GDMSR	**0.2618**	**0.1138**	**0.2632**	**0.7036**	**0.3405**	**0.6434**	**0.6396**	**0.3496**	**0.6137**
	Δ	4.93%	10.06%	2.50%	0.34%	3.24%	1.10%	3.06%	3.68%	2.10%

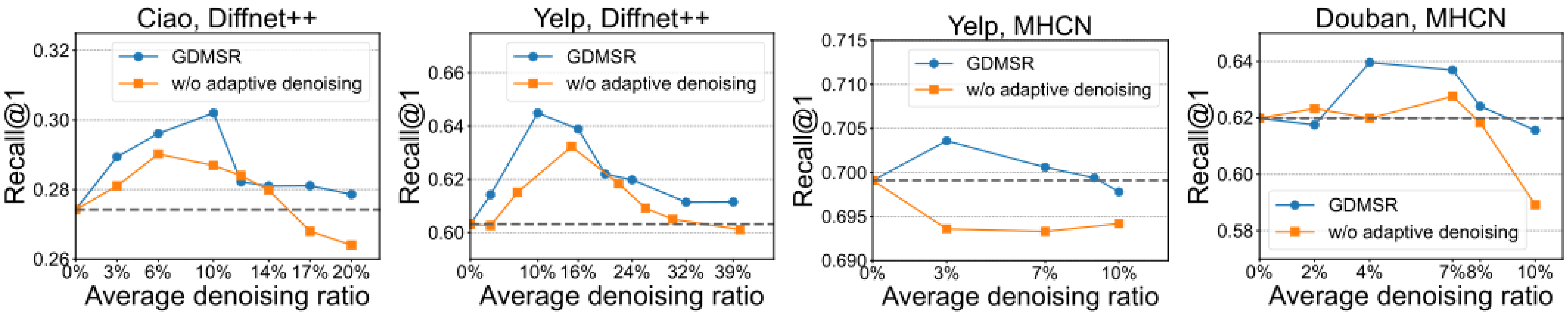
Experiments



(a) co-optimization weight (b) co-optimization weight (c) relation confidence modeling (d) relation confidence modeling

Figure 3: (a)-(b) Performance on different α for co-optimization. (c)-(d) Performance comparison among different relation confidence modeling structures.

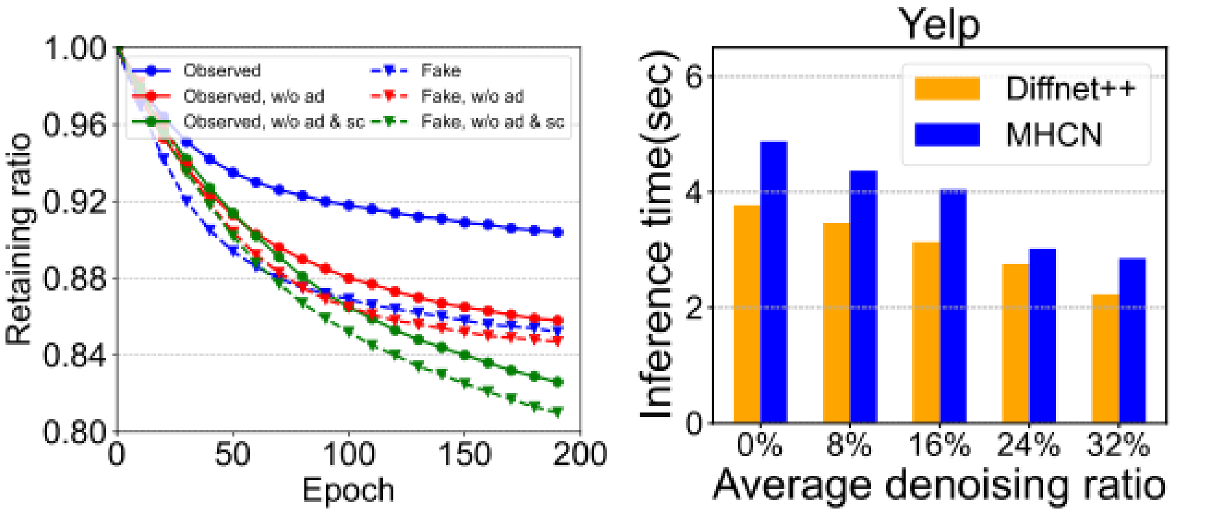
Experiments



(a) (b) (c) (d)

Figure 4: Analysis of denoising robustness with respect to recommendation accuracy under different denoising ratio.

Experiments



(a) denoising capability (b) inference efficiency

Figure 5: (a) Retaining ratio of social relations adopting different denoising strategies (synthetic). (b) Efficiency comparison *w.r.t.* social graph related inference time (Yelp).

Experiments

Table 3: Overall performance of GDMSR based on 30% of interaction data.

Dataset		Ciao			Yelp			Douban		
Basemodel	Method	R@1	R@3	N@3	R@1	R@3	N@3	R@1	R@3	N@3
Diffnet++	w/o denoising	0.2742	0.1109	0.2639	0.6031	0.3072	0.5897	0.5165	0.2156	0.4988
	GDMSR	**0.2899**	**0.1175**	**0.2755**	**0.6305**	**0.3189**	**0.6021**	**0.5490**	**0.2497**	**0.5295**
	Δ	5.73%	5.95%	4.40%	4.54%	3.81%	2.10%	6.29%	15.82%	6.15%
MHCN	w/o denoising	0.2330	0.0884	0.2297	0.6991	0.3252	0.6364	**0.6198**	0.3167	0.5933
	GDMSR	**0.2616**	**0.1093**	**0.2638**	**0.7005**	**0.3438**	**0.6445**	0.6148	**0.3360**	**0.6012**
	Δ	12.27%	23.64%	14.85%	0.20%	5.72%	1.27%	-0.81%	6.09%	1.33%

Experiments

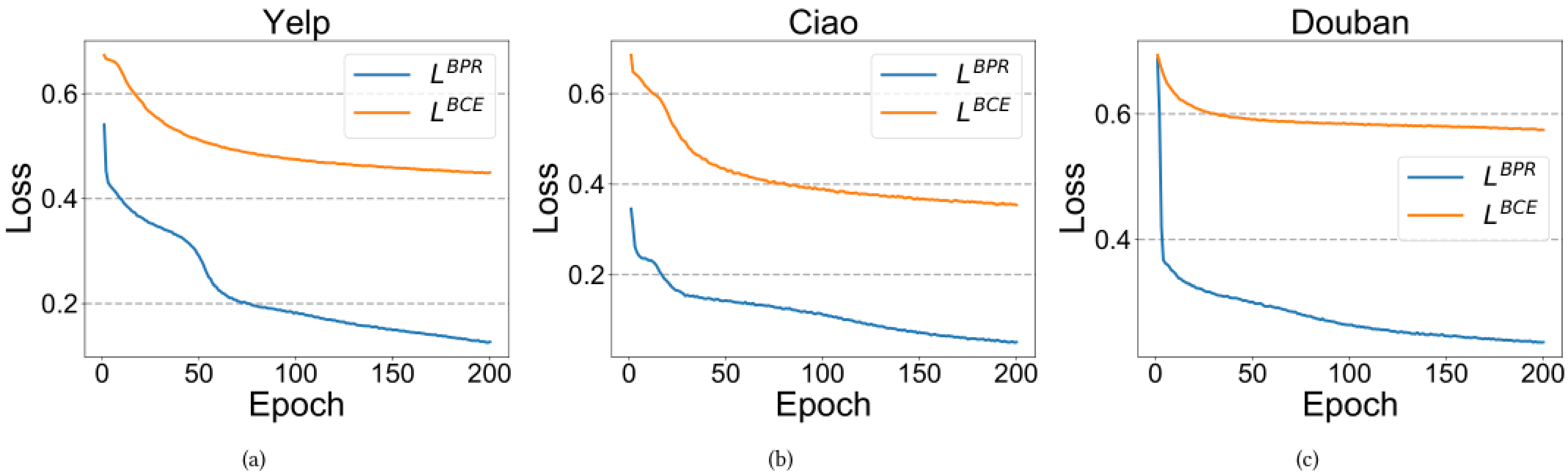
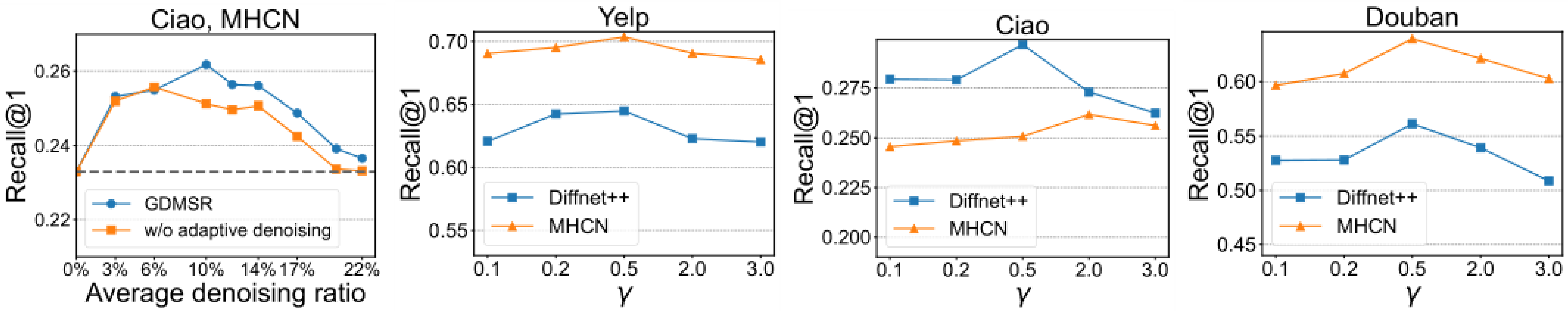


Figure 6: Loss curve of denoising training

Experiments



(a) (b) (c) (d)

Figure 7: (a)Performance comparison between different denoising strategy. (b)Performance on different γ for adaptive denoising

Experiments

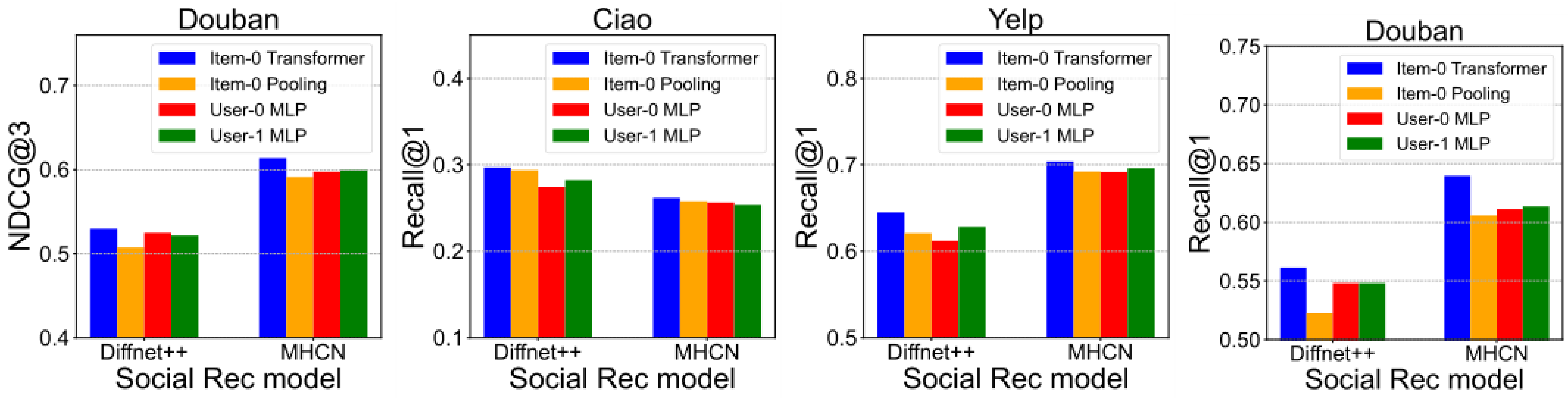


Figure 8: Performance comparison among different relation confidence modeling structures.

Thank you！