

图神经网络在推荐系统中的应用

报告人： 吴咏萱

报告时间： 2023. 4. 27



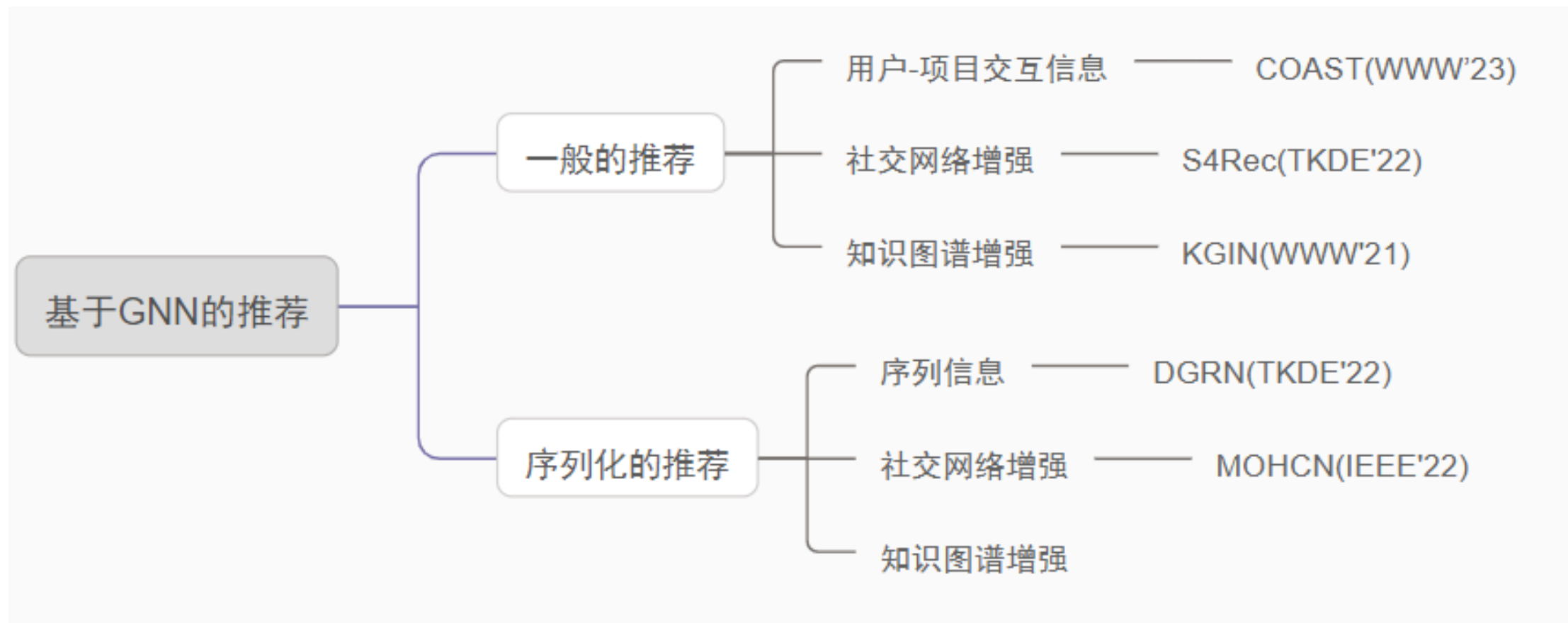
中国科学院 信息工程研究所
INSTITUTE OF INFORMATION ENGINEERING, CAS

背景

推荐系统主要的挑战是从历史交互和边信息中学习有效的用户和商品表示，基于图神经网络的推荐系统在许多方面都取得了最先进的水平。基于图神经网络的推荐系统的成功可以从以下三个角度来解释：

- 数据--GNN可以利用多种形式的的数据，同时在表示学习方面具有强大能力
- 高阶连通性--基于GNN的模型可以有效地捕获高阶连接性。
- 监督信号--监督信号在采集的数据中通常是稀疏的，而基于GNN的模型可以在表示学习过程中利用半监督信号来缓解这一问题

背景



背景

做GNN在推荐系统上的应用的国内外企业及高校

百度	清华大学
阿里巴巴	北京大学
腾讯	中科院计算所
华为	香港科技大学
字节跳动	新加坡国立大学
美团	斯坦福大学
网易	加州大学伯克利分校
亚马逊	
微软亚洲研究院	
IBM	
谷歌	
Facebook	

目录

1. 基于用户-项目交互的推荐

2. 基于序列信息的推荐

3. 社会网络增强的推荐

4. 知识图谱增强的推荐

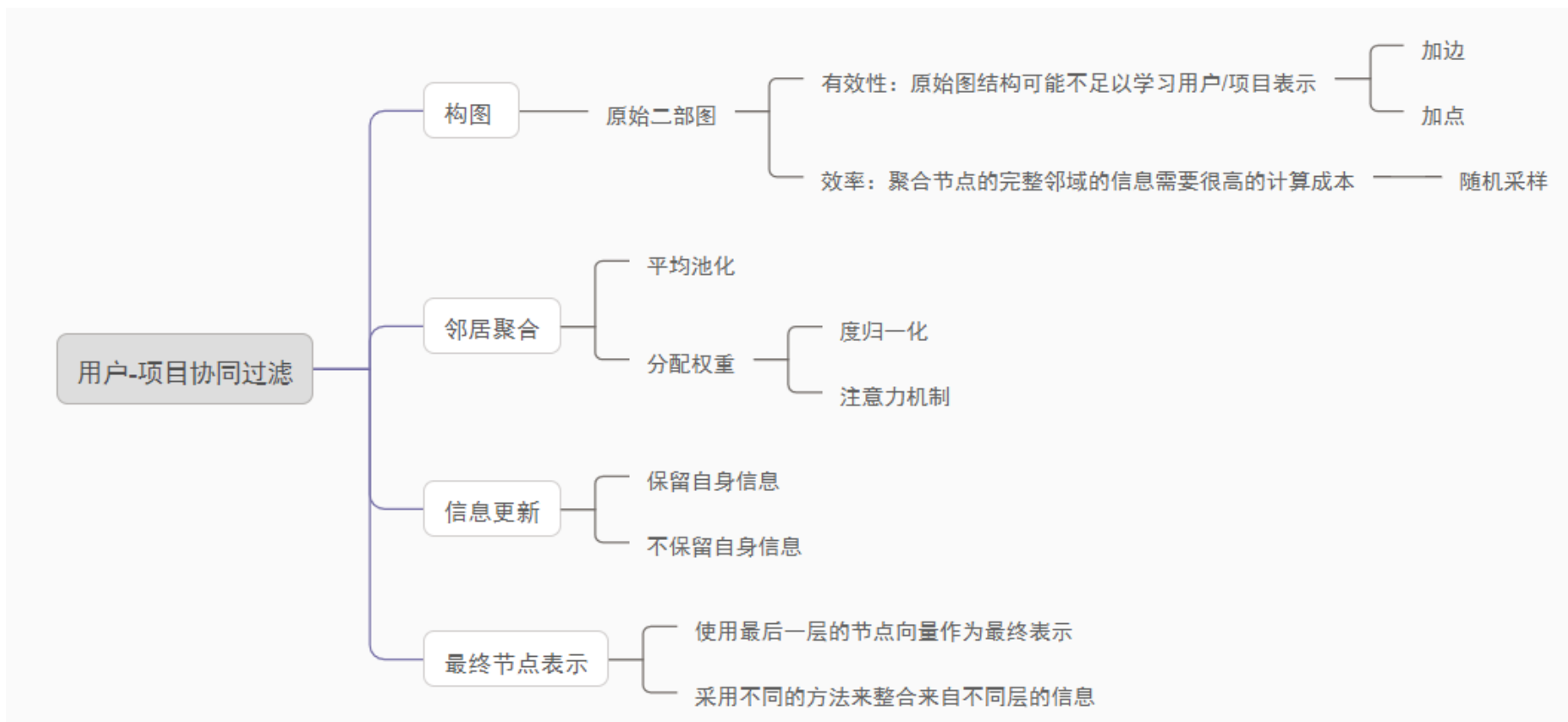
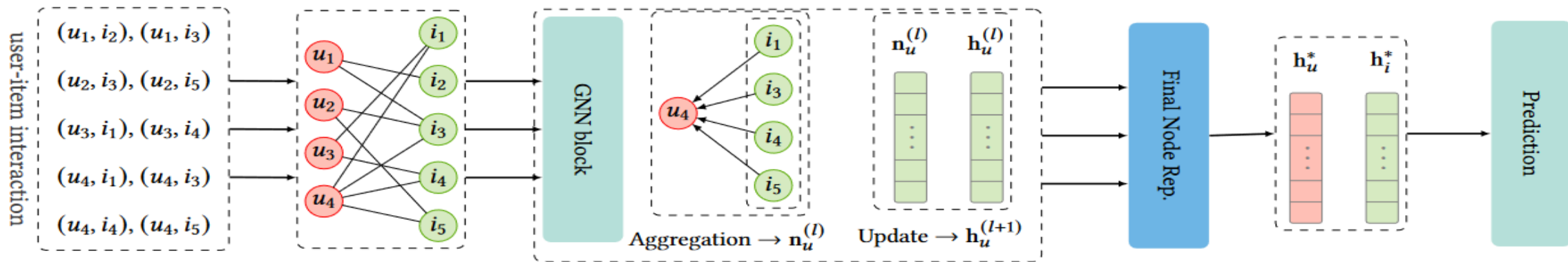
5. 总结

基于用户-项目交互的推荐

1.

基于用户-项目交互的推荐

基于用户-项目交互的推荐



Cross-domain recommendation via user interest alignment

Chuang Zhao

College of Management and
Economics, Tianjin University
Tianjin, China
zhaochuang@tju.edu.cn

Hongke Zhao

College of Management and
Economics, Tianjin University
Tianjin, China
hongke@tju.edu.cn

Ming HE

AI Lab at Lenovo Research
Beijin, China
heming01@foxmail.com

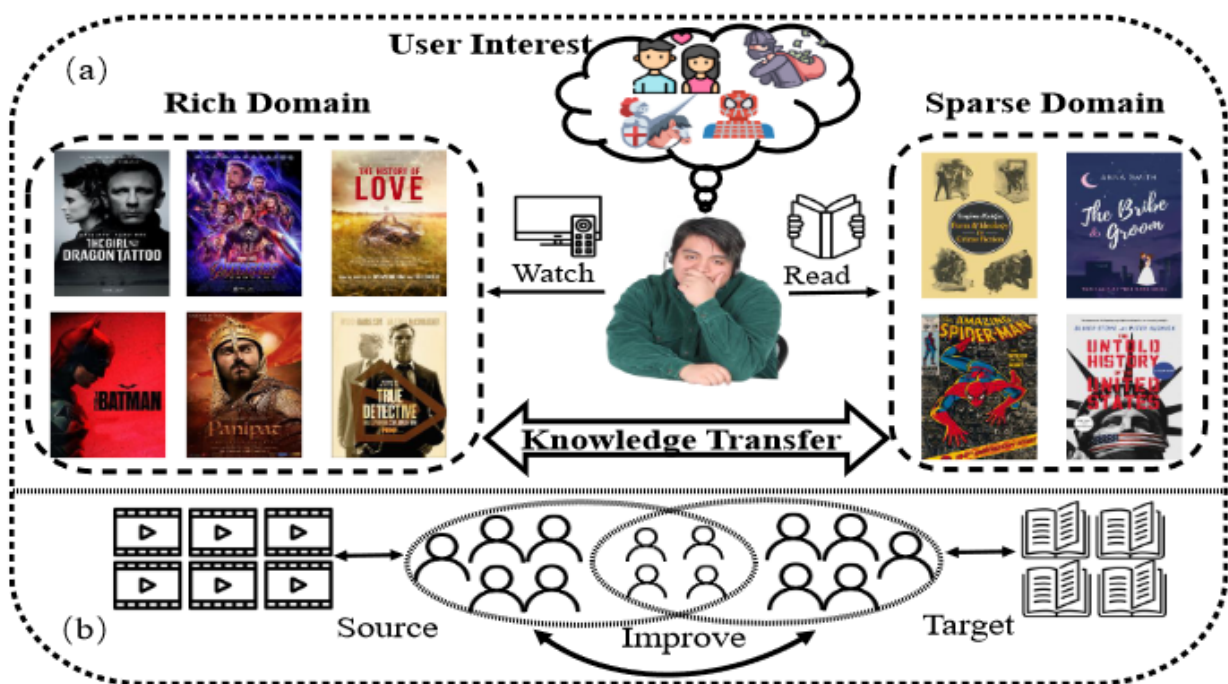
Jian Zhang

School of Cyberspace Security,
Hangzhou Dianzi University
Hangzhou, China
zhang.jian-94@outlook.com

Jianping Fan

AI Lab at Lenovo Research
Beijin, China
jfan1@lenovo.com

COAST(WWW' 23)



现有方法的不足

- 传统的跨域推荐方法的忽略了用户和项目之间潜在的跨域相似性。
- 忽略了用户之间的兴趣对齐，甚至违背了用户原始的兴趣分布。

本文贡献

- 构建了统一的跨域异构图，开发了一种新的消息传递机制来捕获实体之间的跨域相似性。
- 采用对比学习和梯度对齐来分别约束用户-用户和用户-项目兴趣对齐，从而增强跨域的兴趣一致性。

COAST(WWW' 23)

1. 构图: $D_S = (u_S, v_S, \mathcal{E}_S, x_S)$ $D_T = (u_T, v_T, \mathcal{E}_T, x_T)$

$$u_S = \{u_s, u_o\} \quad u_T = \{u_t, u_o\}$$

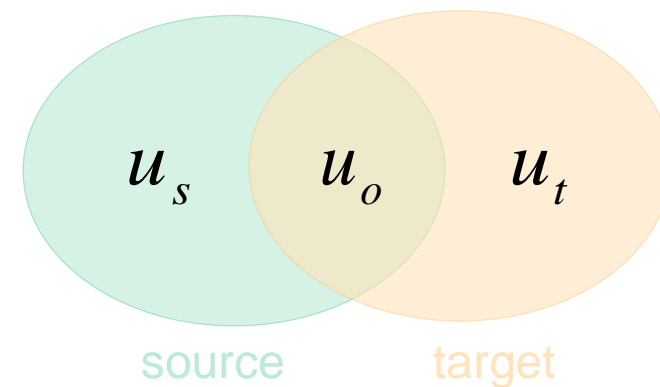
$$A_S = \{0,1\}^{|u_S| \times |v_S|} \quad A_T = \{0,1\}^{|u_T| \times |v_T|}$$

2. 跨域图卷积:

2.1 原始嵌入:

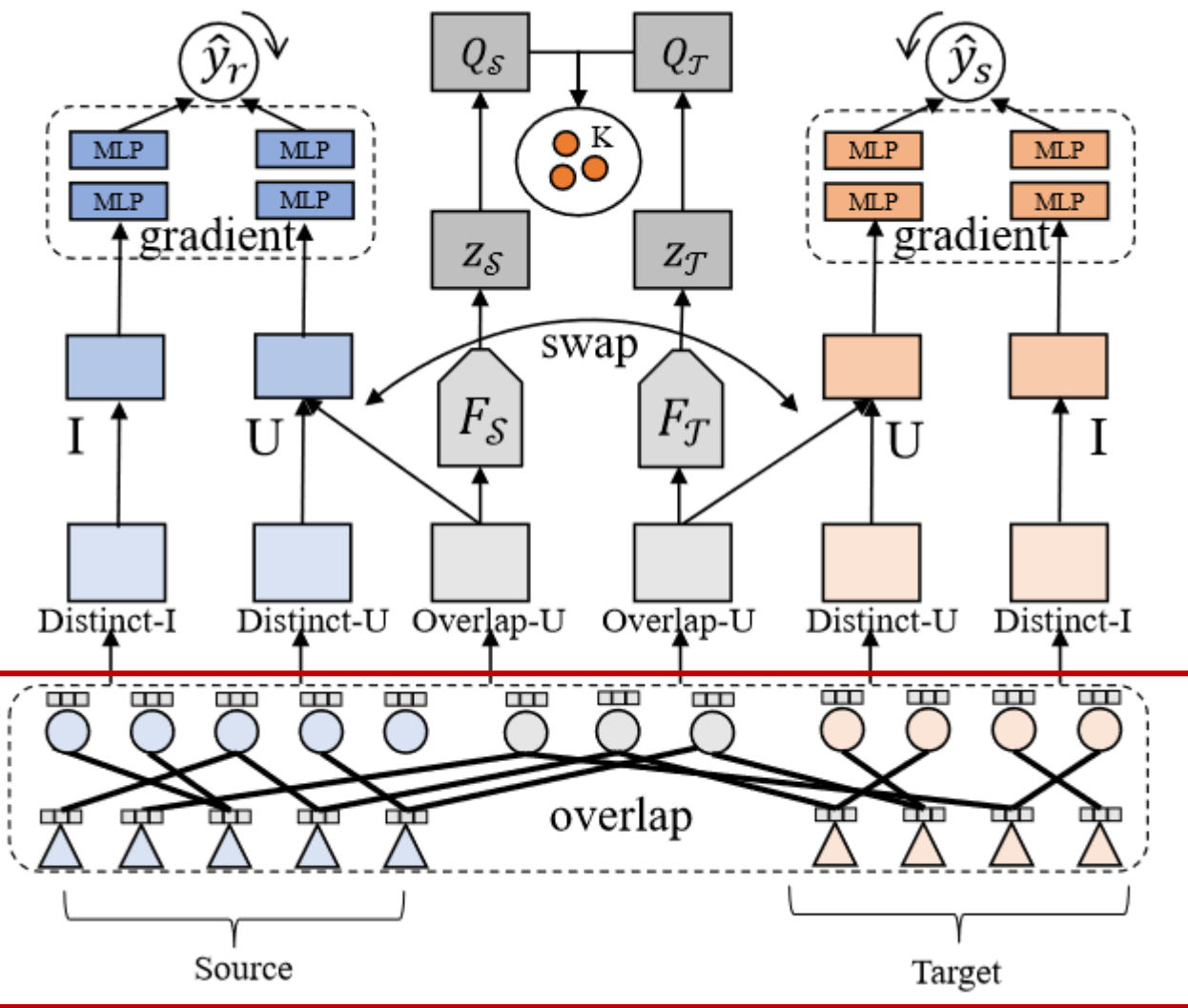
$$e^u = \begin{cases} e_S^u & \text{if } u \in \mathcal{U}_s \\ e_T^u & \text{if } u \in \mathcal{U}_t \\ e_S^u \otimes e_T^u & \text{if } u \in \mathcal{U}_o \end{cases}$$

最大池化



- 为不同领域的用户-项目交互形成了一个统一的图结构，直观且易于捕获跨领域相似性。

COAST(WWW' 23)



2.2 消息传递:

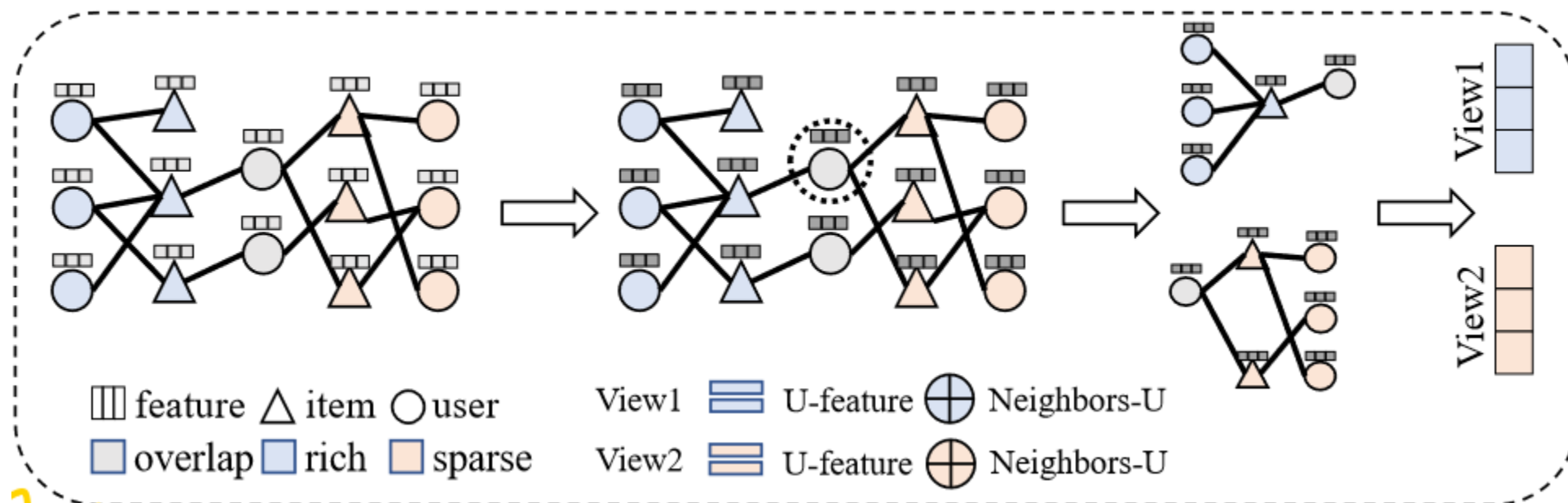
$$m_{u \leftarrow v} = \frac{1}{\sqrt{|N_u| |N_v^S| |N_v^T|}} (W_1 e^u + W_2 (e_v^S \odot e^u) + W_3 (e_v^T \odot e^u))$$

- 将消息传递机制推广到跨域场景，增强了传统GNN的实用性。

COAST(WWW' 23)

3. 用户兴趣对齐:

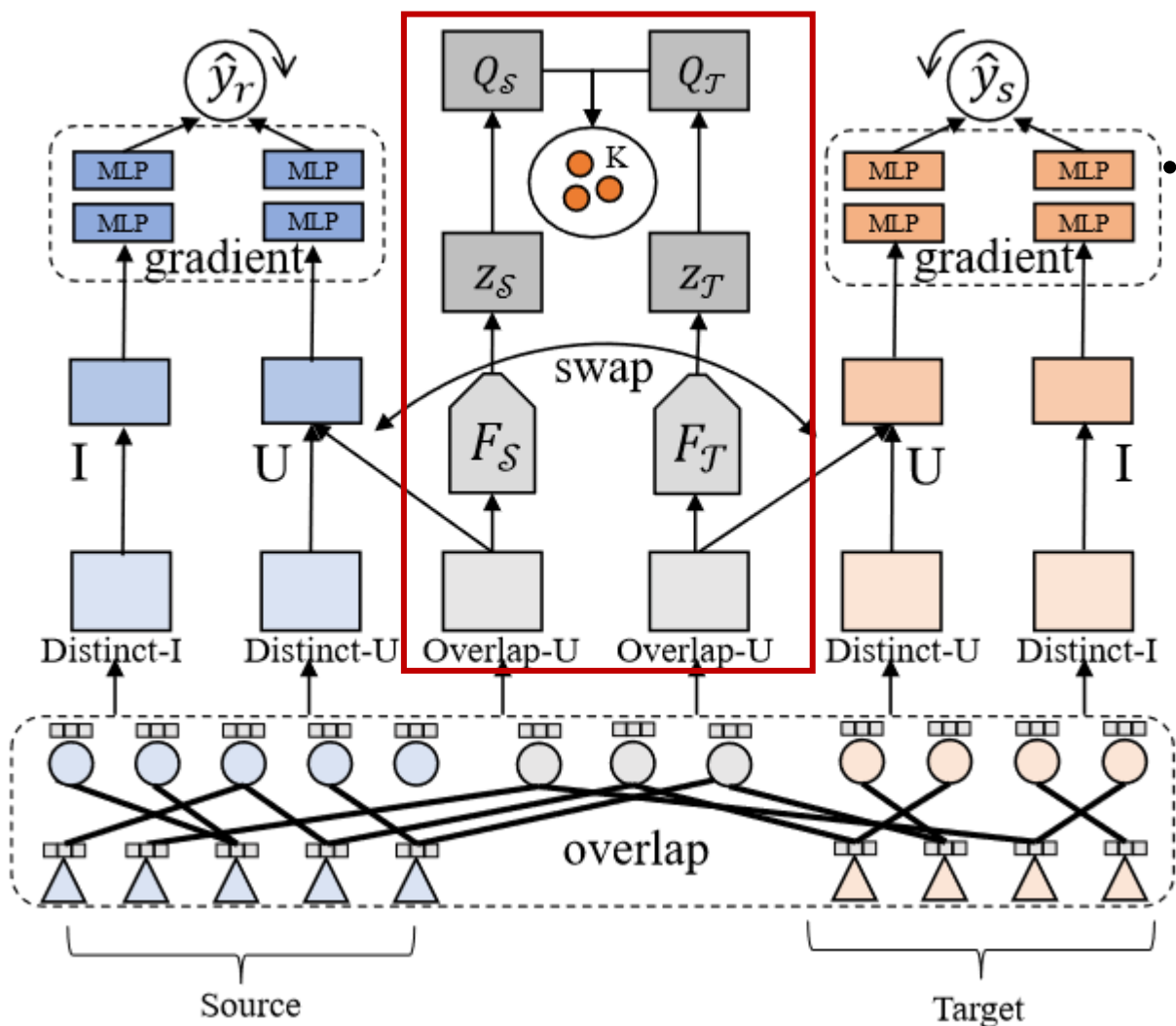
3.1. 用户-用户对齐:



- 分别聚合用户在不同领域的二阶邻居表示，得到相应的对比视图

COAST(WWW' 23)

3.1. 用户-用户对齐:



鼓励用户的对比view在兴趣分布上具有相同的聚类结果

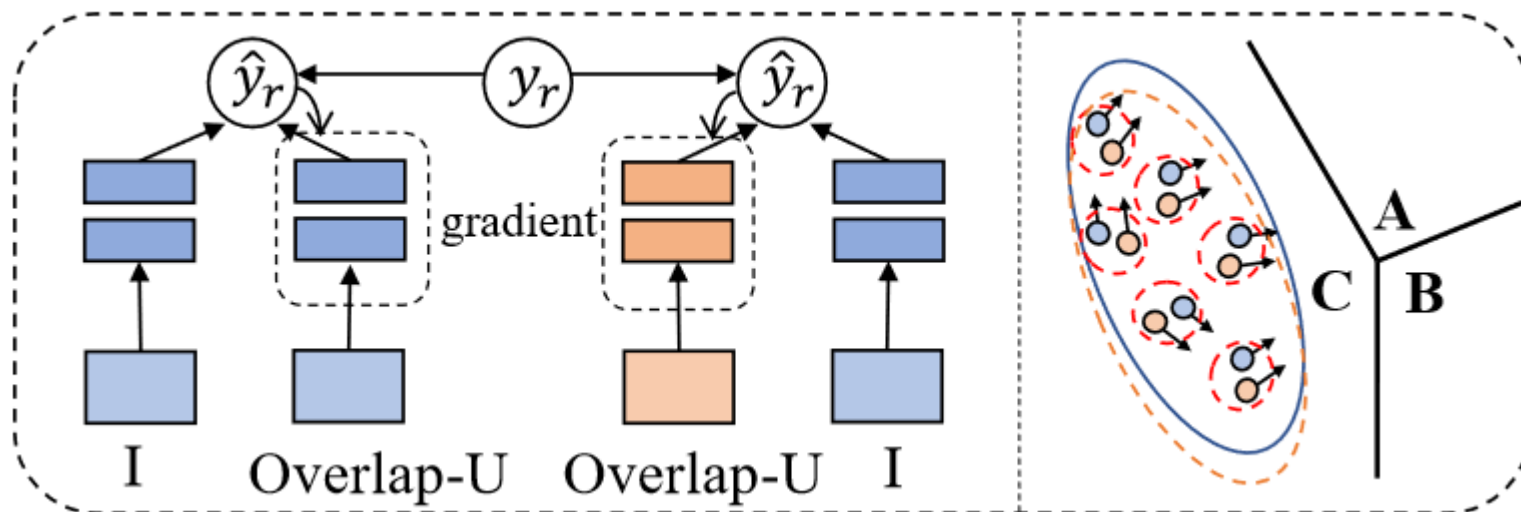
$$\ell(z_{\mathcal{T}}, q_S) = - \sum_k q_S^{(k)} \log p_{\mathcal{T}}^{(k)}$$

$$p_{\mathcal{T}}^{(k)} = \frac{\exp(\frac{1}{\tau} z_{\mathcal{T}}^{\top} c_k)}{\sum_{k'} \exp(\frac{1}{\tau} z_{\mathcal{T}}^{\top} c_{k'})}$$

$$\mathcal{L}_{u,u} = \ell(z_{\mathcal{T}}, q_S) + \ell(z_S, q_{\mathcal{T}})$$

COAST(WWW' 23)

3.2. 用户-项目对齐:



- 通过梯度对齐来诱导不同的view遵循交互项目的相同优化路径，确保用户对项目的兴趣一致

$$g_S = \mathbb{E}_{(u,v) \sim (\mathcal{U}_o, \mathcal{V}_S)} [\nabla_{\theta_{f_s^u}} \ell_{ce}(F_s^u(e^u) \cdot (F_s^v(e^v)))^\top, y_{u,v}]$$

$$g_T = \mathbb{E}_{(u,v) \sim (\mathcal{U}_o, \mathcal{V}_T)} [\nabla_{\theta_{f_t^u}} \ell_{ce}(F_t^u(e^u) \cdot (F_t^v(e^v)))^\top, y_{u,v}]$$

$$\mathcal{L}_{U,I} = 1 - \frac{g_S^\top \cdot g_T}{\|g_S\|_2 \|g_T\|_2}$$

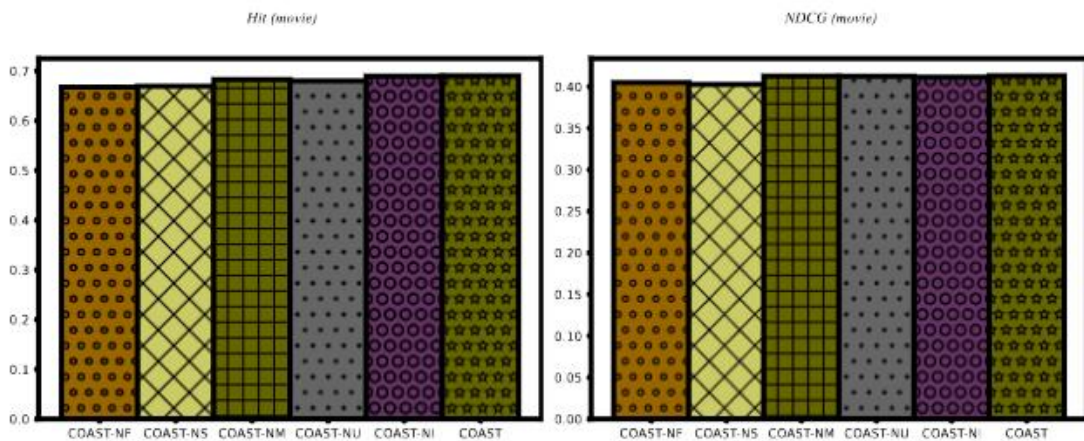
COAST(WWW' 23)

数据集:

data sets	Douban			Industrial Platform	
Domains	Movie	Music	Book	Mall	Community
Users	2,712	1,672	2,110	35,233	29,355
Items	34,893	5,567	6,777	1,749	2,452
Interactions	1,278,401	69,709	96,041	319,795	175,802
Density	1.35%	0.75%	0.67%	0.52%	0.24%
Tasks		Richer	Sparser	Overlap	
Douban	Task1	Movie	Book	2,106	
	Task2	Movie	Music	1,666	
	Task3	Book	Music	1,566	
Industrial Platform	Task4	Mall	Community	3,146	

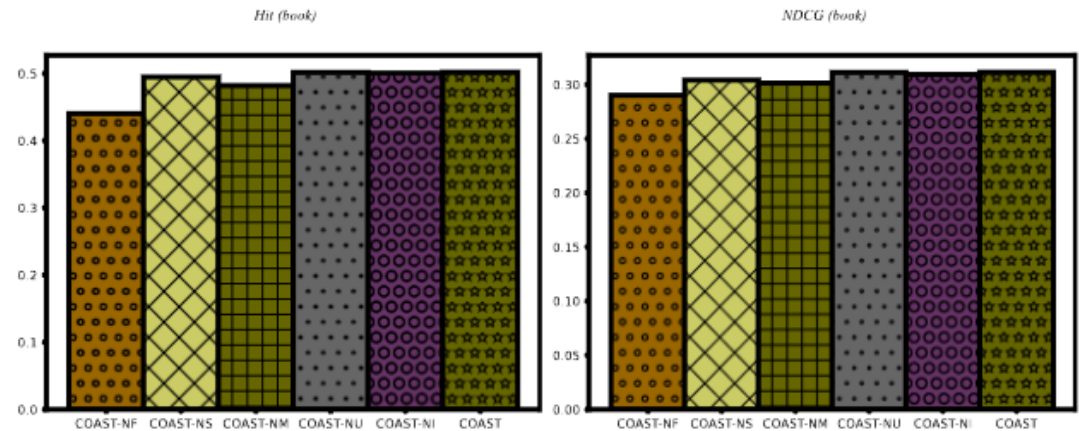
COAST(WWW' 23)

Algorithm	Task1				Task2				Task3				Task4			
	Movie		Book		Movie		Music		Book		Music		Mall		Community	
	Hit	NDCG	Hit	NDCG	Hit	NDCG	Hit	NDCG	Hit	NDCG	Hit	NDCG	Hit	NDCG	Hit	NDCG
NMF	0.5445	0.3154	0.3916	0.2224	0.5445	0.3154	0.3959	0.2206	0.3916	0.2224	0.3959	0.2206	0.5850	0.3265	0.3793	0.2048
LightGCN	0.6174	0.3492	0.3805	0.2226	0.6174	0.3492	0.3528	0.2023	0.3805	0.2226	0.3528	0.2023	0.5848	0.2933	0.5119	0.2490
MVDNN	0.6382	0.3689	0.4654	0.2575	0.6414	0.3641	0.3965	0.2238	0.5104	0.2947	0.3923	0.2390	0.5963	0.3002	0.5211	0.2507
DTCDR	0.6420	0.3794	0.4302	0.2394	0.6197	0.4278†	0.3593	0.2211	0.5108†	0.3263†	0.2848	0.2017	0.5580	0.3109	0.3632	0.2643
DDTCDR	0.5937	0.3558	0.4436	0.2511	0.5921	0.3722	0.3467	0.2189	0.4540	0.2666	0.3086	0.2042	0.5135	0.2884	0.3729	0.1886
DML	0.6060	0.3638	0.4662	0.2662	0.6093	0.4059	0.3821	0.2287	0.4521	0.2616	0.4253†	0.2548†	0.5491	0.3181	0.4283	0.2124
GADTCDR	0.6817†	0.4205†	0.4882†	0.3026†	0.6818†	0.4276	0.4383†	0.2498†	0.4492	0.2761	0.3571	0.1933	0.6654†	0.4055†	0.5173†	0.2907†
CDRIB	0.6114	0.3301	0.4630	0.2772	0.6411	0.3578	0.4103	0.2272	0.5021	0.2654	0.2866	0.2038	0.5744	0.3007	0.4802	0.2814
COAST	0.6905	0.4271	0.5052	0.3174	0.6938	0.4292	0.4497	0.2515	0.5138	0.3293	0.4688	0.2712	0.6769	0.4073	0.5503	0.3195
Improvement	1.2909%	1.5696%	3.4821%	4.8909%	1.7600%	0.3273%	2.6001%	0.6805%	0.5873%	0.9194%	10.2280%	6.4364%	1.7283%	0.4439%	6.3793%	9.9071%



(a) Hit@10 of Douban-movie.

(b) NDCG@10 of Douban-movie.



(c) Hit@10 of Douban-book.

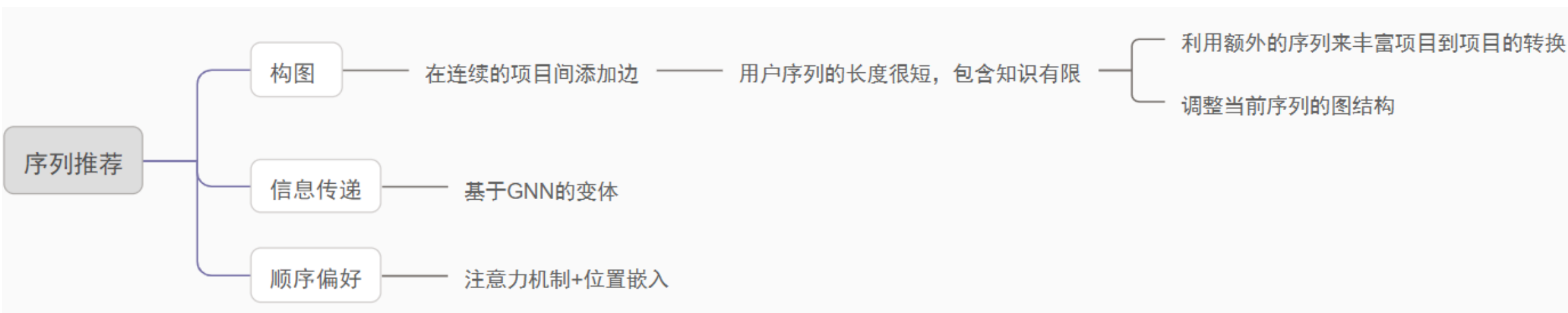
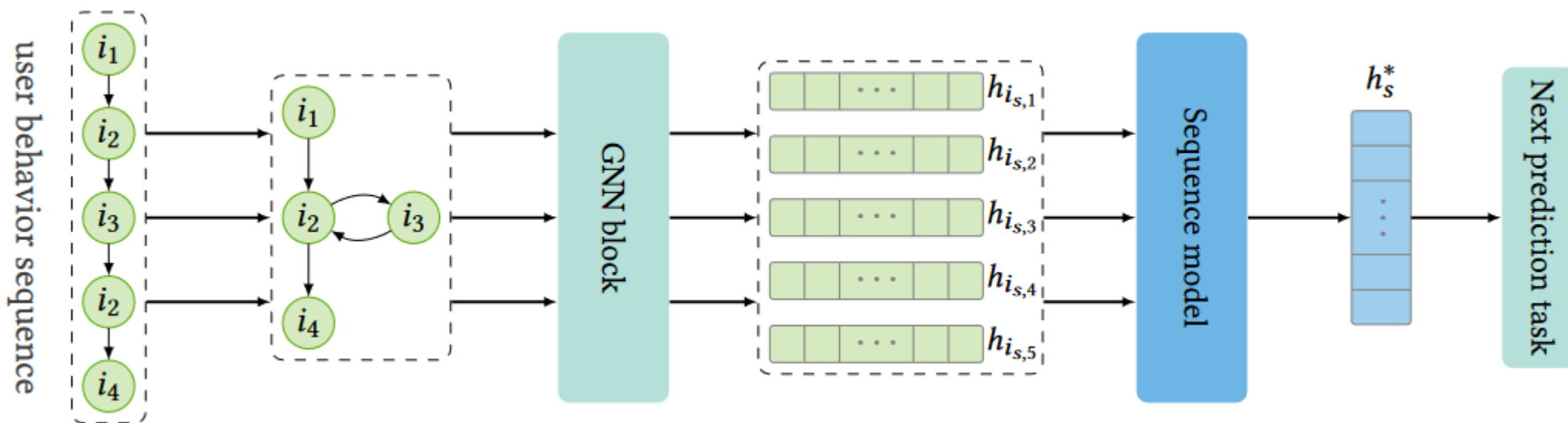
(d) NDCG@10 of Douban-book.

基于序列信息的推荐

2.

基于序列信息的推荐

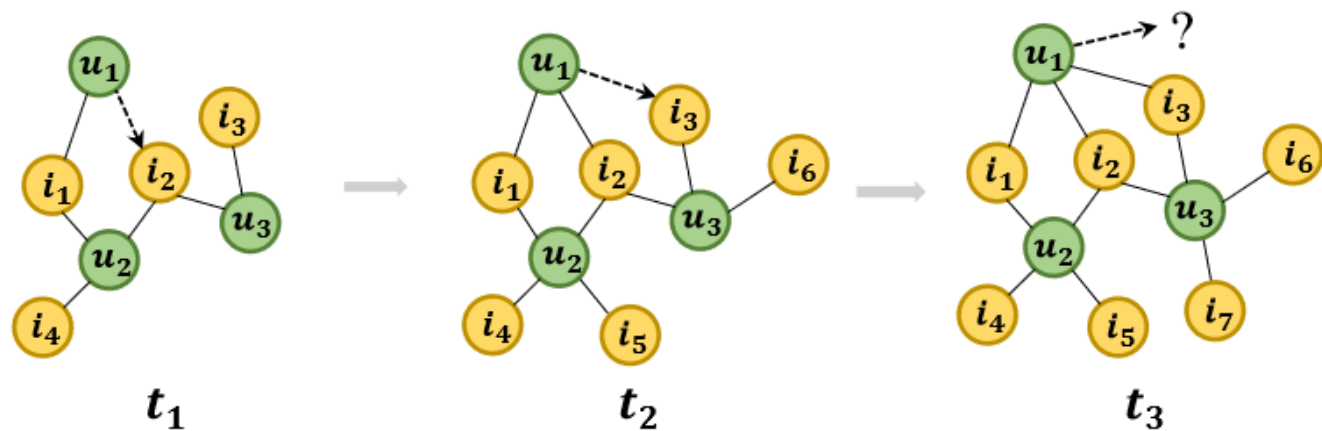
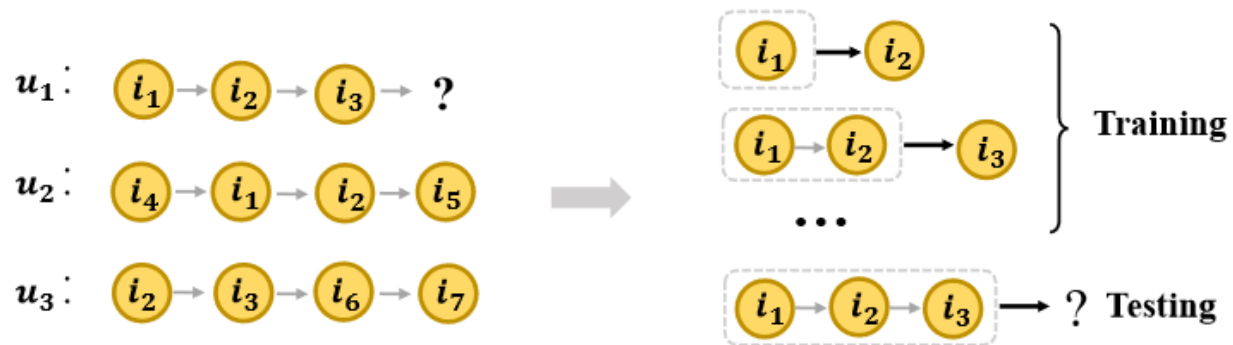
基于序列信息的推荐



Dynamic Graph Neural Networks for Sequential Recommendation

Mengqi Zhang, Shu Wu, *Member, IEEE*, Xueli Yu, Qiang Liu, *Member, IEEE*, Liang Wang, *Fellow, IEEE*

DGRN(TKDE'22)



现有方法的不足

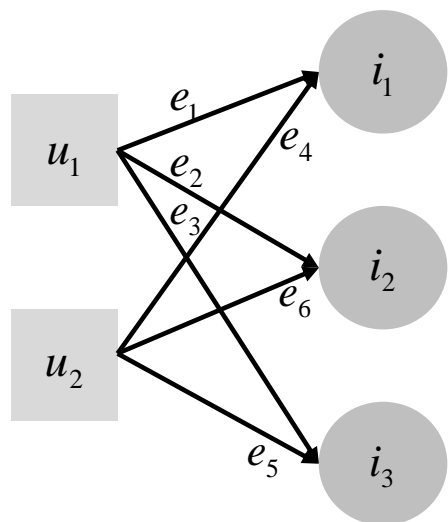
- 没有显式地利用不同用户序列之间的协作信息
- 忽略了高阶协作信息在不同时间的动态影响

本文的贡献

- 子图采样来动态提取包含用户序列和关联序列的子图捕捉高阶协作信息。
- 设计动态图形推荐网络（DGRN），从动态图中提取用户和项目的长短期偏好和特征，并将下一项预测任务转换为动态图中用户节点和项目节点之间的链接预测。

DGRN(TKDE'22)

1. 异构图构建: $G = \{(u, i, t, o_u^i, o_i^u) \mid u \in U, i \in V\} \quad G^t \in G$



$$e_1: (u_1, i_1, t_1, 1, 1)$$

$$e_2: (u_1, i_2, t_2, 2, 1)$$

$$e_3: (u_1, i_3, t_3, 3, 1)$$

$$e_4: (u_2, i_1, t_4, 1, 2)$$

$$e_5: (u_2, i_3, t_5, 2, 2)$$

$$e_6: (u_2, i_2, t_6, 3, 2)$$

DGRN(TKDE'22)

2.Sub-Graph Sampling:

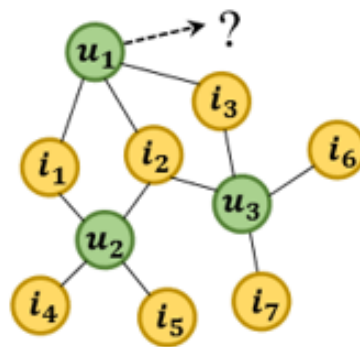
Algorithm 1: Sub-graph Sampling Algorithm

Input : Sequence $S^u = (i_1, i_2, \dots, i_k)$, timestamp sequence $T^u = (t_1, t_2, \dots, t_k)$, dynamic graph \mathcal{G}^{t_k} , and the order of sub-graph m .

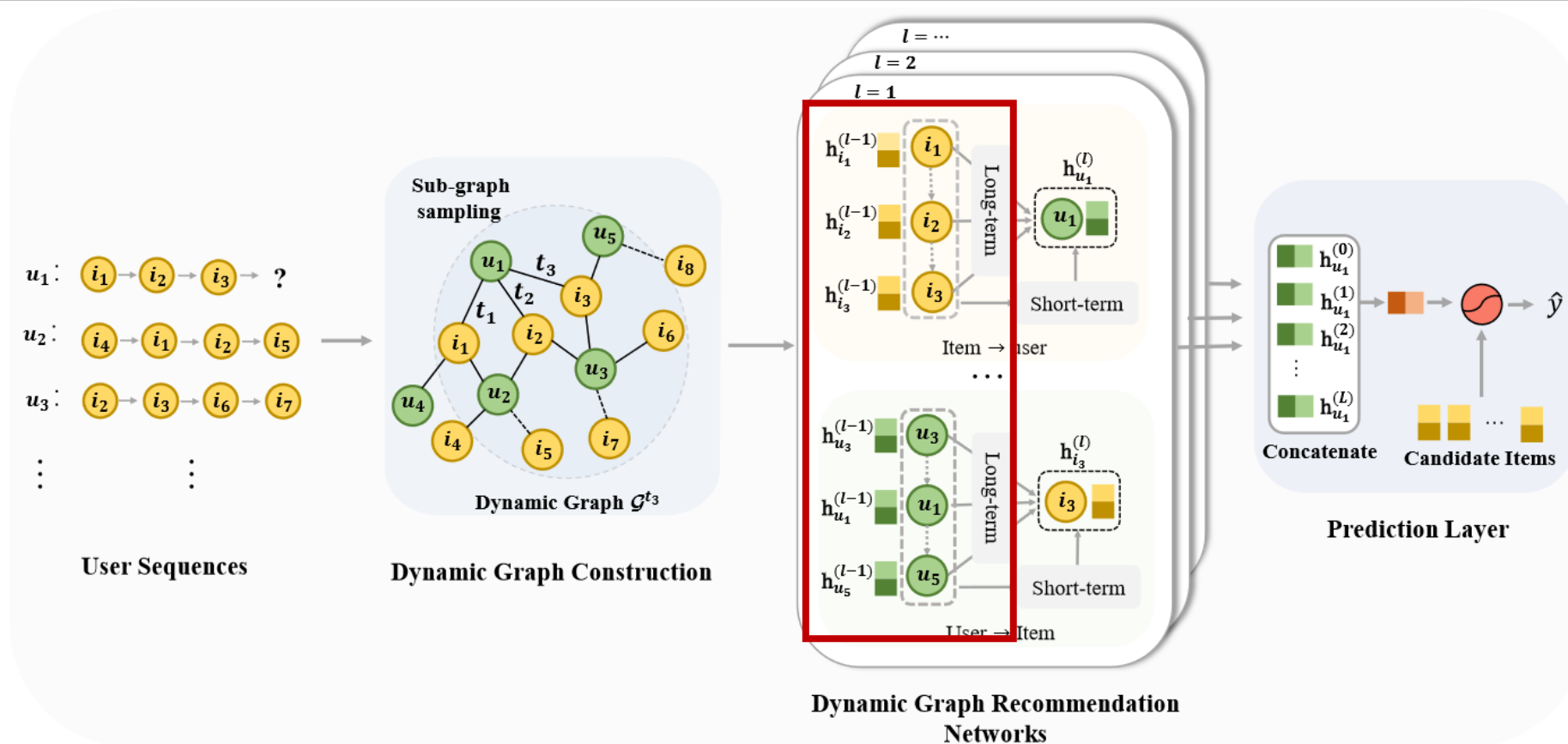
Output: The m -order sub-graph $\mathcal{G}_u^m(t_k)$.

```
1 // Initialization
2  $\mathcal{U}_m, \mathcal{U}_{temp} \leftarrow \{u\}, \mathcal{I}_m, \mathcal{I}_{temp} \leftarrow \{i_1, \dots, i_k\}, j = 0$ 
3 // Node sampling
4 while  $j \leq m$  do
5   for  $i \in \mathcal{I}_{temp}$  do
6      $\mathcal{U}_{temp} \leftarrow \mathcal{U}_{temp} \cup N_i$ 
7    $\mathcal{U}_{temp} \leftarrow \mathcal{U}_{temp} \setminus \mathcal{U}_m$ 
8    $\mathcal{U}_m \leftarrow \mathcal{U}_m \cup \mathcal{U}_{temp}$ 
9   if  $\mathcal{U}_{temp} = \emptyset$  then
10    Break
11  for  $u \in \mathcal{U}_{temp}$  do
12     $\mathcal{I}_{temp} \leftarrow \mathcal{I}_{temp} \cup N_u$ 
13   $\mathcal{I}_{temp} \leftarrow \mathcal{I}_{temp} \setminus \mathcal{I}_m$ 
14   $\mathcal{I}_m \leftarrow \mathcal{I}_m \cup \mathcal{I}_{temp}$ 
15  if  $\mathcal{I}_{temp} = \emptyset$  then
16    Break
17   $j = j + 1$ 
18 // Sub-graph generation
19  $\mathcal{G}_u^m(t_k) = (\mathcal{U}_m, \mathcal{I}_m), \mathcal{U}_m, \mathcal{I}_m \in \mathcal{G}^{t_k}$ 
```

1. 从 G^{t_k} 图中选择 u 的 n 个一阶邻居, 也就是说, 与 u 交互过的历史项目, 写成 N_u
2. 对于每一项 i , 我们使用它们作为一个锚节点去采样与它们交互过的用户, 写成 N_i
3. 记录已经被作为锚节点的用户和项目节点, 避免重复抽样
4. 重复采样, 生成用户 u 的 m 阶子图 $\mathcal{G}_u^m(t_k)$



DGRN(TKDE'22)



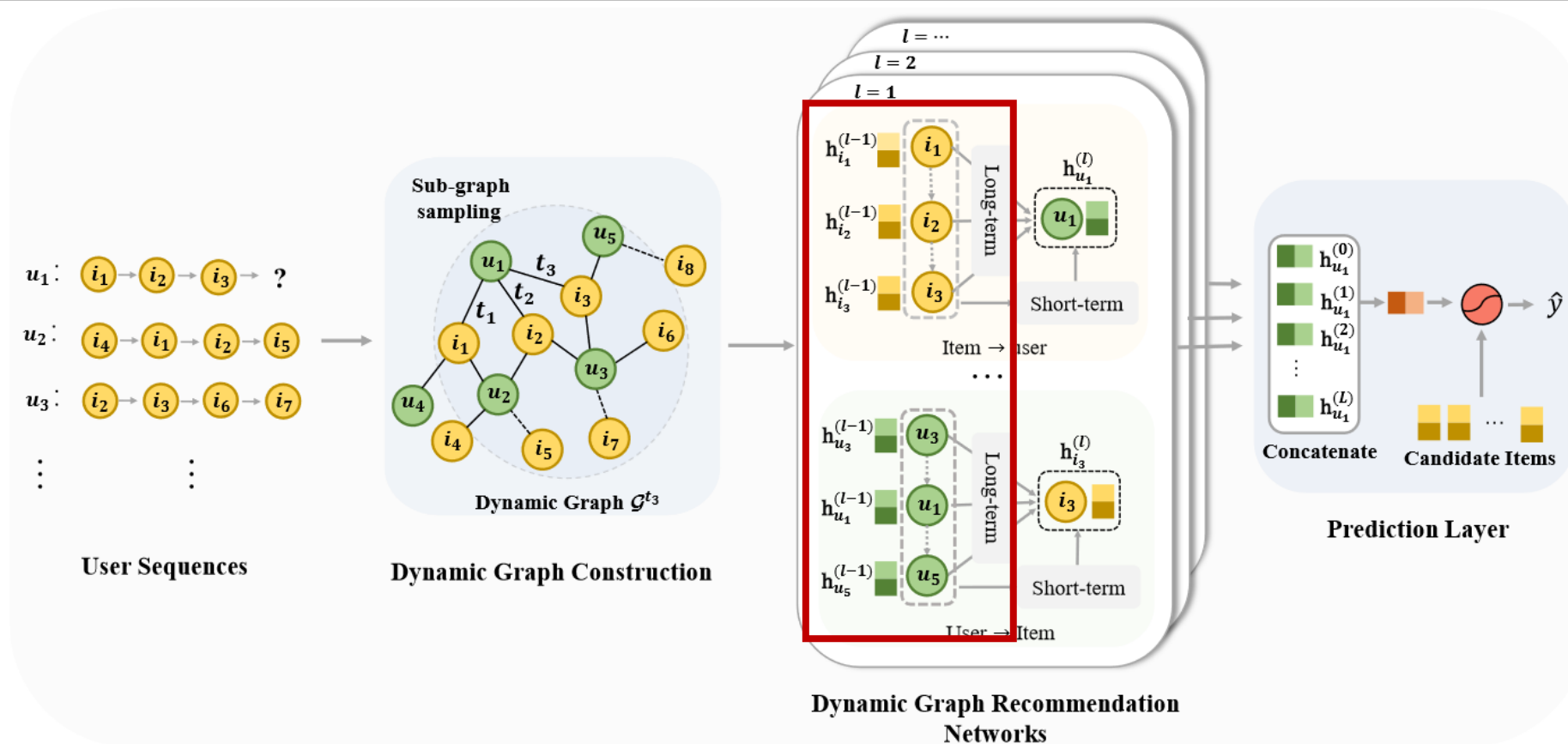
3.DGRN:

3.1. 消息传递:

长期的信息: $GCN+RNN+Attention$

$$h_u^L = \frac{1}{|\mathcal{N}_u|} \sum_{i \in \mathcal{N}_u} \mathbf{w}_1^{(l-1)} h_i^{(l-1)} \quad h_u^L = \text{GRU}_U^{(l)}(h_{i_1}^{(l-1)}, \dots, h_{i_{|\mathcal{N}_u|}}^{(l-1)}), i \in \mathcal{N}_u$$

DGRN(TKDE'22)



Attention:

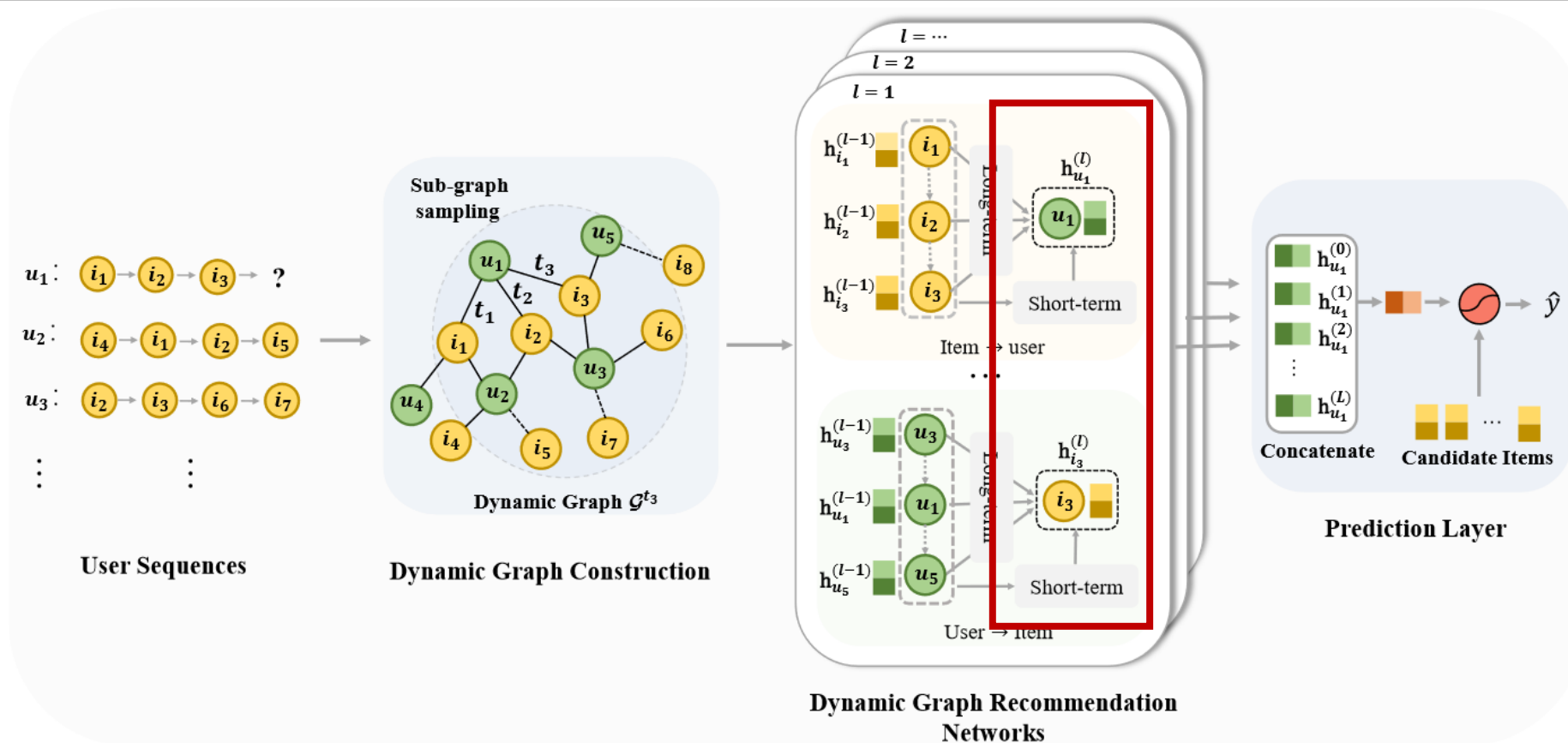
$$e_{ui} = \frac{(\mathbf{W}_2^{(l-1)} \mathbf{h}_u^{(l-1)})^\top (\mathbf{W}_1^{(l-1)} \mathbf{h}_i^{(l-1)} + \mathbf{p}_{r_u^i}^K)}{\sqrt{d}}$$

$$r_u^i = |\mathcal{N}_u| - o_u^i$$

融合了顺序信息

$$\alpha_{ui} = \text{softmax}(e_{ui}). \quad \mathbf{h}_u^L = \sum_{i \in \mathcal{N}_u} \alpha_{ui} (\mathbf{W}_1^{(l-1)} \mathbf{h}_i^{(l-1)} + \mathbf{p}_{r_u^i}^V)$$

DGRN(TKDE'22)



短期的消息: *Attention*

$$\hat{\alpha}_{ui} = \text{softmax} \left(\frac{\left(\mathbf{W}_3^{(l-1)} \mathbf{h}_{i_{|\mathcal{N}_u|}}^{(l-1)} \right)^T \left(\mathbf{W}_2^{(l-1)} \mathbf{h}_i^{(l-1)} \right)}{\sqrt{d}} \right)$$

$$\mathbf{h}_u^S = \sum_{i \in \mathcal{N}_u} \hat{\alpha}_{ui} \mathbf{h}_i^{(l-1)}$$

DGRN(TKDE'22)

3.2.节点更新：将长期嵌入、短期嵌入和前一层嵌入进行聚合

$$\mathbf{h}_u^{(l)} = \tanh \left(\mathbf{W}_3^{(l)} \left[\mathbf{h}_u^L \parallel \mathbf{h}_u^S \parallel \mathbf{h}_u^{(l-1)} \right] \right)$$

4.推荐与最优化：在模型中，预测用户 u 在 t_k 的下一次交互相当于预测子图 $G_u^m(t_k)$ 中 u 的链接

$$\mathbf{s}_{ui} = \mathbf{h}_u^T \mathbf{W}_P \mathbf{e}_i$$

$$\hat{\mathbf{y}}_u = \text{softmax}(\mathbf{s}_u)$$

$$Loss = - \sum_S \sum_{i=1}^{|\mathcal{I}|} y_{ui} \log(\hat{y}_{ui}) + (1 - y_{ui}) \log(1 - \hat{y}_{ui}) + \lambda \|\Theta\|_2$$

DGRN(TKDE'22)

数据集:

Datasets	Beauty	Games	CDs
# of Users	52,024	31,013	17,052
# of Items	57,289	23,715	35,118
# of Interactions	394,908	287,107	472,265
Average length	7.6	9.3	27.6
Density	0.01%	0.04%	0.08%

模型效果:

Datasets	Metric	BPR-MF	FPMC	GRU4Rec+	Caser	SASRec	SR-GNN	HGN	TiSASRec	HyperRec	DGSR	Gain
Beauty	NDCG@10	21.83	28.91	26.42	25.47	32.19	32.33	<u>32.47</u>	30.45	23.26	35.90	10.56%
	Hit@10	37.75	43.10	43.98	42.64	48.54	48.62	<u>48.63</u>	46.87	34.71	52.40	7.75%
Games	NDCG@10	28.75	46.80	45.64	45.93	<u>53.60</u>	53.25	49.34	50.19	48.96	55.70	3.92%
	Hit@10	37.75	68.02	67.15	68.83	<u>73.98</u>	73.49	71.42	71.85	71.24	75.57	2.15%
CDs	NDCG@10	36.26	33.55	44.52	45.85	49.23	48.95	<u>49.34</u>	48.97	47.16	51.22	3.81%
	Hit@10	56.27	51.22	67.84	68.65	71.32	69.63	<u>71.42</u>	71.00	71.02	72.43	1.41%

DGRN(TKDE'22)

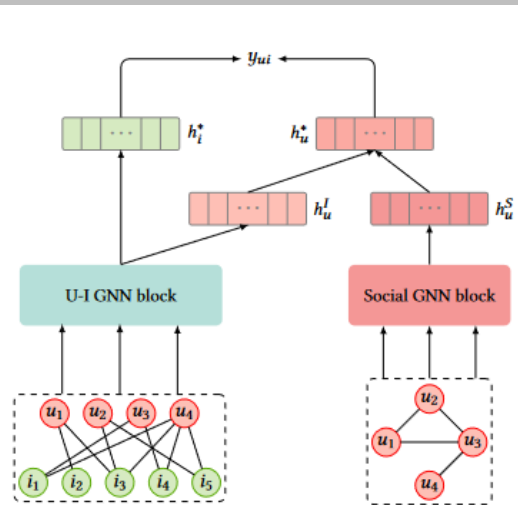
Variants	Ablation		Beauty		Games		CDs	
	Long-term	Short-term	NDCG@10	Hit@10	NDCG@10	Hit@10	NDCG@10	Hit@10
DGSR-G	GCN	–	33.75	49.94	53.44	73.23	48.66	70.43
DGSR-R	RNN	–	34.81	50.90	54.70	74.73	49.57	71.22
DGSR-D	DAT	–	35.25	51.36	55.12	74.83	49.66	71.26
DGSR-L	–	Last	30.87	46.13	52.43	72.18	46.23	67.38
DGSR-A	–	ATT	34.76	51.00	54.30	74.32	48.78	70.09
DGSR-GL	GCN	Last	35.24	51.18	54.76	74.58	49.62	70.76
DGSR-RL	RNN	Last	35.47	51.68	54.86	74.84	50.26	71.24
DGSR-DL	DAT	Last	35.62	51.92	55.53	75.07	50.72	72.06
DGSR-GA	GCN	ATT	35.00	51.05	54.97	74.78	50.05	71.46
DGSR-RA	RNN	ATT	35.17	51.46	55.02	74.88	51.19	72.55
DGSR-DA	DAT	ATT	35.90	52.40	55.70	75.57	51.22	72.43

社会网络增强的推荐

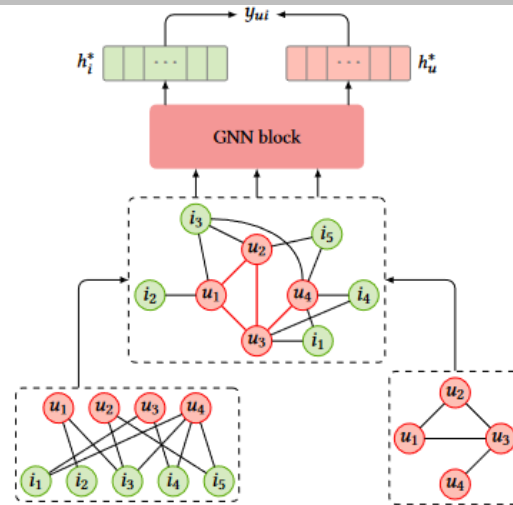
3.

社会网络增强的推荐

社会网络增强的推荐



(a) The framework of GNN on the bipartite graph and social network graph separately.



(b) The framework of GNN on the unified graph of user-item interactions and social network.

基于社会影响的推荐

朋友的影响力

同等影响力×

注意力机制

社会关系并不总是可靠

具有显性社交关系的用户可能没有影响力

—— 过滤不相关的关系

获得的社会关系可能是不完整的

—— 加入隐式关系

偏好整合

分离图

线性整合

—— 相同权重


非线性整合

—— 动态分配权重

统一图

分层聚合

Multi-Order Hypergraph Convolutional Neural Network for Dynamic Social Recommendation System

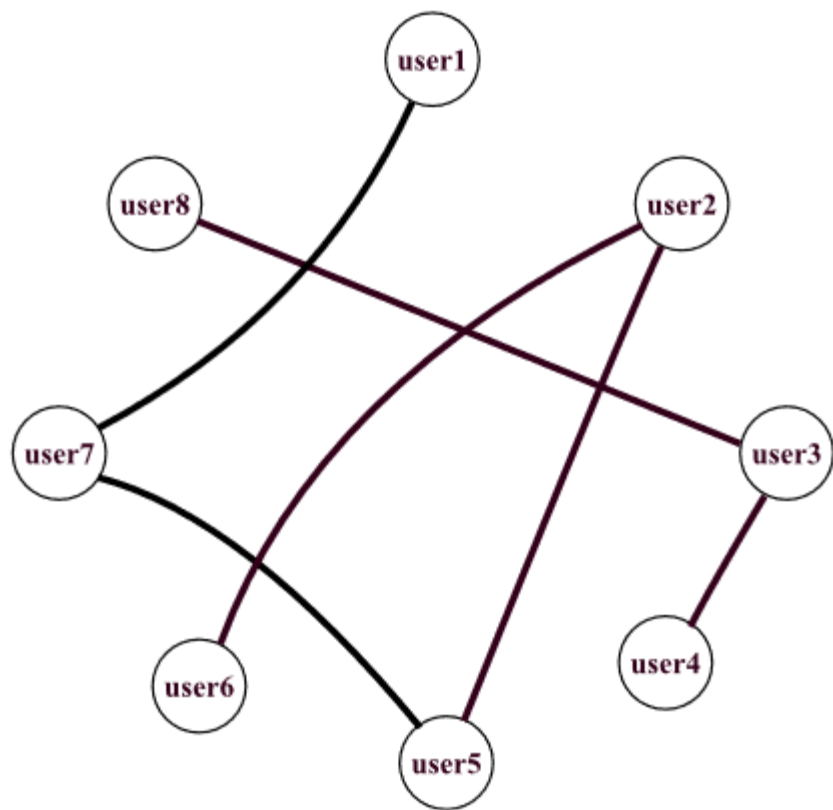
YU WANG¹  **AND QILONG ZHAO**²

¹Key Laboratory of High Confidence Software Technologies of Ministry of Education, School of Computer Science, Institute of Software, Peking University, Beijing 100871, China

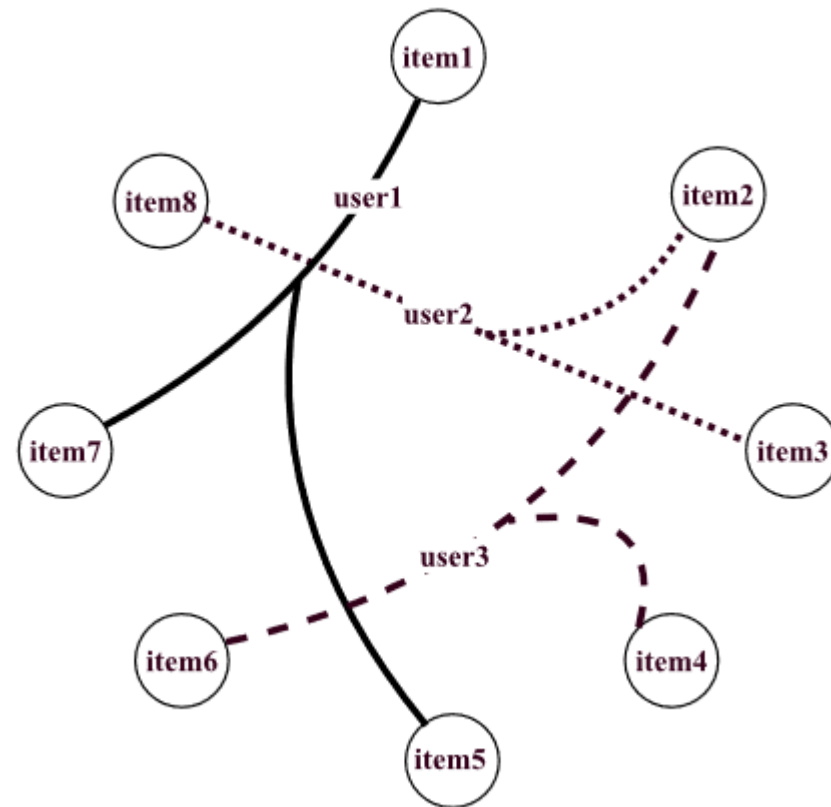
²Tencent, Beijing 100193, China

Corresponding author: Yu Wang (wangyu18@pku.edu.cn)

MOHCN(IEEE'22)

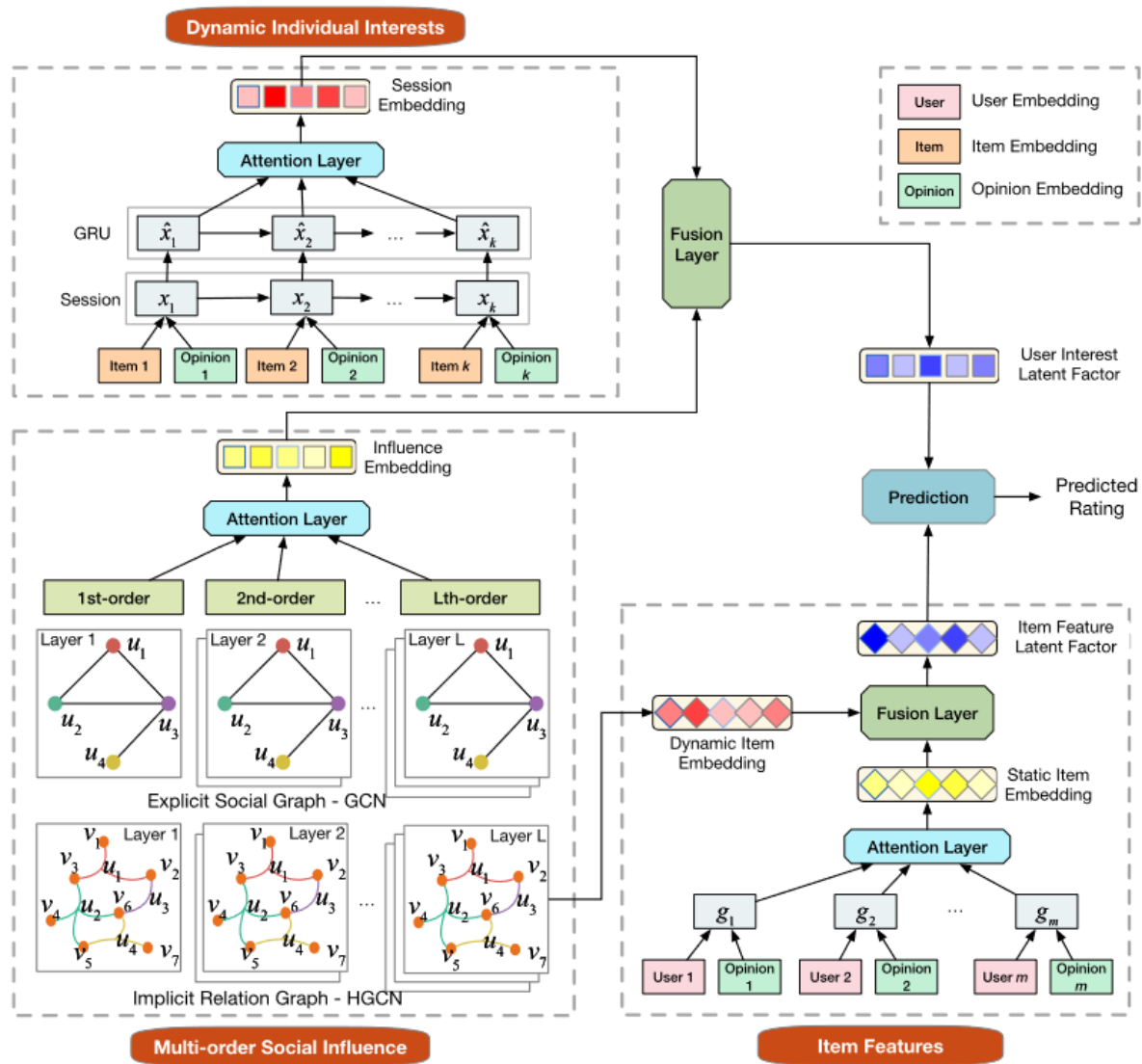


显式



隐式

MOHCN(IEEE'22)



现有方法的不足

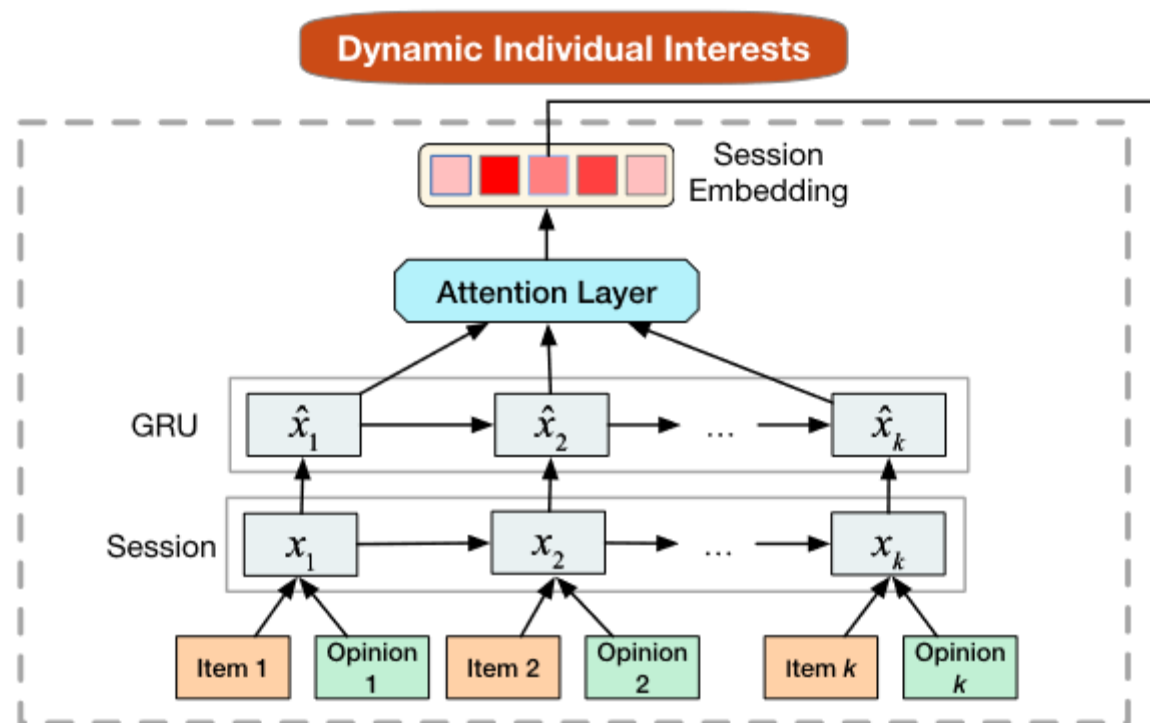
- 现有的高阶方法忽略了用户之间隐含的社会关系以及用户的兴趣随时间动态变化。

本文的贡献

- 突出社交网络中的多阶社交关系，以平衡不同阶的高阶社交，并通过超图建模增强高阶显性和隐性朋友的影响
- 提出MOHCN在会话级别对用户的动态兴趣演变进行建模

MOHCN(IEEE'22)

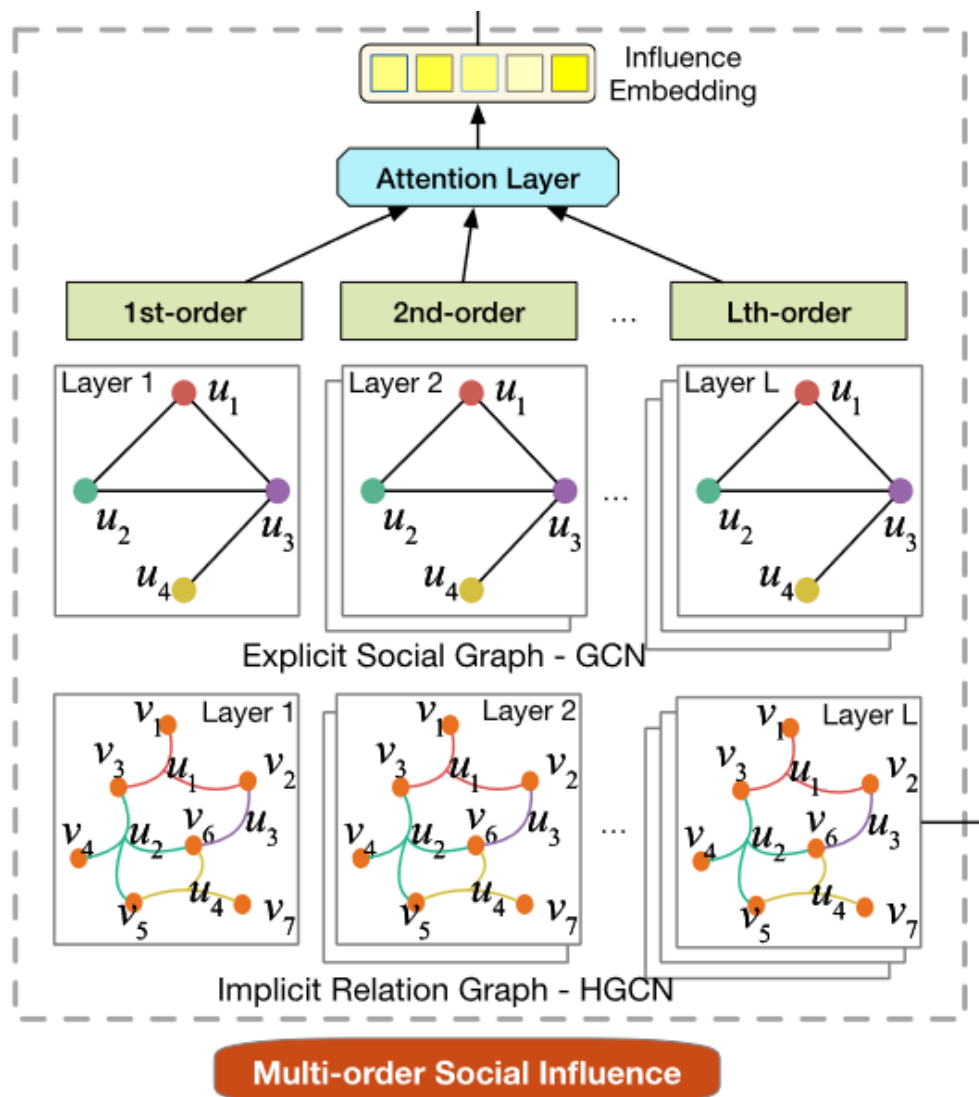
动态个人兴趣建模:



- 联合利用用户-物品图中的交互信息和意见信息来学习用户表示
- 考虑到用户的兴趣随时间的演变可以看作是一个会话，再使用GRU在会话级别提取用户的动态兴趣

MOHCN(IEEE'22)

多阶社会影响建模:



隐式关系:

- 顶点和超边分别代表项目和用户，用户的兴趣由连接到超边的顶点聚合。
- 通过注意力机制得到第 i 阶隐式好友影响的用户兴趣。
- 通过在隐式关系超图中堆叠 L 个 HGCN 层，我们可以获得受其 L 阶隐式朋友影响的用户兴趣。

显式关系:

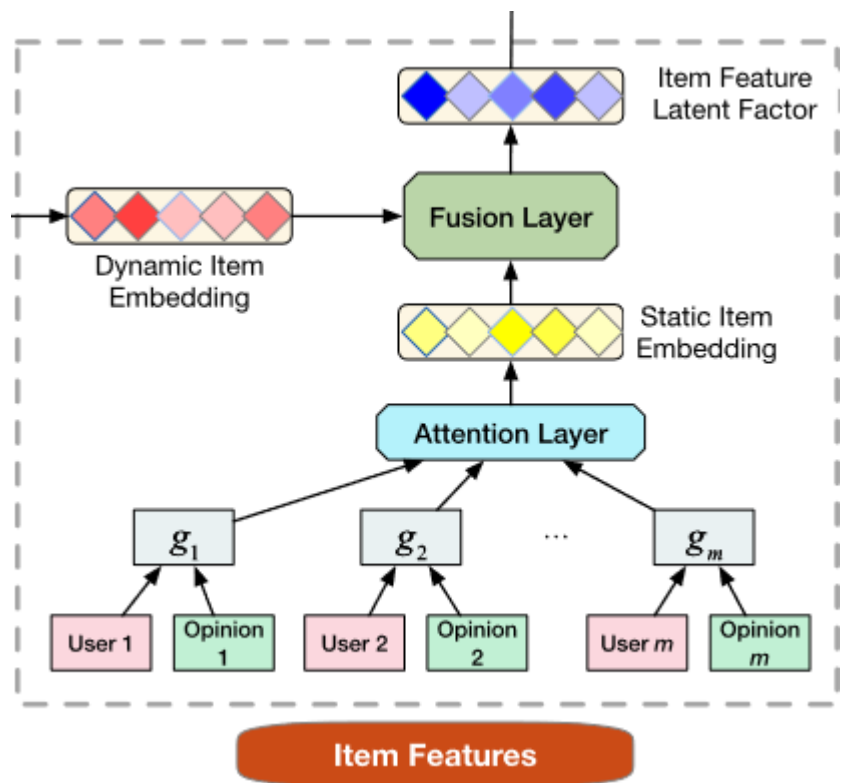
- 堆叠 L 个 GCN 层，用户可以从其 i 阶显式朋友聚合其兴趣。

$$h_u^i = f_y(h_{u_{imp}}^{(i)} \oplus h_{u_{exp}}^{(i)})$$

为了捕捉不同顺序的朋友的不同影响，采用注意力机制聚合不同顺序的高阶朋友对用户兴趣的影响。

MOHCN(IEEE'22)

物品特征建模:



动态的物品特征:

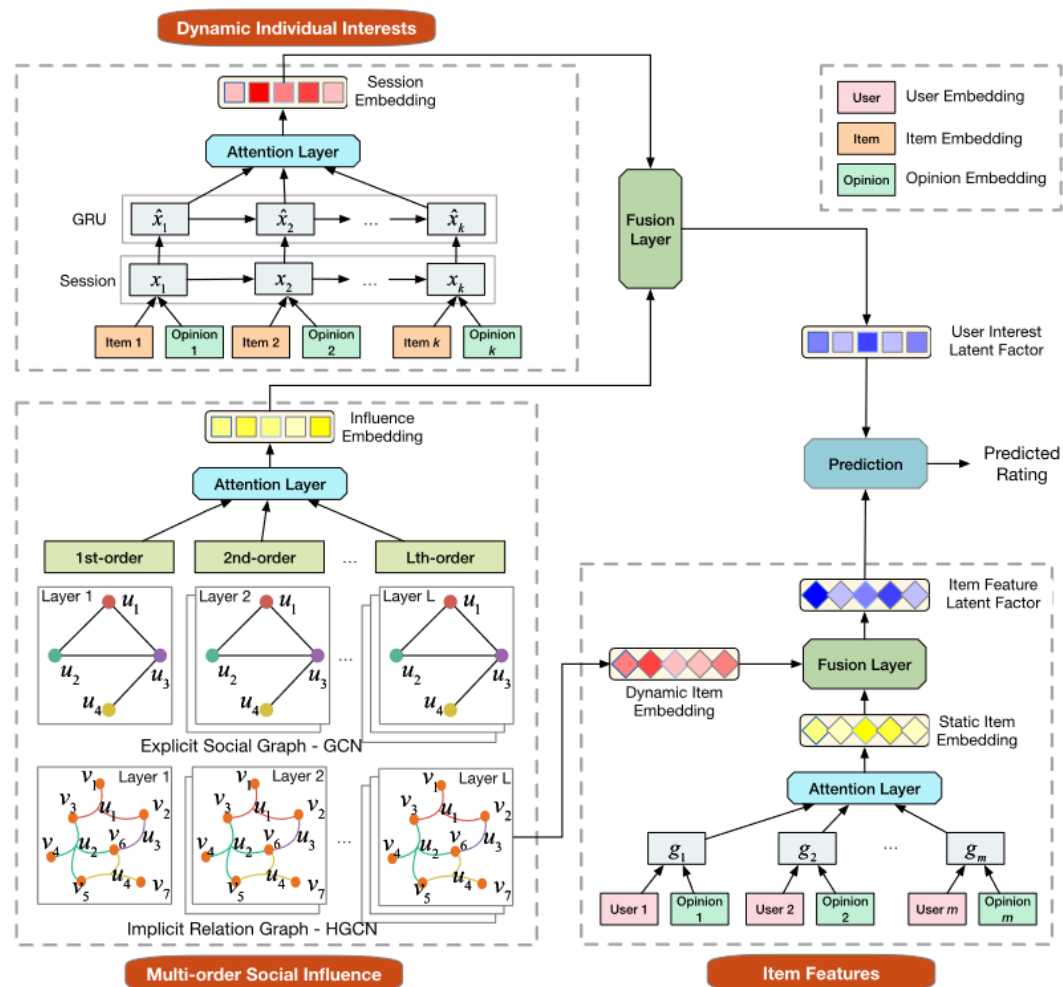
由于通过 HGCN 获得的项目嵌入可以随着用户隐式好友的变化而更新

静态的物品特征:

联合利用用户-物品图中的交互信息和意见信息来学习。

MOHCN(IEEE'22)

评级预测和模型训练:



$$r'_{ij} = f_z(\mathbf{h}_i \oplus \mathbf{y}_j).$$

$$\mathcal{L} = \frac{1}{2|T|} \sum_{(i,j) \in T} (r'_{ij} - r_{ij})^2 + \lambda \|\theta\|^2$$

MOHCN(IEEE'22)

Model Class	Models	Ciao		Epinions		Yelp	
		MAE	RMSE	MAE	RMSE	MAE	RMSE
Classic	PMF (A)	0.9021	1.1238	0.9952	1.2128	1.1740	1.5323
	NeuMF (A)	0.8062	1.0617	0.9072	1.1476	1.1368	1.3927
Social	SoRec (B)	0.8410	1.0652	0.8961	1.1437	0.9571	1.2485
	SoReg (B)	0.8611	1.0848	0.9119	1.1703	0.9487	1.2512
	SocialMF (B)	0.8270	1.0501	0.8837	1.1328	0.9365	1.2061
	TrustMF (B)	0.7690	1.0479	0.8410	1.1395	0.9270	1.1899
Dynamic	SASRec (C)	0.7791	1.0393	0.8414	1.1126	0.9177	1.1602
	SR-GNN (C)	0.7763	1.0375	0.8391	1.1027	0.9124	1.1589
	CD-GCN (C)	0.7706	1.0281	0.8310	1.0904	0.9092	1.1534
Social+Deep Learning	DeepSoR (D)	0.7739	1.0316	0.8383	1.0972	0.9033	1.1452
	DGRec (D)	0.7531	1.0127	0.8263	1.0846	0.8975	1.1301
	TrustEV (D)	0.9388	1.2491	0.8151	1.0755	0.8755	1.1316
	GraphRec (D)	0.7387	0.9794	0.8168	1.0631	0.8621	1.1288
	HOSR (D)	0.7372 [†]	0.9740	0.8177	1.0643	0.8412 [†]	1.1127 [†]
	ConsisRec (D)	0.7394	0.9722 [†]	0.8046 [†]	1.0495 [†]	0.8437	1.1204
Our Proposal	MOHCN	0.6988 ± 0.0031	0.9362 ± 0.0044	0.7905 ± 0.0060	1.0327 ± 0.0065	0.8272 ± 0.0042	1.0920 ± 0.0057
	Impr.	5.21%	3.70%	1.75%	1.60%	1.77%	1.86%

Semantic and Structural view Fusion Modeling for Social Recommendation

Kun Yuan, Guannan Liu, Junjie Wu, Hui Xiong, *Fellow, IEEE*

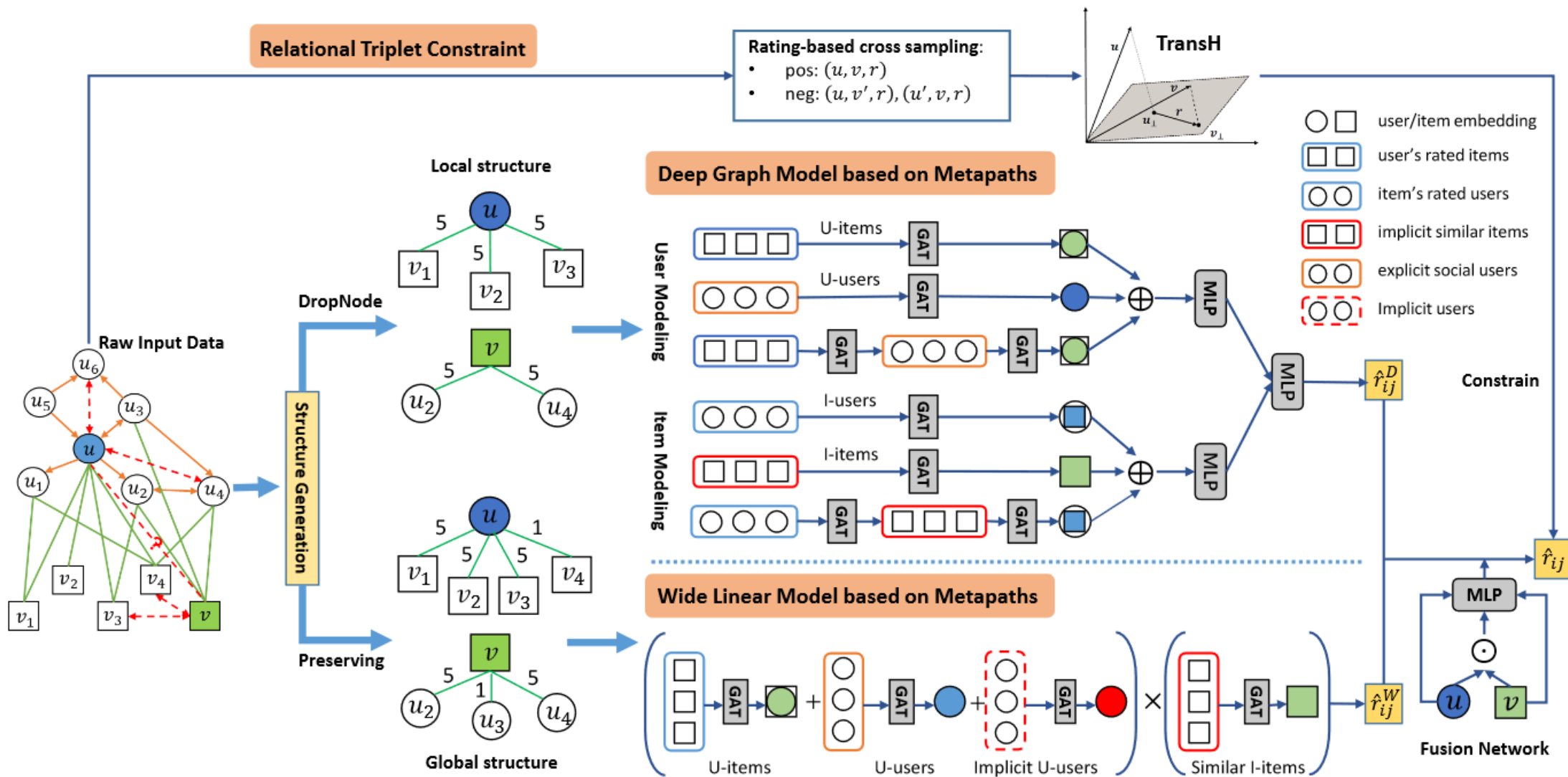
现有工作的不足

- 观察到的社交关系的稀疏性和不平衡分布阻碍了社会化推荐的进一步发展。
- 没有关注到对于用户评分行为，不同级别的评分可以表示用户和物品在embedding空间中的相对位置差异，即高评分物品应该比低评分物品更接近用户

本文的贡献

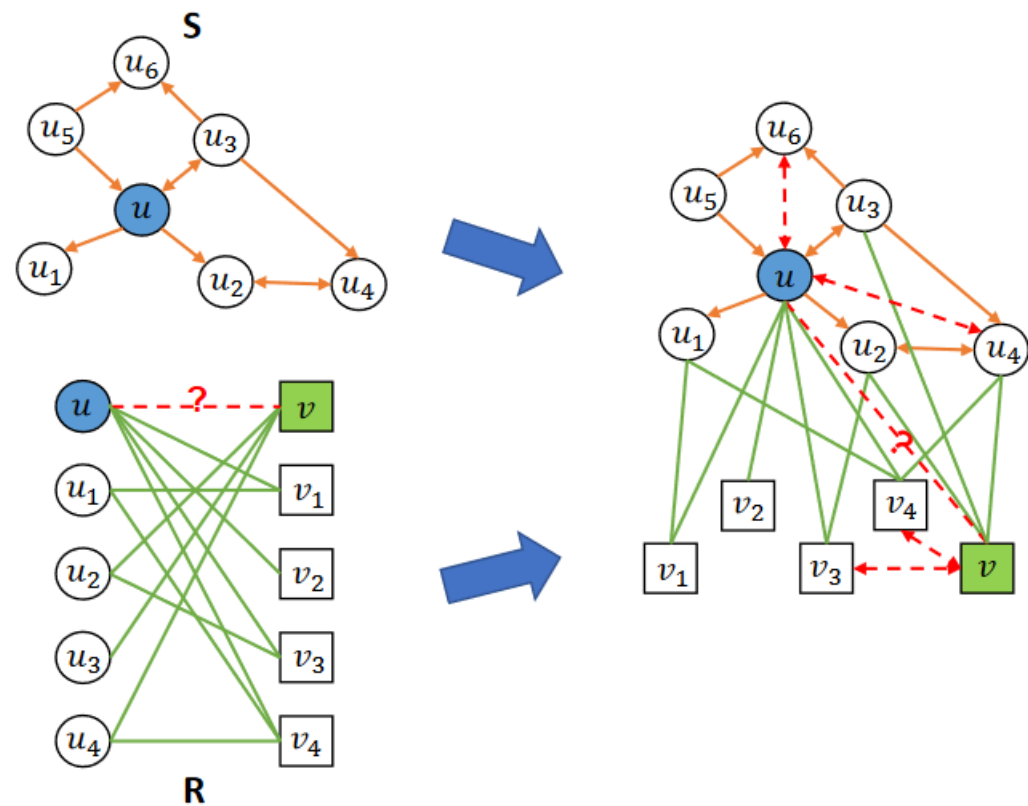
- 结合显式和隐式社会关系，构建了一个异构信息网络，通过定义不同类型的语义元路径，捕获全面的用户和物品表示。
- 提出了一个用于社会化推荐的语义和结构视图融合框架（S4Rec），该框架由一个深度图模型和广度注意力模型组成，将局部和全局结构作为输入，实现准确而稳定的评分预测。
- 提出了相对位置差异问题，设计（用户，评分，物品）三元组以提供一种良好的泛化策略，从而提高整体性能。

S4Rec(TKDE'22)



S4Rec(TKDE'22)

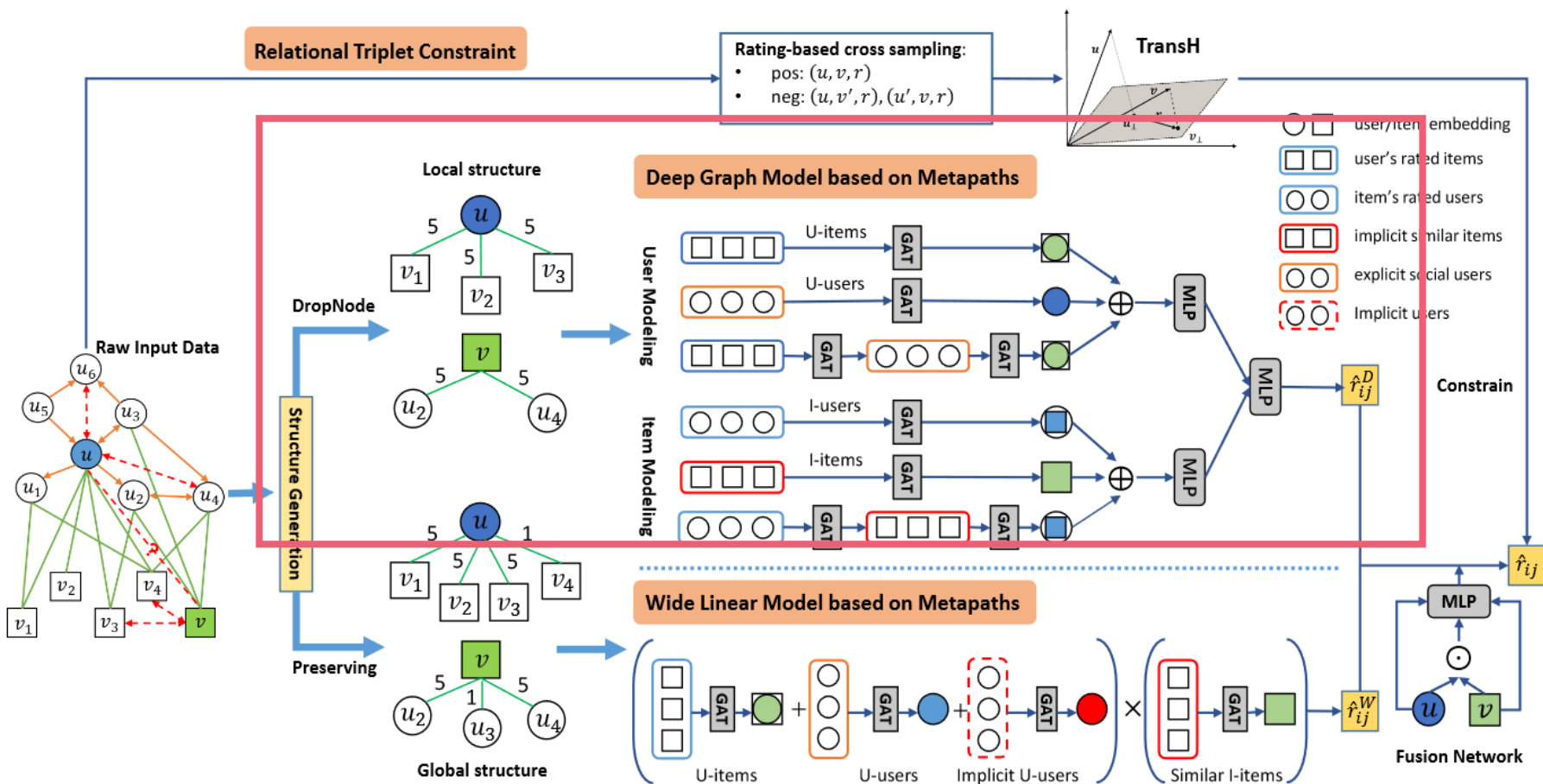
用于社会化推荐的元路径



	Metapaths	Example
user	user-items user-users user-implicit users user-users-items user-implicit users-items	$u \leftarrow \{v_1, v_2, v_4\}$ $u \leftarrow \{u_3, u_5\}$ $u \leftarrow \{u_4, u_6\}$ $u \leftarrow u_3 \leftarrow \{v\}$ $u \leftarrow u_4 \leftarrow \{v, v_4\}$
item	item-users item-implicit items item-implicit items-users	$v \leftarrow \{u_2, u_3, u_4\}$ $v \leftarrow \{v_3, v_4\}$ $v \leftarrow v_4 \leftarrow \{u, u_1, u_4\}$

S4Rec(TKDE'22)

用于局部视图的深度图模型



考虑到活跃用户和具有大量交互的流行物品将生成大量的元路径

随机扔掉一些节点到固定的大小

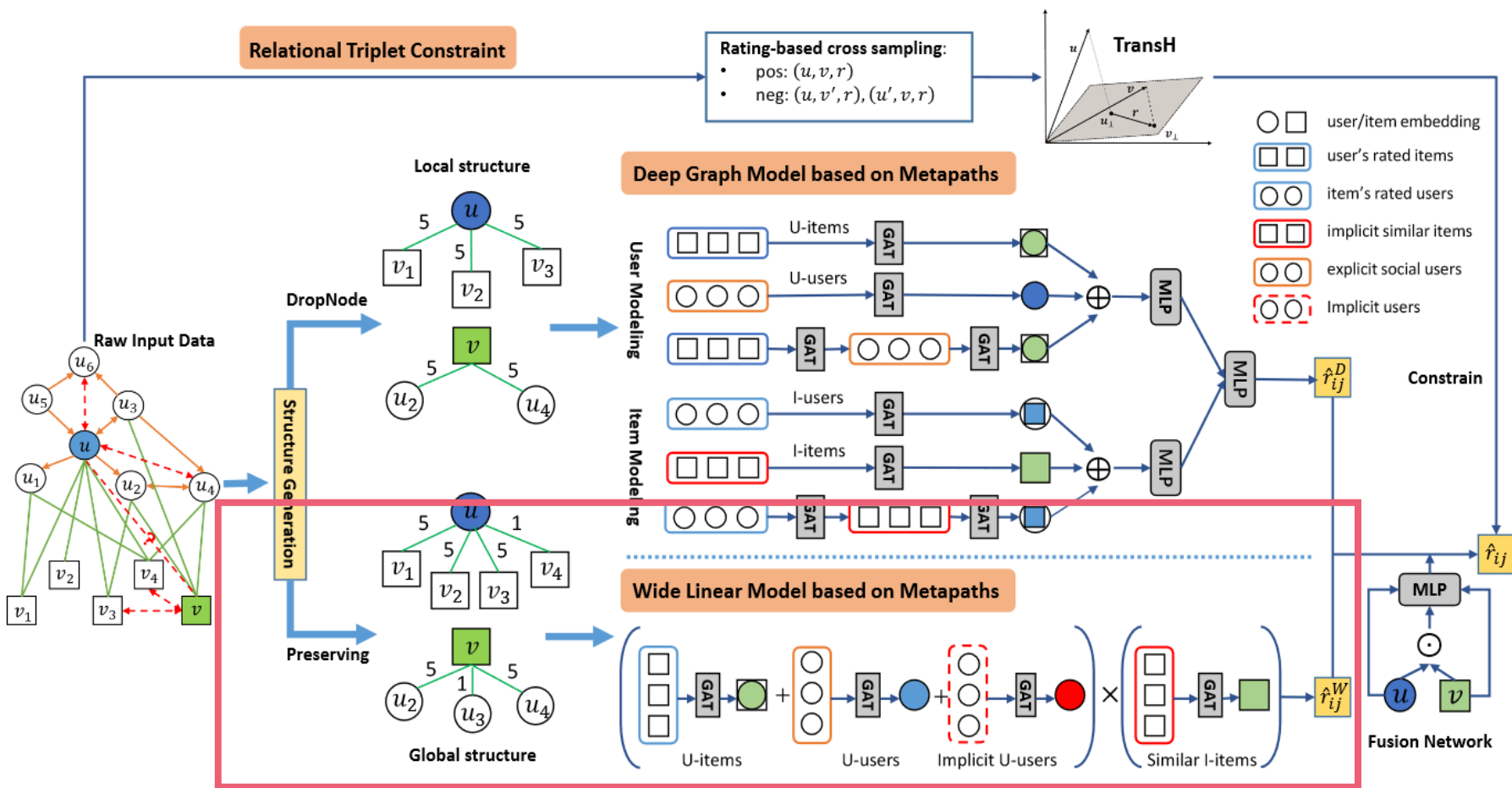
对于每个元路径，使用元路径引导的GAT

用户: $\hat{p}_i = MLP_{user}([p_i^V \oplus p_i^S \oplus p_i^{SV} \oplus p_i^H \oplus p_i^{HV}])$

项目: $\hat{q}_j = MLP_{item}([q_j^U \oplus q_j^H \oplus q_j^{HU}])$

S4Rec(TKDE'22)

用于全局视图的广度线性注意模型

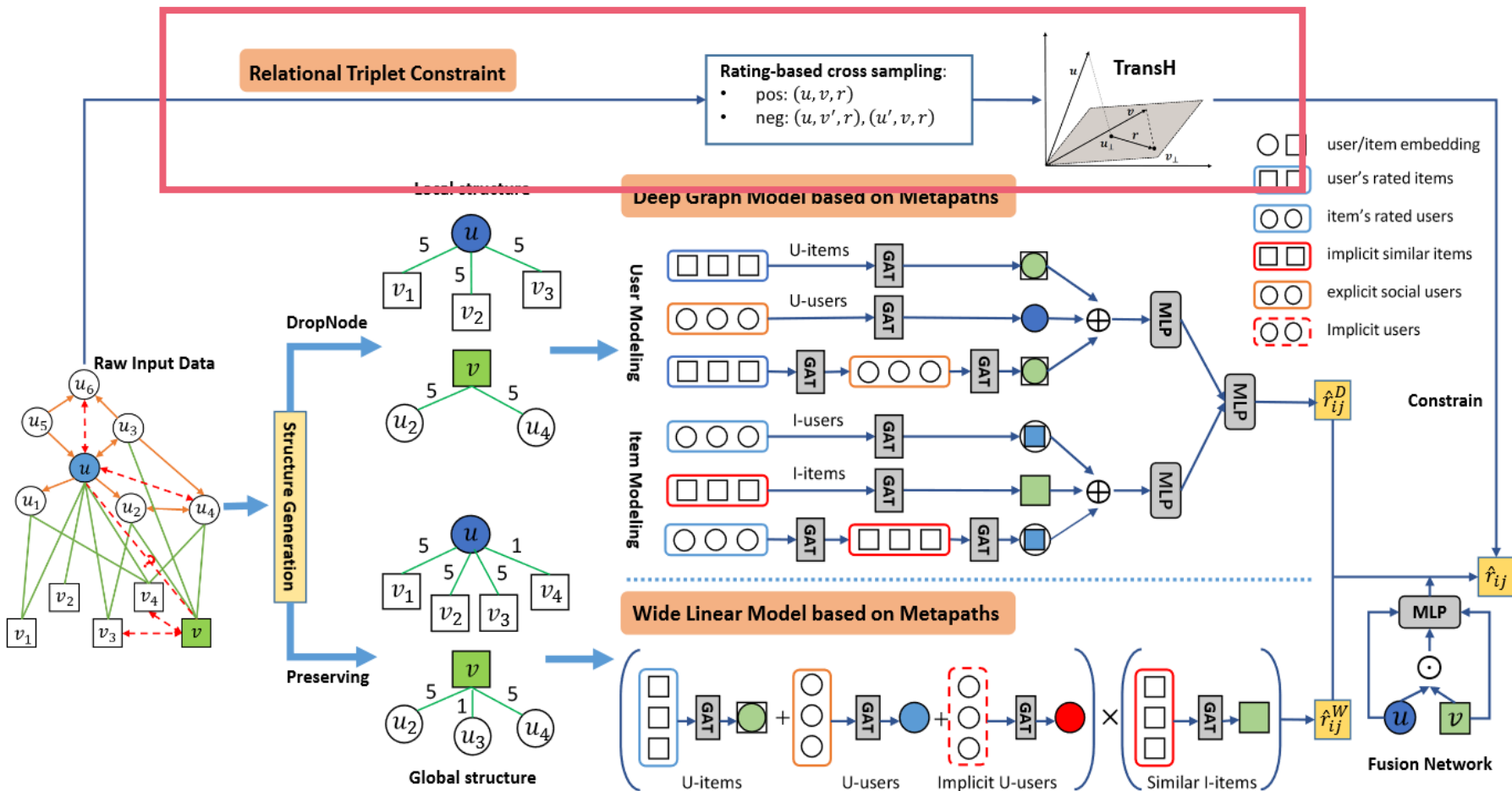


DropNode生成的局部结构导致对大多数物品的高精度预测结果，但对次要的低评分物品的高方差

从单跳关系的全局角度出发，提出一种宽而浅的稳定性建模模型

S4Rec(TKDE'22)

关系三元组约束



用户具有较高评分的物品应该更好地反映用户偏好



将用户评分行为转换为
(用户, 评分, 物品)
形式的三元组

$$f(u, r, v) = \| p_u^r + e_r - q_v^r \|^2$$

S4Rec(TKDE'22)

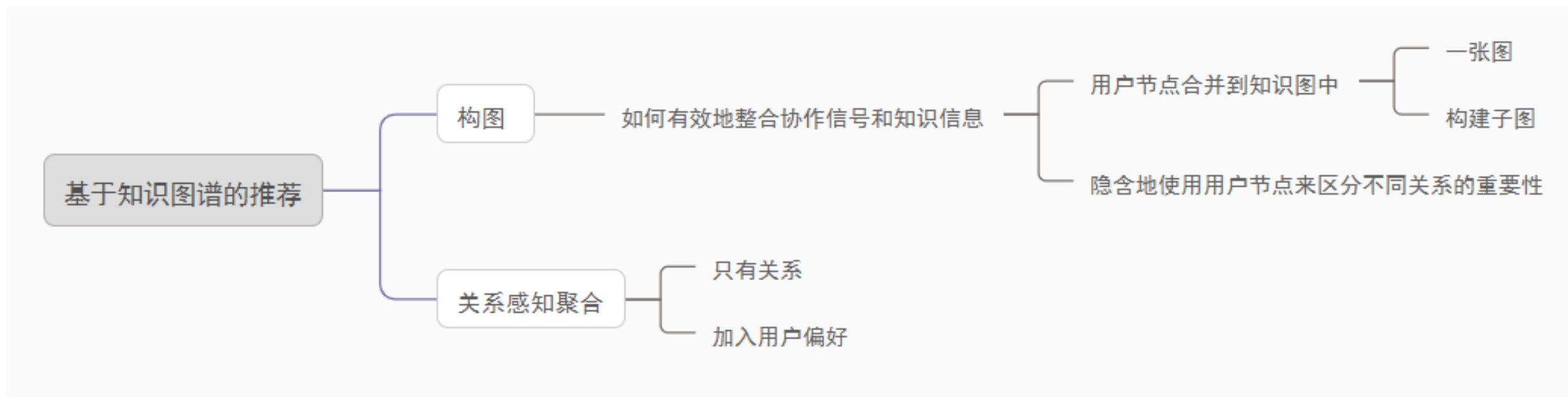
Model	Epinions		Ciao		yelp	
	MAE	RMSE	MAE	RMSE	MAE	RMSE
NeuMF	0.7903(± 0.0039)	1.0707(± 0.0113)	0.7168(± 0.0110)	0.9832(± 0.0135)	0.7885(± 0.0027)	1.0281(± 0.0037)
CUNE	0.8172(± 0.0041)	1.0826(± 0.0145)	0.7353(± 0.0049)	1.0005(± 0.0072)	0.7947(± 0.0037)	1.0409(± 0.0049)
IF-BPR	0.8045(± 0.0016)	1.0707(± 0.0026)	0.7302(± 0.0054)	0.9957(± 0.0075)	0.7867(± 0.0025)	1.0346(± 0.0032)
SocialMF	0.8254(± 0.0032)	1.0789(± 0.0056)	0.7482(± 0.0052)	0.9890(± 0.0076)	0.8037(± 0.0034)	1.0368(± 0.0045)
TrustMF	0.8298(± 0.0028)	1.0954(± 0.0062)	0.7416(± 0.0055)	0.9917(± 0.0081)	0.8149(± 0.0048)	1.0680(± 0.0052)
TrustSVD	0.8035(± 0.0028)	1.0557(± 0.0045)	0.7216(± 0.0041)	<u>0.9575(± 0.0065)</u>	0.7802(± 0.0029)	<u>1.0085(± 0.0040)</u>
ConsisRec	0.8029(± 0.0023)	1.0541(± 0.0036)	0.7252(± 0.0043)	0.9581(± 0.0069)	0.7829(± 0.0023)	1.0145(± 0.0037)
GraphRec	0.8029(± 0.0019)	1.0656(± 0.0037)	0.7253(± 0.0069)	0.9850(± 0.0081)	0.7900(± 0.0028)	1.0254(± 0.0035)
Danser	<u>0.7866(± 0.0016)</u>	<u>1.0454(± 0.0046)</u>	<u>0.7117(± 0.0048)</u>	0.9641(± 0.0055)	<u>0.7766(± 0.0041)</u>	1.0118(± 0.0031)
S4Rec-SocialMF	0.7838(± 0.0031)	1.0441(0.0021)	0.7138(± 0.0067)	0.9584(± 0.0089)	0.7710(± 0.0023)	1.0074(± 0.0036)
S4Rec-TrustMF	0.7813(± 0.0023)	1.0583(± 0.0055)	0.7093(± 0.0075)	0.9717(± 0.0140)	0.7726(± 0.0075)	1.0251(± 0.0048)
S4Rec	0.7754(± 0.0028)**	1.0396(± 0.0046)**	0.7007(± 0.0052)**	0.9498(± 0.0063)**	0.7643(± 0.0023)**	1.0026(± 0.0028)**

知识图谱增强的推荐

4.

知识图谱增强的推荐

知识图谱增强的推荐



Learning Intents behind Interactions with Knowledge Graph for Recommendation

Xiang Wang¹, Tinglin Huang², Dingxian Wang³, Yancheng Yuan⁴, Zhenguang Liu²,
Xiangnan He^{5*}, Tat-Seng Chua¹

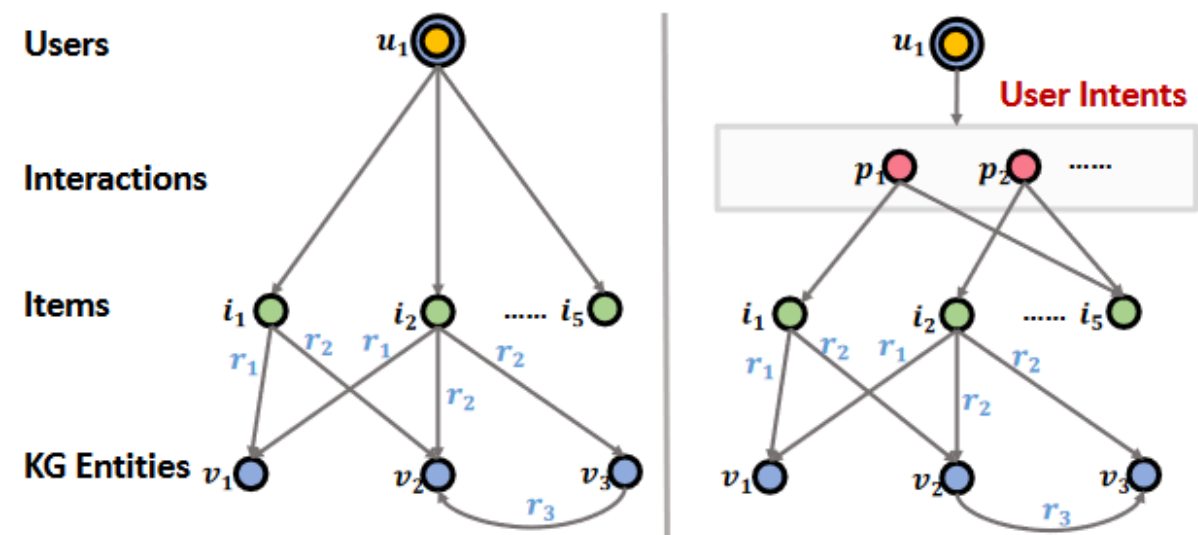
¹National University of Singapore, ²Zhejiang University, ³eBay,

⁴The Hong Kong Polytechnic University, ⁵University of Science and Technology of China

xiangwang@u.nus.edu, tinglin.huang@zju.edu.cn, diwang@ebay.com

{yanchengyuanmath, liuzhenguang2008, xiangnanhe}@gmail.com, dcscts@nus.edu.sg

KGIN(WWW'21)



现有方法的不足

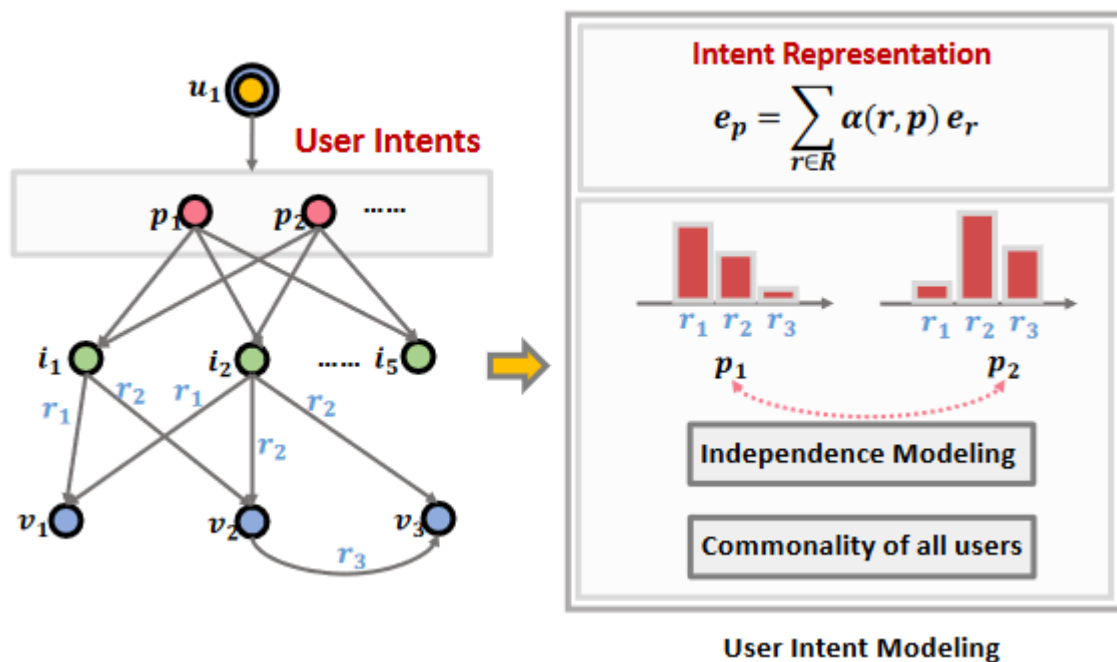
- 无法在细粒度的意图级别识别用户-项目关系
- 无法利用关系依赖性来保留远程连接的语义

本文的贡献

- 在基于KG的推荐中揭示交互背后的用户意图，提高模型容量和解释能力
- 提出了一个新的模型KGIN，该模型考虑了GNN范式下更细粒度的意图和关系路径的远程语义下的用户项关系

KGIN(WWW'21)

用户意图建模:



意图图

将 p 和KG中的所有关系进行attention处理

$$\mathbf{e}_p = \sum_{r \in \mathcal{R}} \alpha(r, p) \mathbf{e}_r$$

$$\alpha(r, p) = \frac{\exp(w_r p)}{\sum_{r' \in \mathcal{R}} \exp(w_{r'} p)}$$

p 有可能是相似的

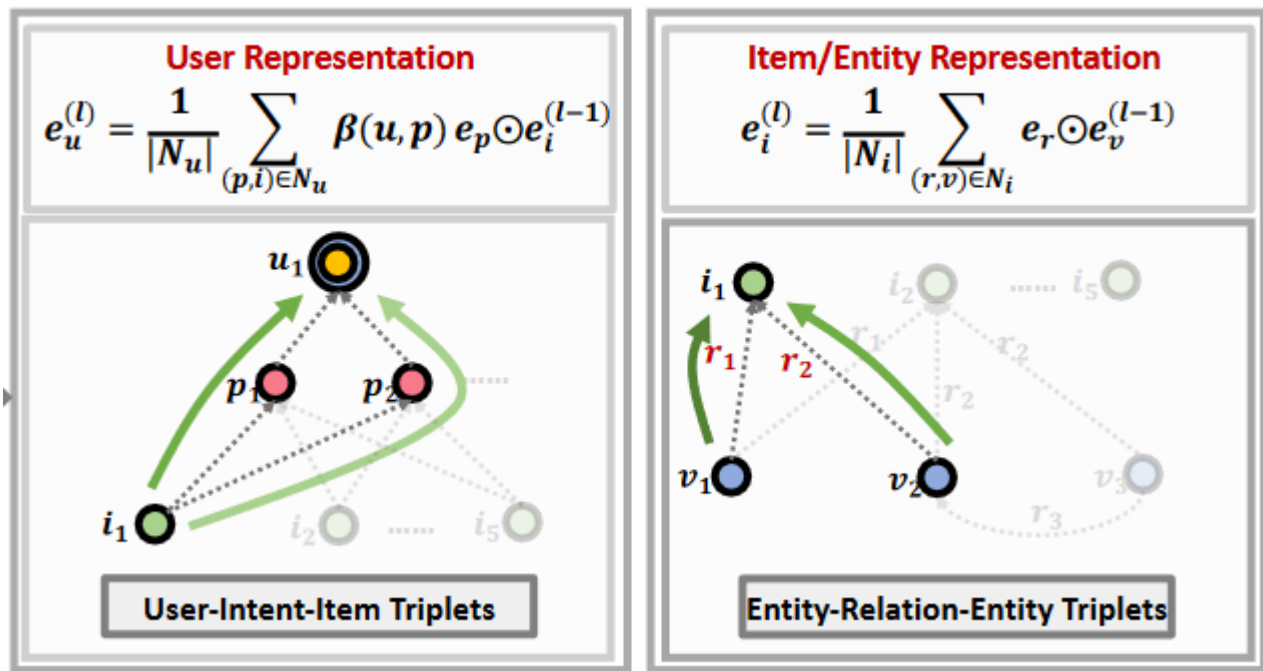
减少互信息或者距离相关性

$$\mathcal{L}_{\text{IND}} = \sum_{p \in \mathcal{P}} -\log \frac{\exp(s(\mathbf{e}_p, \mathbf{e}_p)/\tau)}{\sum_{p' \in \mathcal{P}} \exp(s(\mathbf{e}_p, \mathbf{e}_{p'})/\tau)}$$

$$\mathcal{L}_{\text{IND}} = \sum_{p, p' \in \mathcal{P}, p \neq p'} d\text{Cor}(\mathbf{e}_p, \mathbf{e}_{p'})$$

KGIN(WWW'21)

关系路径感知聚合：



意图图上的聚合层：

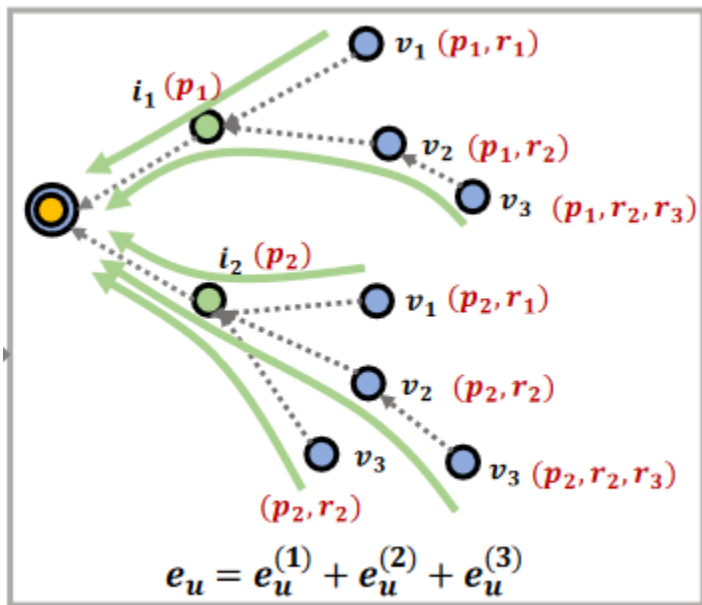
- 考虑不同的意图将会有不同的贡献来激励他的行为。因此引入了一个注意力评分来区分意图p的重要性

知识图谱上的聚合层：

- 由于一个实体可以涉及多个KG三元组，因此它可以将其他连接的实体作为其属性

KGIN(WWW'21)

捕获关系路径：



Final User (or Item) Representation

堆叠更多聚合层，收集高阶邻居的信息。

$$\mathbf{e}_u^{(l)} = f_{IG}\left(\{(\mathbf{e}_u^{(l-1)}, \mathbf{e}_p, \mathbf{e}_i^{(l-1)}) \mid (p, i) \in \mathcal{N}_u\}\right)$$

$$\mathbf{e}_i^{(l)} = f_{KG}\left(\{(\mathbf{e}_i^{(l-1)}, \mathbf{e}_r, \mathbf{e}_v^{(l-1)}) \mid (r, v) \in \mathcal{N}_i\}\right)$$

模型预测和优化：

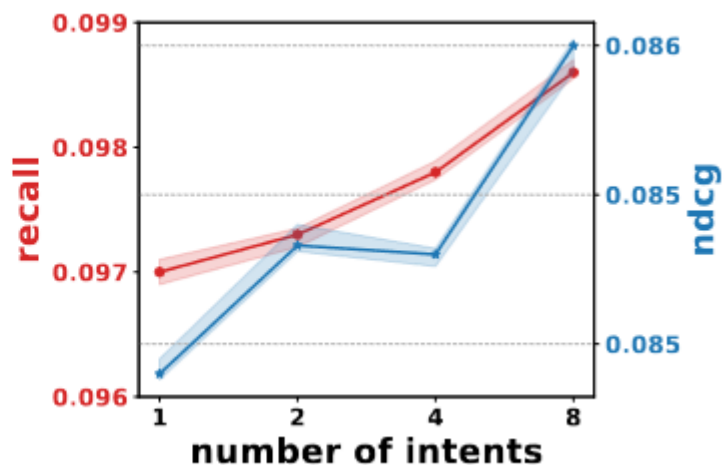
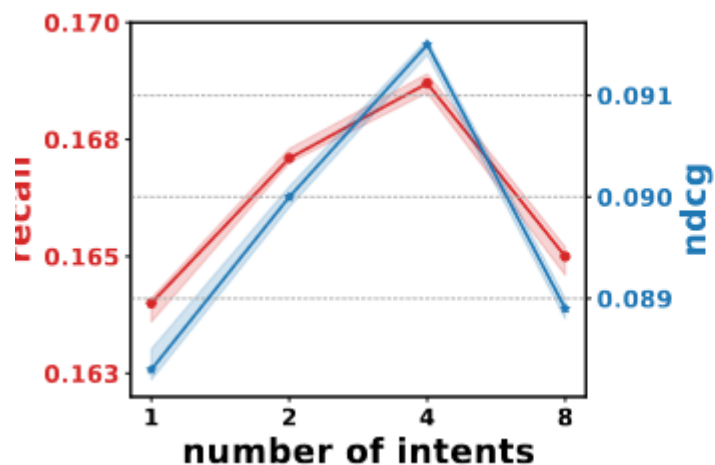
$$\mathbf{e}_u^* = \mathbf{e}_u^{(0)} + \dots + \mathbf{e}_u^{(L)}, \quad \mathbf{e}_i^* = \mathbf{e}_i^{(0)} + \dots + \mathbf{e}_i^{(L)} \quad \hat{y}_{ui} = \mathbf{e}_u^{*\top} \mathbf{e}_i^*$$

$$\mathcal{L}_{\text{BPR}} = \sum_{(u,i,j) \in \mathcal{O}} -\ln \sigma(\hat{y}_{ui} - \hat{y}_{uj}) \quad \mathcal{L}_{\text{KGIN}} = \mathcal{L}_{\text{BPR}} + \lambda_1 \mathcal{L}_{\text{IND}} + \lambda_2 \|\Theta\|_2^2$$

KGIN(WWW'21)

Table 2: Overall performance comparison.

	Amazon-Book		Last-FM		Alibaba-iFashion	
	recall	ndcg	recall	ndcg	recall	ndcg
MF	0.1300	0.0678	0.0724	0.0617	0.1095	0.0670
CKE	0.1342	0.0698	0.0732	0.0630	<u>0.1103</u>	<u>0.0676</u>
KGAT	<u>0.1487</u>	<u>0.0799</u>	0.0873	<u>0.0744</u>	0.1030	0.0627
KGNN-LS	0.1362	0.0560	<u>0.0880</u>	0.0642	0.1039	0.0557
CKAN	0.1442	0.0698	0.0812	0.0660	0.0970	0.0509
R-GCN	0.1220	0.0646	0.0743	0.0631	0.0860	0.0515
KGIN-3	0.1687*	0.0915*	0.0978*	0.0848*	0.1147*	0.0716*
%Imp.	13.44%	14.51%	11.13%	13.97%	3.98%	5.91%



总结

5.

总结

总结

- 如何从有限的数据中获得更多的用户项目信息是推荐系统的核心。
- 基于会话的推荐可以融合用户及项目的历史信息，使推荐更为精准。
- 目前很多的工作关注到利用超图然后将社交网络融入的基本的推荐方式中。
- 经过知识图谱增强的推荐，项目之间丰富的语义相关性有助于探索它们之间的联系，改善项目的表示。
- 但GNN计算复杂度较高，目前面对工业推荐场景，数十亿节点量级的算法还较少。
- 在现实的推荐场景中，节点之间的关系并不总是可靠的，这是一种常见的现象，例如用户不小心点击了项目。此外也可能有攻击者向推荐系统注入虚假数据，现有的基于GNN的推荐很少关注对抗性学习。

敬请批评指正



中国科学院 信息工程研究所
INSTITUTE OF INFORMATION ENGINEERING, CAS