MS-HGAT: Memory-enhanced Sequential Hypergraph Attention Network for Information Diffusion Prediction

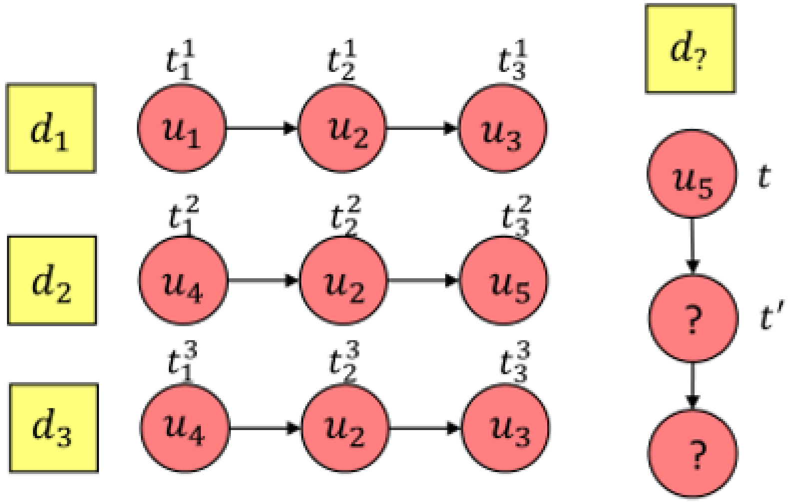
Ling Sun, Yuan Rao*, Xiangbo Zhang, Yuqian Lan, Shuanghe Yu
Xi'an Key Laboratory of Social Intelligence and Complexity Data Processing,
School of Software Engineering, Xi'an Jiaotong University, China
Shaanxi Joint Key Laboratory for Artifact Intelligence, China
{sunling, nick1001, Yuqian_Lan_xjtu, yushuanghe1997}@stu.xjtu.edu.cn, raoyuan@mail.xjtu.edu.cn

https://github.com/slingling/MS-HGAT

AAAI 2022

Reported by Nengqiang Xiang

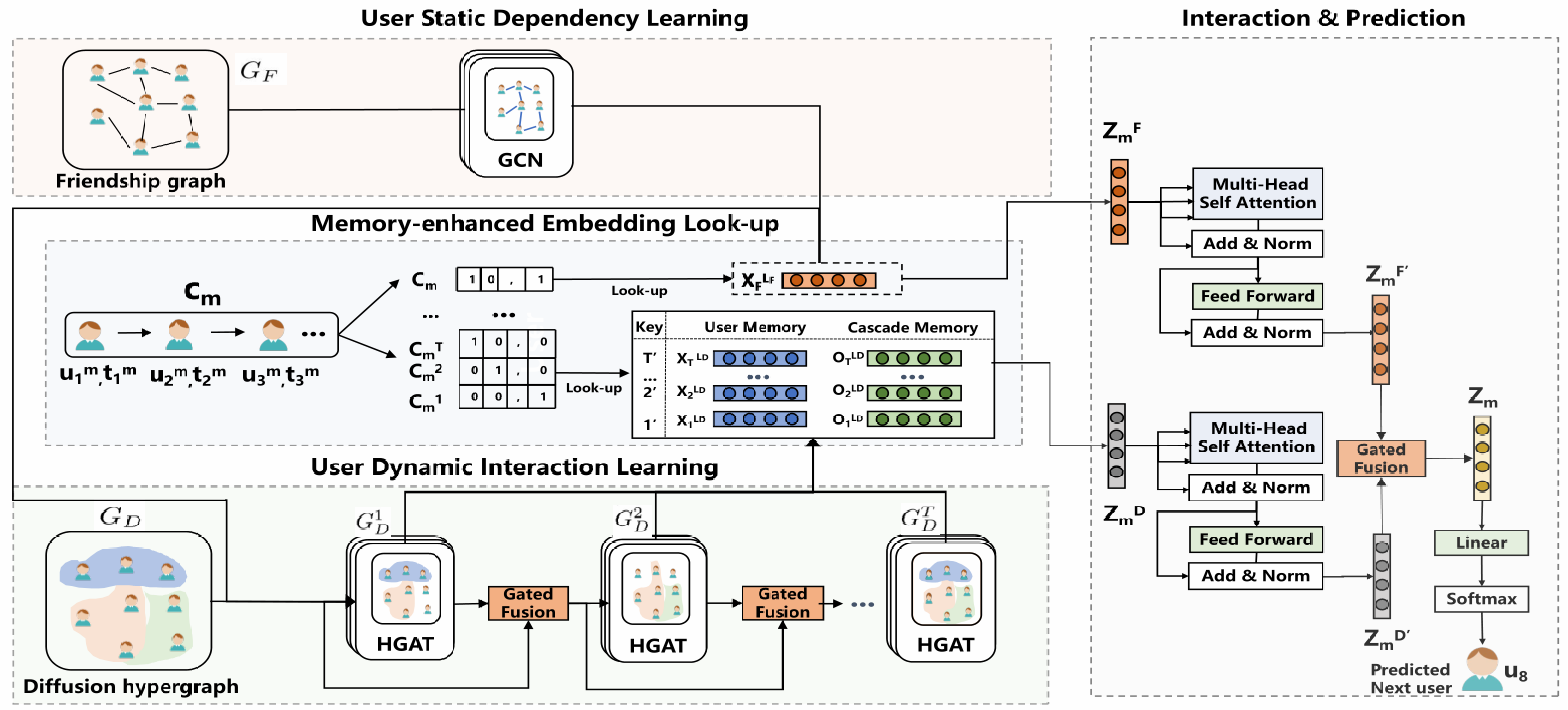
Introduction



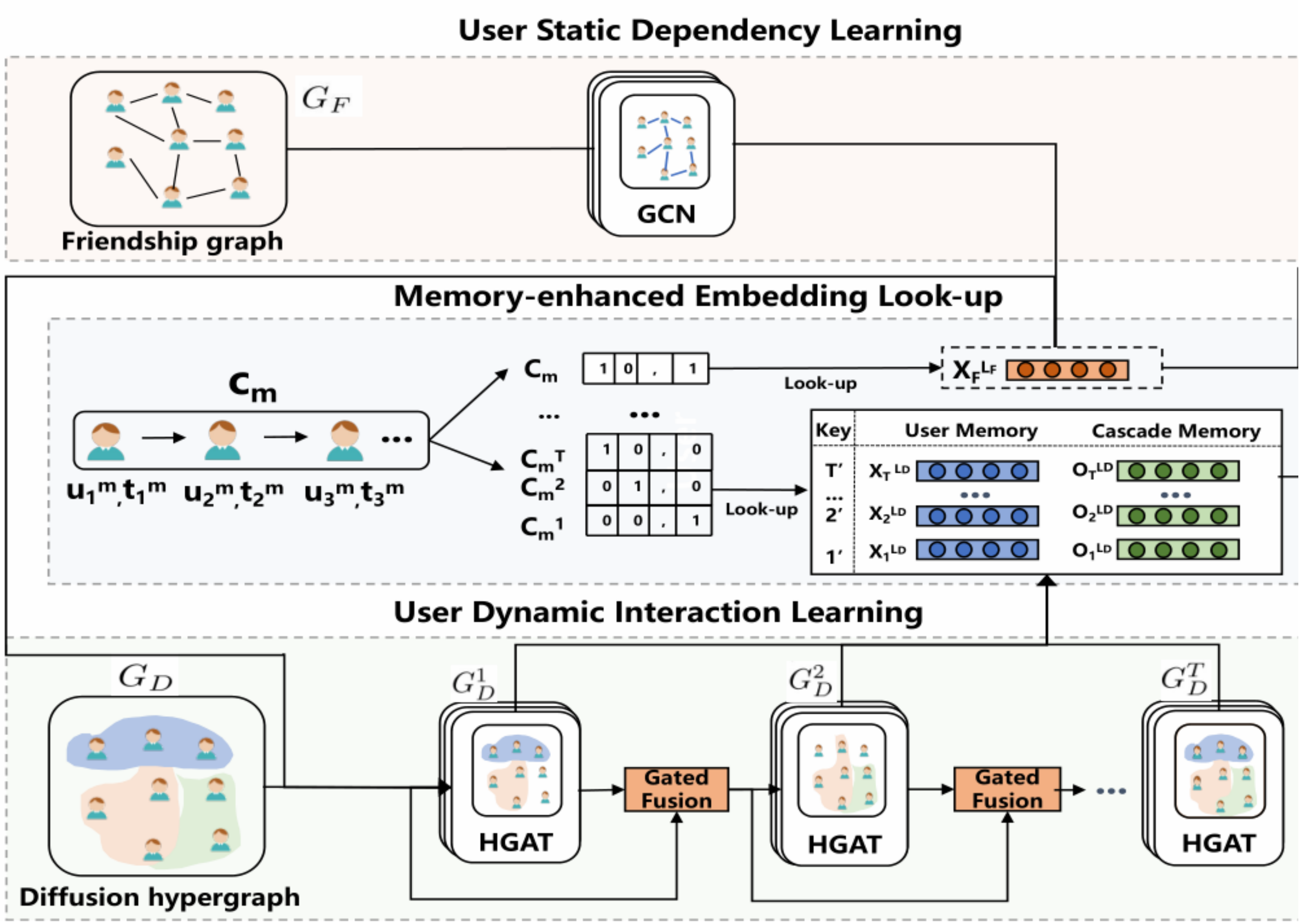
Previous methods usually focus on the order or structure of the infected users in a single cascade, thus ignoring the global dependencies of users and cascades, limiting the performance of prediction.

To address the above issues, this paper propose a novel information diffusion prediction model named Memory-enhanced Sequential Hypergraph Attention Networks (MS-HGAT).

Method


User Static Dependency Learning
Interaction & Prediction
G_F
GCN
Friendship graph
Memory-enhanced Embedding Look-up
C_m
u_1^m, t_1^m
u_2^m, t_2^m
u_3^m, t_3^m
Look-up
X_F^{LF}
Key
User Memory
Cascade Memory
User Dynamic Interaction Learning
G_D
G_D^1
G_D^2
G_D^T
HGAT
Gated Fusion
Diffusion hypergraph
Z_m^F
Multi-Head Self Attention
Add & Norm
Feed Forward
$Z_m^{F'}$
Z_m
Z_m^D
$Z_m^{D'}$
Linear
Softmax
Predicted Next user
u_8

Method



$$U = \{u_1, u_2, ..., u_n\}, |U| = N$$

$$C = \{c_1, c_2, ..., c_M\}, |C| = M$$

$$G_F = (U, E)$$

$$G_D = \{G_D^t | t = 1, 2, ..., T\},\ G_D^t = (U^t, \mathcal{E}^t)$$

$$c_m = \{(u_i^m, t_i^m) | u_i^m \in U\}$$

$$\mathbf{X}_F^{l+1} = \sigma\left(\tilde{\mathbf{D}}_\mathbf{F}^{-\frac{1}{2}} \tilde{\mathbf{A}}_\mathbf{F} \tilde{\mathbf{D}}_\mathbf{F}^{-\frac{1}{2}} \mathbf{X}_F^l \mathbf{W}_F\right) \qquad (1)$$

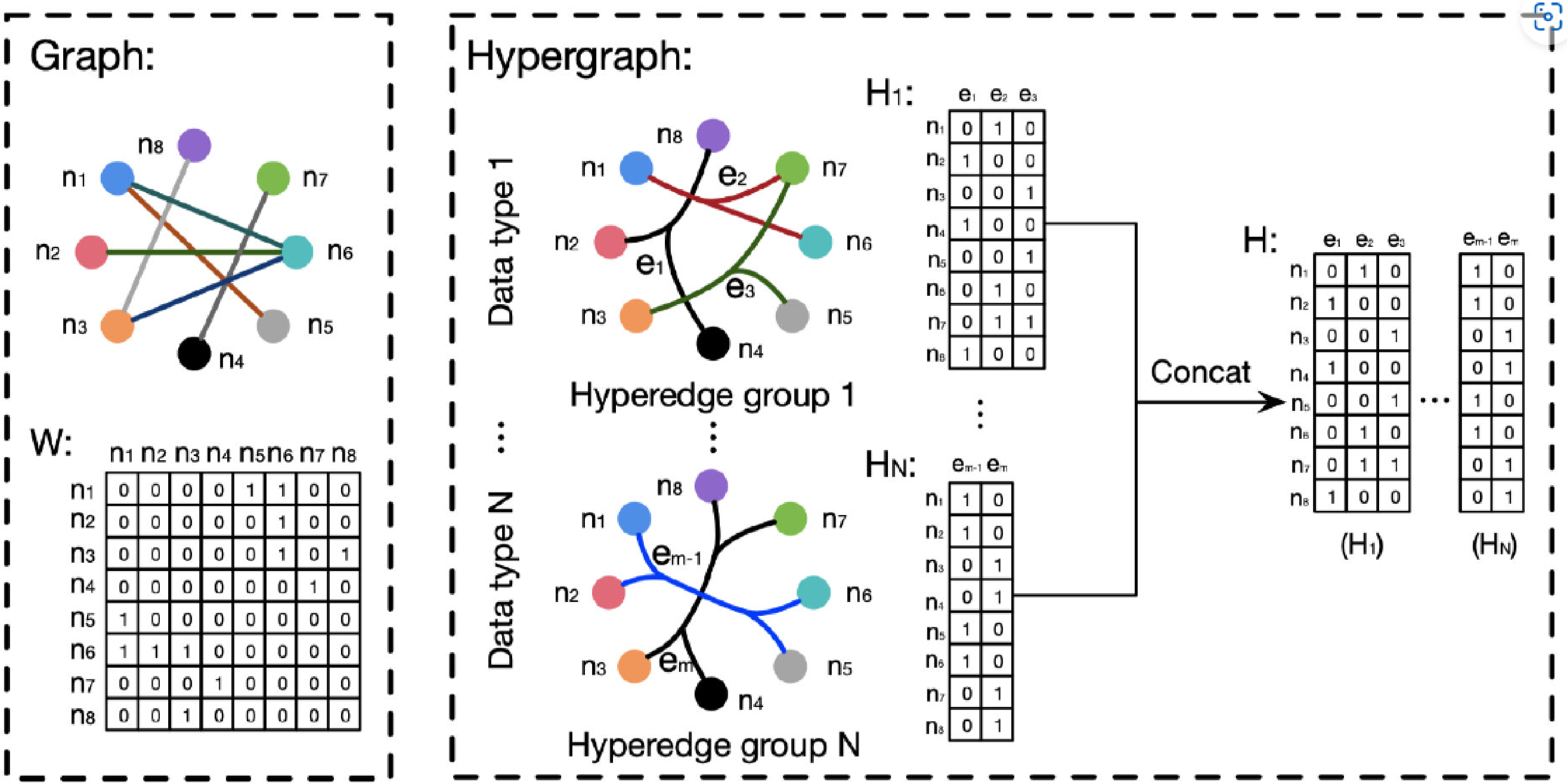
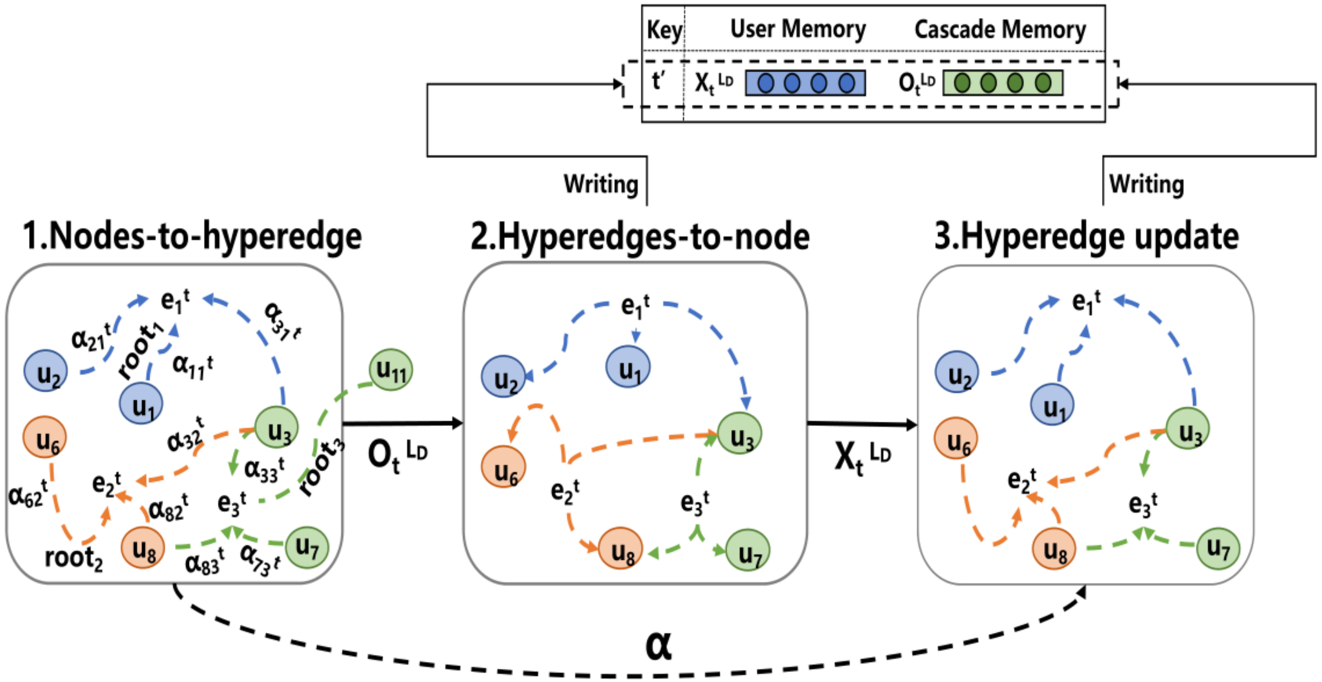


Figure 2: The comparison between graph and hypergraph.

Method



Nodes-to-hyperedge aggregation

$$\mathbf{o}_{j,t}^{l+1} = \sigma(\sum_{u_i^t \in e_j^t} \alpha_{ij}^t \mathbf{W}_1 \mathbf{x}_{i,t}^l) \quad (2)$$

$$\alpha_{ij}^t = \frac{\exp(-\operatorname{dis}(\mathbf{W}_1 \mathbf{x}_{i,t}^l, \mathbf{W}_1 \mathbf{r}_j^l))}{\sum_{u_p^t \in e_j^t} \exp(-\operatorname{dis}(\mathbf{W}_1 \mathbf{x}_{p,t}^l, \mathbf{W}_1 \mathbf{r}_j^l))} \quad (3)$$

Hyperedges-to-node aggregation

$$\mathbf{x}_{i,t}^{l+1} = \sigma(\sum_{e_j^t \in \mathcal{E}_i^t} \mathbf{W}_2 \mathbf{o}_{j,t}^{l+1}) \quad (4)$$

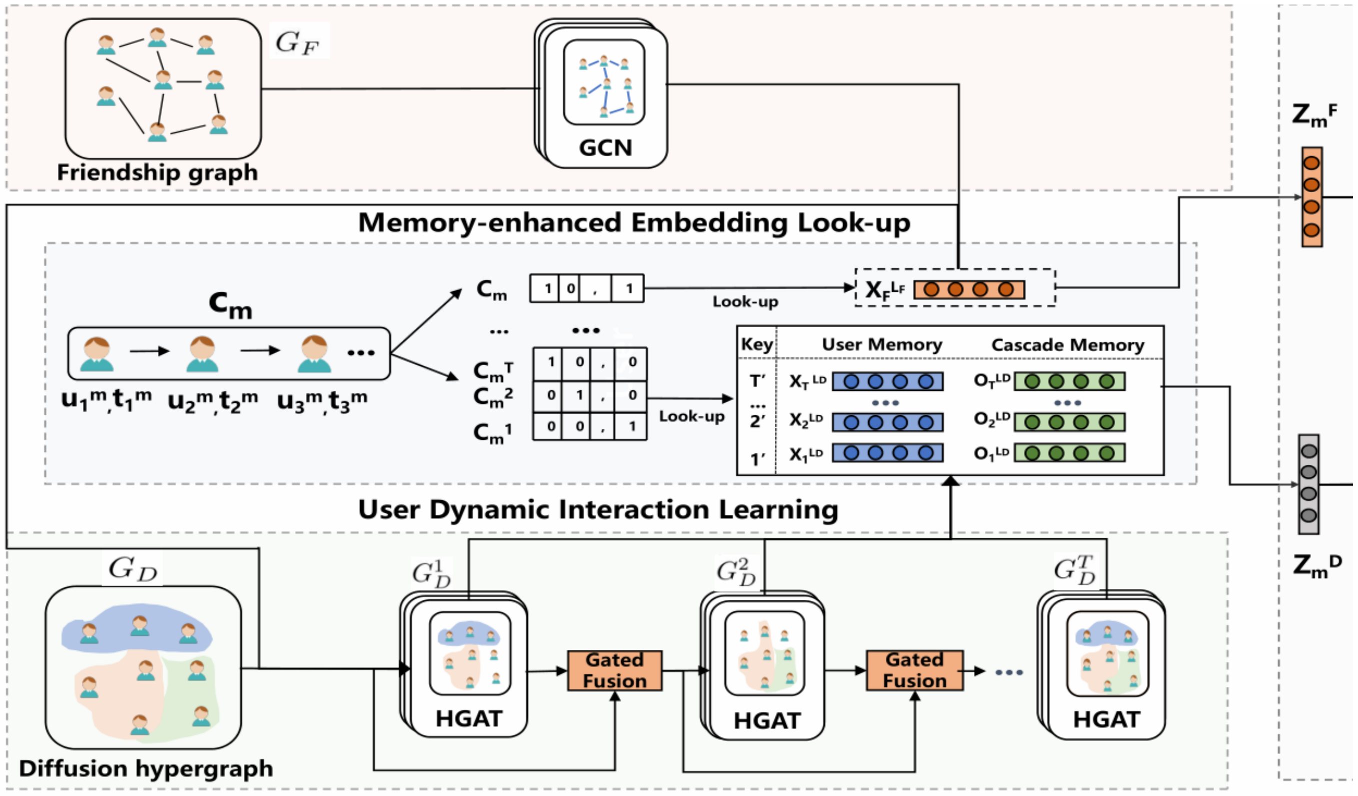
Update of hyperedges

$$\mathbf{o}_{j,t}^{l+1'} = \sigma(\sum_{u_i^t \in e_j^t} \alpha_{ij}^t \mathbf{W}_3 \mathbf{x}_{i,t}^{l+1}) \quad (5)$$

Memory Module

$$M_D = \left\{t' : (X_t^{L_D}, O_t^{L_D})\right\}, t = 1, 2, ..., T \quad (6)$$

Method



Gated Fusion

$$\mathbf{x}^0_{i,t+1} = g_{R_1}\mathbf{x}^{L_D}_{i,t} + (1 - g_{R_1})\mathbf{x}^0_{i,t}$$

$$g_{R_1} = \frac{\exp(\mathbf{W}^T_{Z_1}\sigma(\mathbf{W}_{R_1}\mathbf{x}^{L_D}_{i,t}))}{\exp(\mathbf{W}^T_{Z_1}\sigma(\mathbf{W}_{R_1}\mathbf{x}^{L_D}_{i,t}))+\exp(\mathbf{W}^T_{Z_1}\sigma(\mathbf{W}_{R_1}\mathbf{x}^0_{i,t}))} \tag{7}$$

Memory-enhanced Embedding Look-up

$$Z^F_m = [(x_i)] \in \mathbb{R}^{|c_m|\times d}$$

$$t^m_i \geq t' \text{ and } t^m_i < (t+1)'$$

$$q^D_m = [(x_{i,t})] \in \mathbb{R}^{|c_m|\times d}$$

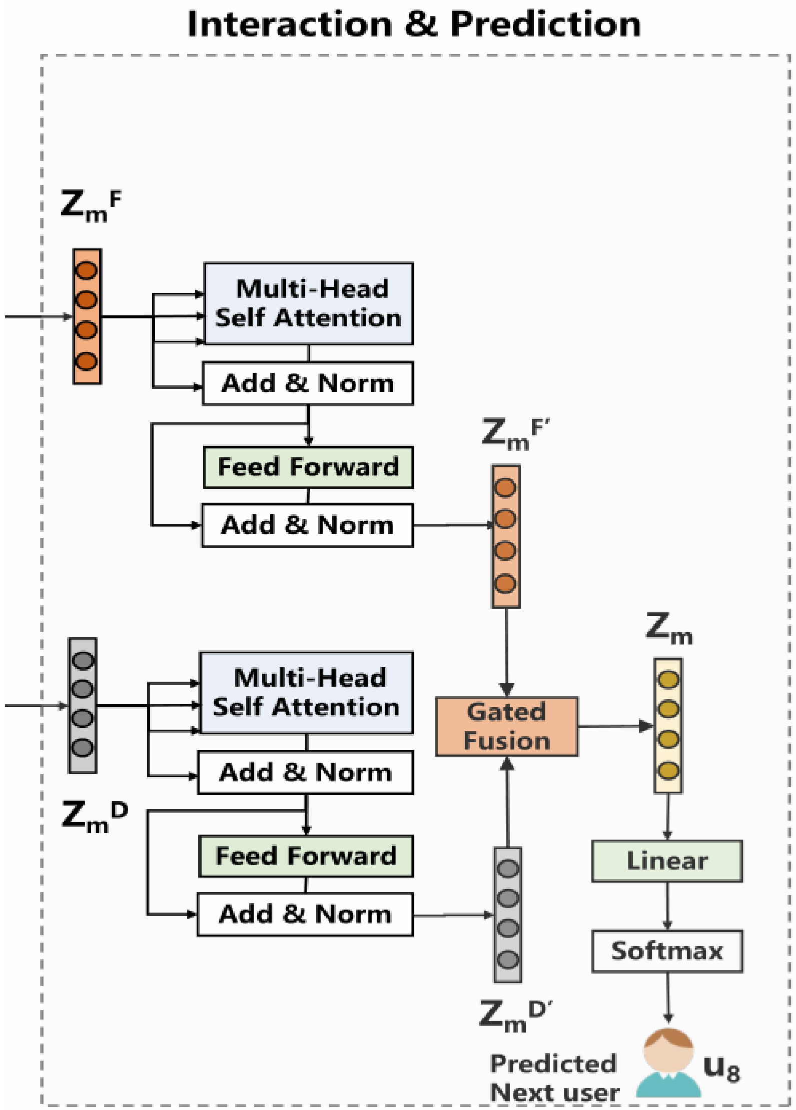
$$p^D_m = [(o_{m,t})] \in \mathbb{R}^{|c_m|\times d}$$

Fusion Layer

$$\mathbf{Z}^D_m = g_{R_2}\mathbf{p}^D_m + (1 - g_{R_2})\mathbf{q}^D_m$$

$$g_{R_2} = \frac{\exp(\mathbf{W}^T_{Z_2}\sigma(\mathbf{W}_{R_2}\mathbf{p}^D_m))}{\exp(\mathbf{W}^T_{Z_2}\sigma(\mathbf{W}_{R_2}\mathbf{p}^D_m))+\exp(\mathbf{W}^T_{Z_2}\sigma(\mathbf{W}_{R_2}\mathbf{q}^D_m))} \tag{8}$$

Method



$$\mathrm{Att}(\mathbf{Q},\mathbf{K},\mathbf{V}) = \mathrm{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d'}} + \mathbf{M}\right)\mathbf{V}$$

$$\mathbf{h}_{i,m}^F = \mathrm{Att}\left(\mathbf{Z}_m^F\mathbf{W}_i^Q, \mathbf{Z}_m^F\mathbf{W}_i^K, \mathbf{Z}_m^F\mathbf{W}_i^V\right) \quad (9)$$

$$\mathbf{h}_m^F = \left[\mathbf{h}_{1,m}^F; \mathbf{h}_{2,m}^F; \ldots; \mathbf{h}_{H,m}^F\right]\mathbf{W}^O$$

$$\mathbf{Z}_m^{F'} = \mathbf{ReLU}\left(\mathbf{h}_m^F\mathbf{W}_{A_1} + \mathbf{b}_1\right)\mathbf{W}_{A_2} + \mathbf{b}_2 \quad (10)$$

$$\mathbf{Z}_m = g_{R_3}\mathbf{Z}_m^{D'} + (1 - g_{R_3})\mathbf{Z}_m^{F'}$$

$$g_{R_3} = \frac{\exp(\mathbf{W}_{Z_3}^T\sigma(\mathbf{W}_{R_3}\mathbf{Z}_m^{D'}))}{\exp(\mathbf{W}_{Z_3}^T\sigma(\mathbf{W}_{R_3}\mathbf{Z}_m^{D'})) + \exp(\mathbf{W}_{Z_3}^T\sigma(\mathbf{W}_{R_3}\mathbf{Z}_m^{F'}))} \quad (11)$$

$$\hat{y} = \mathrm{softmax}(\mathbf{W}_p\mathbf{Z}_m + \mathbf{Mask}_m)$$

$$\mathcal{J}(\theta) = -\sum_{j=2}^{|c_m|}\sum_{i=1}^{|U|}\mathbf{y}_{ji}\log\left(\hat{\mathbf{y}}_{ji}\right) \quad (13)$$

Experiments

Table 1: Statistics of datasets used in our experiments

Datasets	**Twitter**	**Douban**	**Android**	**Christ.**
# Users	12,627	12,232	9,958	2,897
		Frienship		
# Links	309,631	396,580	48,573	35,624
Density	24.52	30.21	4.87	12.30
		Interaction		
# Cascades	3,442	3,475	679	589
Avg. Length	32.60	21.76	33.3	22.9
Density	8.89	6.18	2.27	4.66

Experiments

Table 2: Experimental results on 4 dataset (%) (Hits@k scores for $K = 10, 50, 100$), scores are the higher the better.

Models	Twitter			Douban			Android			Christianity		
	@10	@50	@100	@10	@50	@100	@10	@50	@100	@10	@50	@100
DeepDiffuse	5.79	10.80	18.39	9.02	14.93	19.13	4.13	10.58	17.21	10.27	21.83	30.74
Topo-LSTM	8.45	15.80	25.42	8.57	16.53	21.47	4.56	12.63	16.53	12.28	22.63	31.52
NDM	15.21	28.23	32.30	10.00	21.13	30.14	4.85	14.24	18.97	15.41	31.36	45.86
SNIDSA	25.37	36.64	42.89	16.23	27.24	35.59	5.63	15.22	20.93	17.74	34.58	48.76
FOREST	28.67	42.07	49.75	19.50	32.03	39.08	9.68	17.73	24.08	24.85	42.01	51.28
Inf-VAE	14.85	32.72	45.72	8.94	22.02	35.72	5.98	14.70	20.91	18.38	38.50	51.05
DyHGCN	31.88	45.05	52.19	18.71	32.33	39.71	9.10	16.38	23.09	26.62	42.80	52.47
MS-HGAT (ours)	**33.50**	**49.59**	**58.91**	**21.33**	**35.25**	**42.75**	**10.41**	**20.31**	**27.55**	**28.80**	**47.14**	**55.62**
NOW	**35.83**	**52.25**	**59.96**	**22.42**	**37.41**	**44.39**	**11.52**	**21.56**	**28.24**	**28.99**	**48. 52**	**55.81**

Experiments

Table 3: Experimental results on 4 dataset (%) (MAP@k scores for $K = 10, 50, 100$), scores are the higher the better.

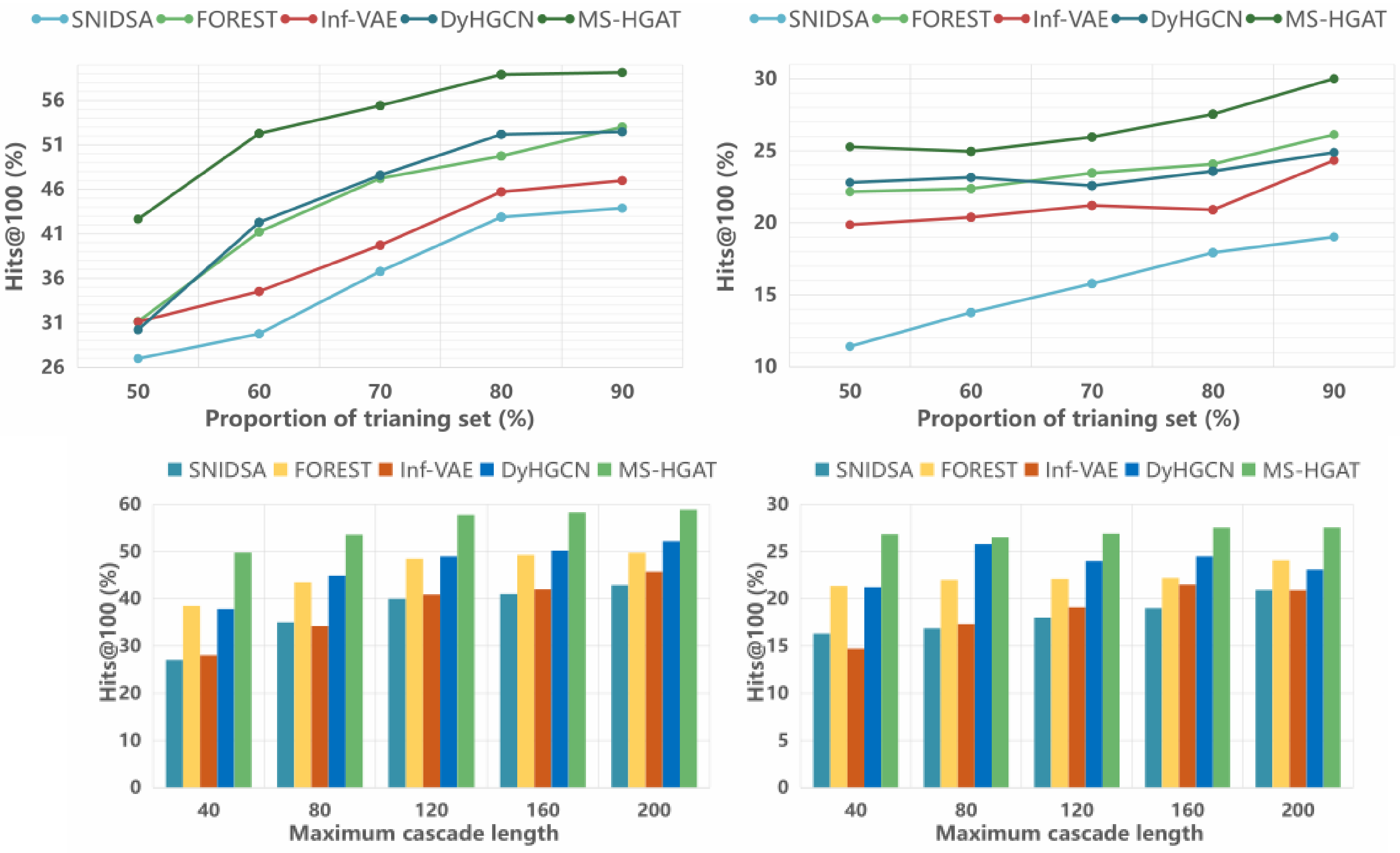
Models	Twitter			Douban			Android			Christianity		
	@10	@50	@100	@10	@50	@100	@10	@50	@100	@10	@50	@100
DeepDiffuse	5.87	6.80	6.39	6.02	6.93	7.13	2.30	2.53	2.56	7.27	7.83	7.84
Topo-LSTM	8.51	12.68	13.68	6.57	7.53	7.78	3.60	4.05	4.06	7.93	8.67	9.86
NDM	12.41	13.23	14.30	8.24	8.73	9.14	2.01	2.22	2.93	7.41	7.68	7.86
SNIDSA	15.34	16.64	16.89	10.02	11.24	11.59	2.98	3.24	3.97	8.69	8.94	9.72
FOREST	19.60	20.21	21.75	11.26	11.84	11.94	5.83	6.17	6.26	14.64	15.45	15.58
Inf-VAE	19.80	20.66	21.32	11.02	11.28	12.28	4.82	4.86	5.27	9.25	11.96	12.45
DyHGCN	20.87	21.48	21.58	10.61	11.26	11.36	6.09	6.40	6.50	15.64	16.30	16.44
MS-HGAT (ours)	**22.49**	**23.17**	**23.30**	**11.72**	**12.52**	**12.60**	**6.39**	**6.87**	**6.96**	**17.44**	**18.27**	**18.40**
NOW	**22.78**	**23.24**	**23. 47**	**12.76**	**13.46**	**13.57**	**6.79**	**7.23**	**7.34**	**17.91**	**18.74**	**18.84**

Experiments

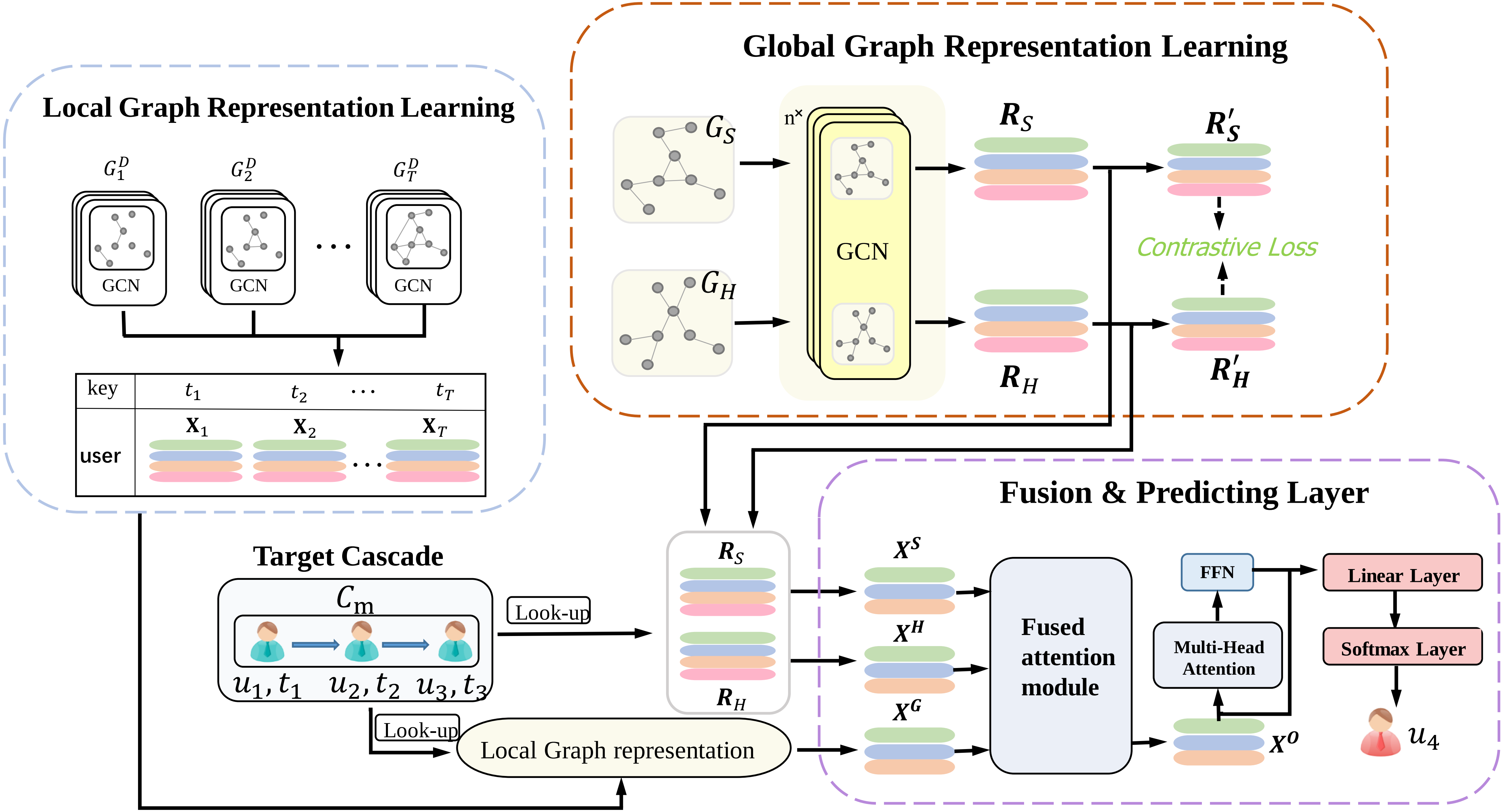
Table 4: Ablation study of MS-HGAT.

Models	Twitter		Android	
	Hits@100	**MAP@100**	**Hits@100**	**MAP@100**
MS-HGAT	58.91	**23.30**	**27.55**	**6.96**
w/o FG	57.20	21.38	26.32	6.86
w/o DH	57.41	22.24	26.74	6.78
w/o UM	58.63	22.74	26.40	6.83
w/o CM	58.32	21.96	27.09	6.77
w/o ATTH	**58.95**	22.76	27.03	6.75
w/o GF	57.93	22.19	27.26	6.89

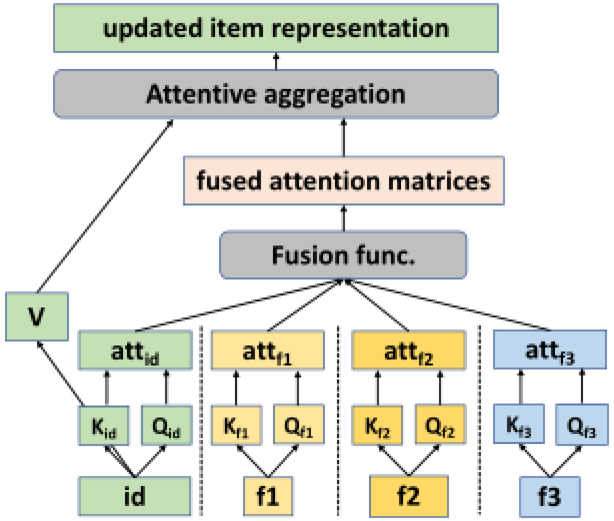
Experiments



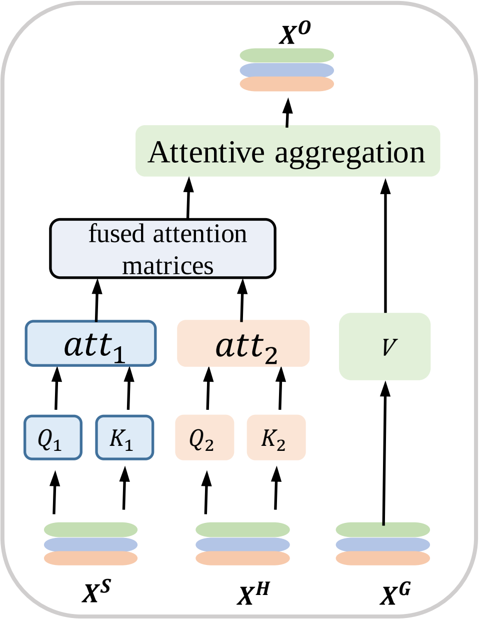
(a) Twitter (b) Android


Local Graph Representation Learning
G_1^D
G_2^D
G_T^D
GCN
GCN
GCN
key
t_1
t_2
t_T
user
$\mathbf{X}_1$
$\mathbf{X}_2$
$\mathbf{X}_T$
Global Graph Representation Learning
G_S
G_H
n×
GCN
R_S
R_H
R'_S
R'_H
Contrastive Loss
Target Cascade
C_m
u_1, t_1
u_2, t_2
u_3, t_3
Look-up
Look-up
Local Graph representation
R_S
R_H
Fusion & Predicting Layer
X^S
X^H
X^G
Fused attention module
FFN
Multi-Head Attention
X^O
Linear Layer
Softmax Layer
u_4

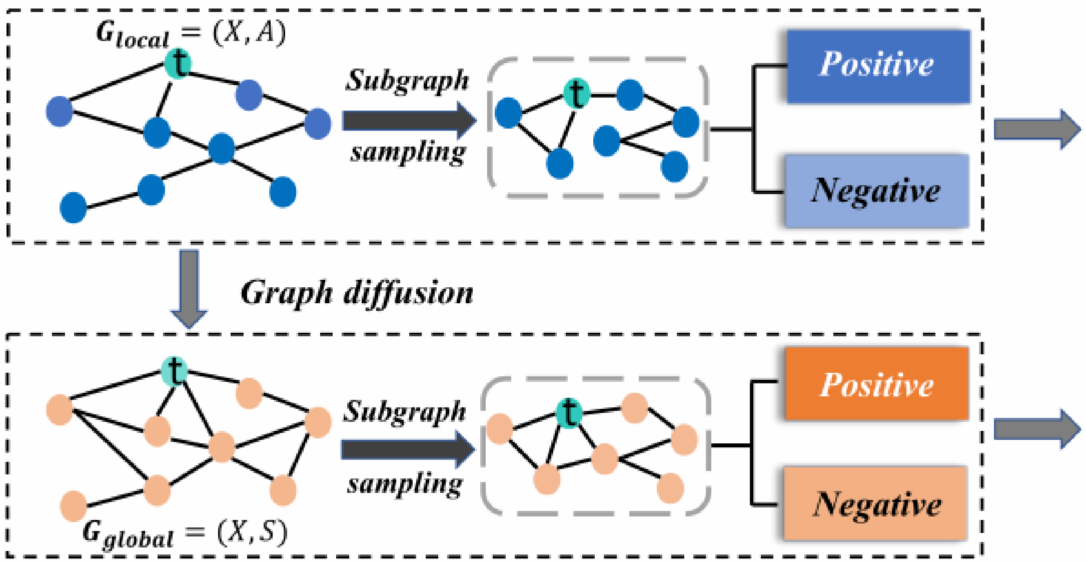
SIGIR’22 Decoupled Side Information Fusion for Sequential Recommendation



(c) DIF-SR.



2022_IJCAI_Reconstruction Enhanced Multi-View Contrastive Learning for Anomaly Detection on Attributed Networks

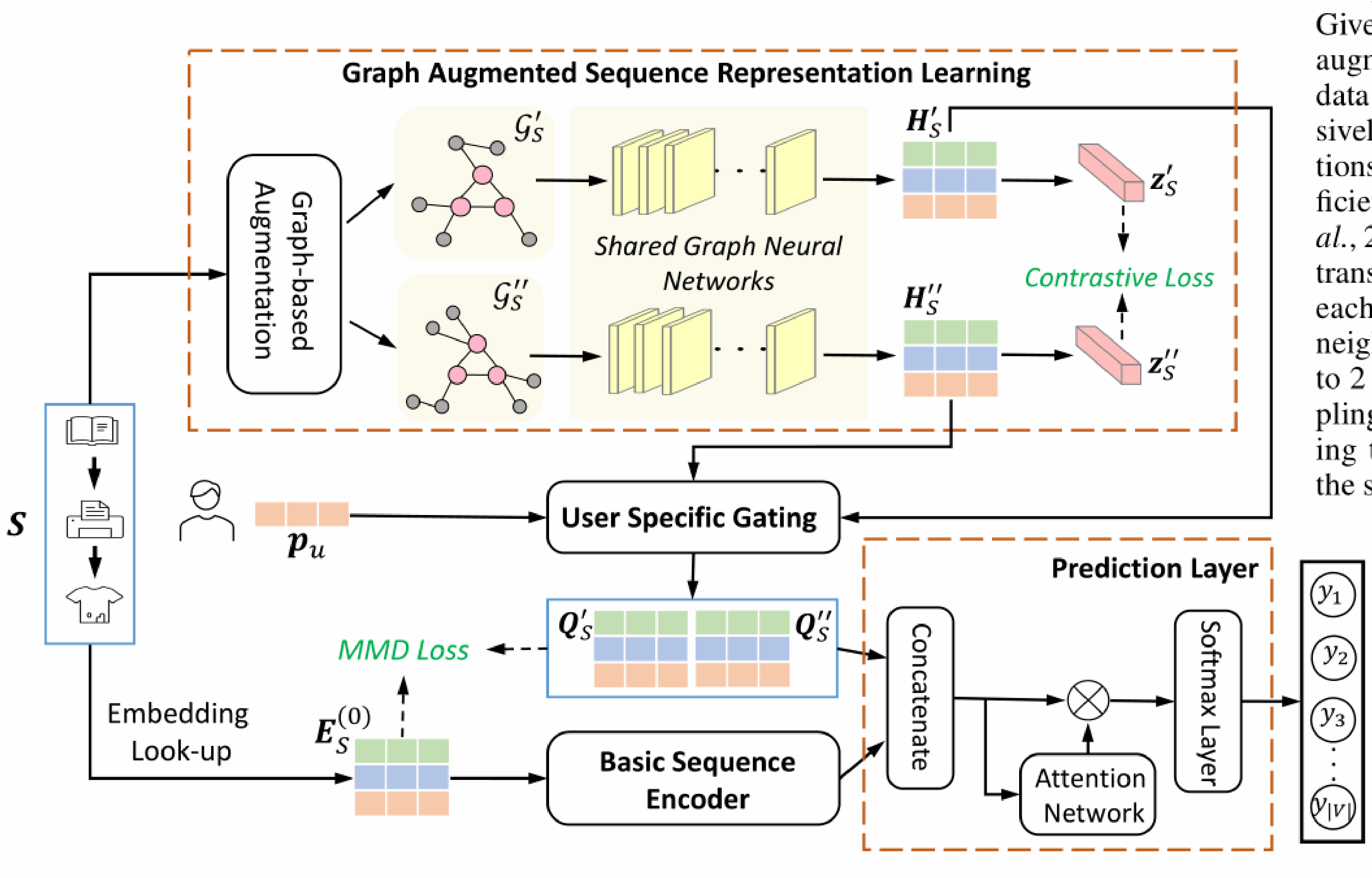


$$\mathbf{S} = \sum_{k=0}^{\infty} \theta_k \mathbf{T}^k \in \mathbb{R}^{N \times N} \quad (1)$$

where θ_k is the weighting coefficient to control the proportion of local and global structure information and $\mathbf{T} \in \mathbb{R}^{N \times N}$ denotes the generalized transition matrix to transfer the adjacency matrix. Note that $\theta_k \in [0, 1]$ and $\sum_{k=0}^{\infty} \theta_k = 1$. In this paper, Personalized PageRank (PPR) [Page *et al.*, 1999] is adopted to power the graph diffusion. Given the adjacency matrix $\mathbf{A} \in \mathbb{R}^{N \times N}$, the identity matrix $\mathbf{I}$ and its degree matrix $\mathbf{D}$, the transition matrix and the weight can be formulated respectively as $\mathbf{T} = \mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{-1/2}$ and $\theta_k = \alpha(1-\alpha)^k$. Then the graph diffusion $\mathbf{S}$ can be reformulated as:

$$\mathbf{S} = \alpha(\mathbf{I} - (1-\alpha)\mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{-1/2})^{-1} \quad (2)$$

2022_IJCAI_Enhancing Sequential Recommendation with Graph Contrastive Learning



Graph-based Augmentation

Given the weighted transition graph $\mathcal{G}$, we first construct two augmented graph views for an interaction sequence S through data augmentation. The motivation is to create comprehensively and realistically rational data via certain transformations on the original sequence. In this work, we use the efficient neighborhood sampling method used in [Hamilton *et al.*, 2017] to generate the augmented graph views from a large transition graph for a given sequence. Specifically, we treat each node $v \in S$ as a central node and interatively sample its neighbors in $\mathcal{G}$ by empirically setting the sampling depth M to 2 and the sampling size N at each step to 20. In the sampling process, we uniformly sample nodes without considering the edge weights, and then preserve the edges between the sampled nodes and their weights in $\mathcal{G}$. For a particular se-

- **Random Crop** (crop)**:** It randomly selects a continuous sub-sequence from positions i to $i + l_c$ from s_u and removes it. l_c is defined by $l_c = i + \lfloor \mu_c \cdot |s_u| \rfloor$ where μ_c $(0 < \mu_c \leq 1)$ is a hyper-parameter. The cropped sequence is defined by: $s_u^{c} = [v_i^u, v_{i+1}^u, \ldots, v_{i+l_c}^u]$.

- **Random Mask** (mask)**:** It randomly selects a proportion μ_m of items from s_u to be masked. Let $g^m(1), g^m(2), \cdots, g^m(n_u^m)$ be the indexes of the items to be masked where $n_u^m = \lfloor \mu_m \cdot |s_u| \rfloor$ and $g^m(x) \in [1, |s_u|]$. An item v_i is replaced with the mask item [m] if selected to be masked. The masked sequence is thus: $s_u^{\text{mask}} = [v_1^u, \cdots, v_{g^m(1)-1}^u, [\text{m}], v_{g^m(1)+1}^u, \cdots, v_{g^m(n_u^m)-1}^u, [\text{m}], v_{g^m(n_u^m)+1}^u, \ldots, v_{|s_u|}^u]$.

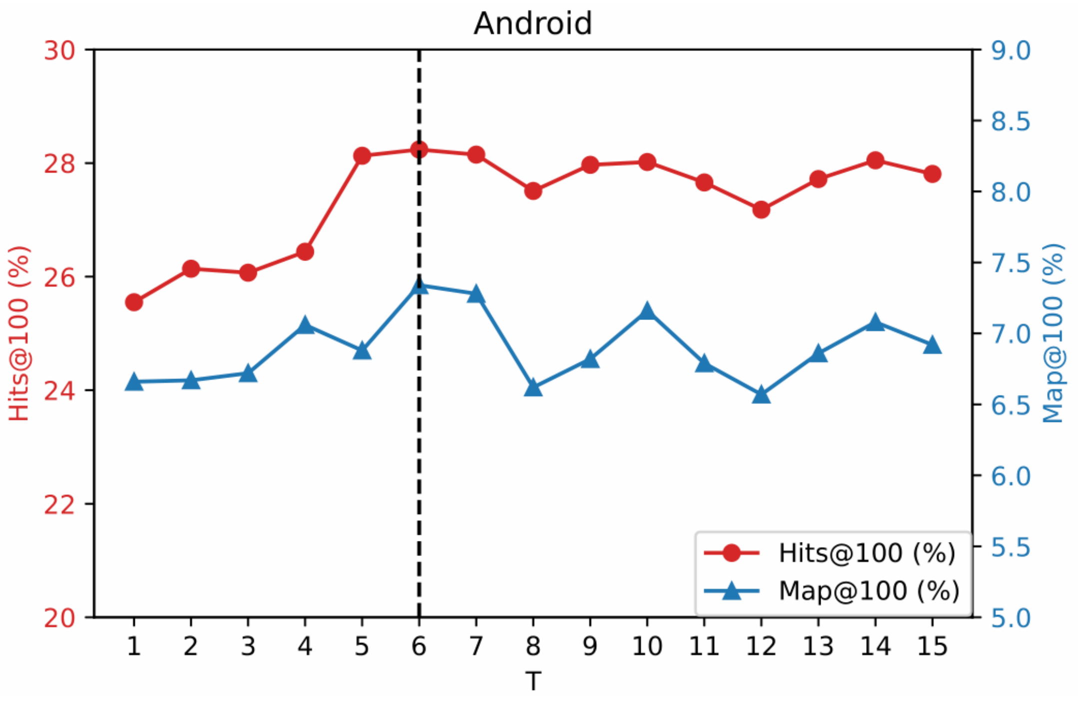
- **Random Reorder** (rord)**:** It first randomly selects a continuous sub-sequence $[v_i^u, v_{i+1}^u, \ldots, v_{i+l_r}^u]$ of length $l_r = \lfloor \mu_r * |s_u| \rfloor$ $(0 \leq \mu_r \leq 1)$. It then randomly shuffles the items in the sub-sequence. Suppose the reordered items, sorted by new positions, are $[\tilde{v}_i^u, \ldots, \tilde{v}_{i+l_r}^u]$. The reordered sequence is thus: $s_u^{\text{rord}} = [v_1^u, \cdots, v_{i-1}^u, \tilde{v}_i^u, \tilde{v}_{i+1}^u, \cdots, \tilde{v}_{i+l_r}^u, v_{i+l_r+1}^u, \cdots v_{|s_u|}^u]$.

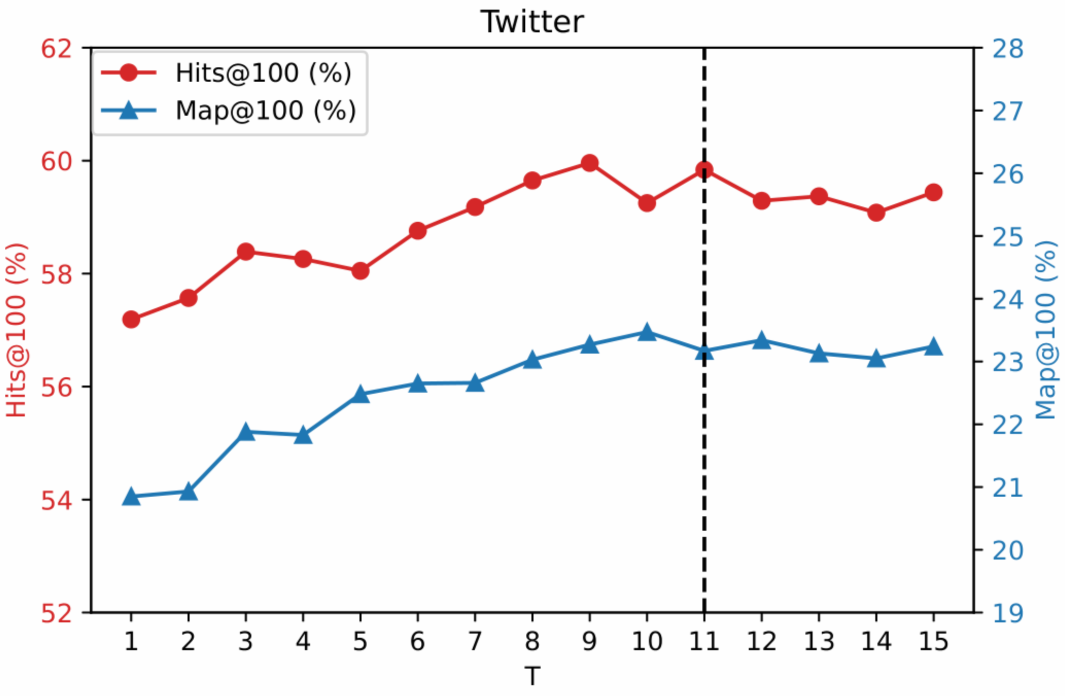
- **Random Retrieval** (rtrl)**:** This operation randomly selects another user sequence $s_{u'}$ that shares the same target (or next) item as the input sequence s_u, i.e., $v_*^u = v_*^{u'}$. The retrieved sequence is thus: $s_u^{\text{rtrl}} = s_{u'}, s.t.\ v_*^u = v_*^{u'}$

Experiments

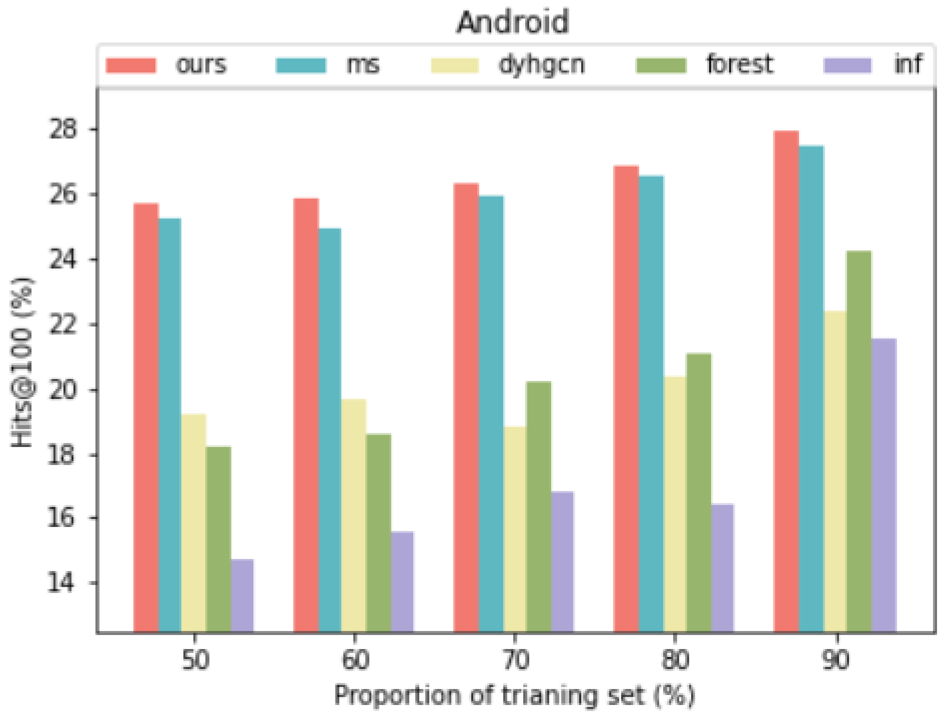
	Twitter		Douban		Android		Chris	
指标	hits@100	map@100	hits@100	map@100	hits@100	map@100	hits@100	map@100
C-L	59.33	23.24	44.18	13.46	27.40	6.83	54.24	17.50
Social	58.68	22.19	43.49	12.63	28.05	7.06	54.83	18.21
H-G	59.77	23.05	44.32	13.04	28.13	7.08	54.36	17.74
D-G	59.42	23.17	44.15	13.43	27.17	6.91	55.22	18.17
FA	59.01	22.67	43.26	12.77	27.47	6.92	55.31	18.21
ALL	**59.96**	**23.47**	**44.39**	**13.57**	**28.24**	**7.34**	**55.81**	**18.84**

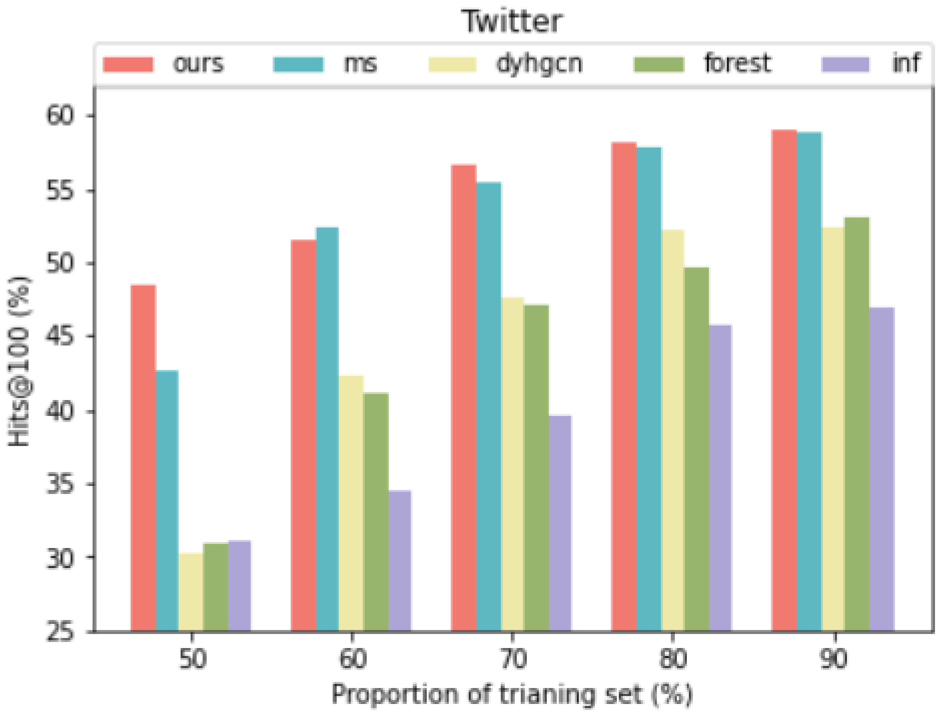
Experiments





Experiments

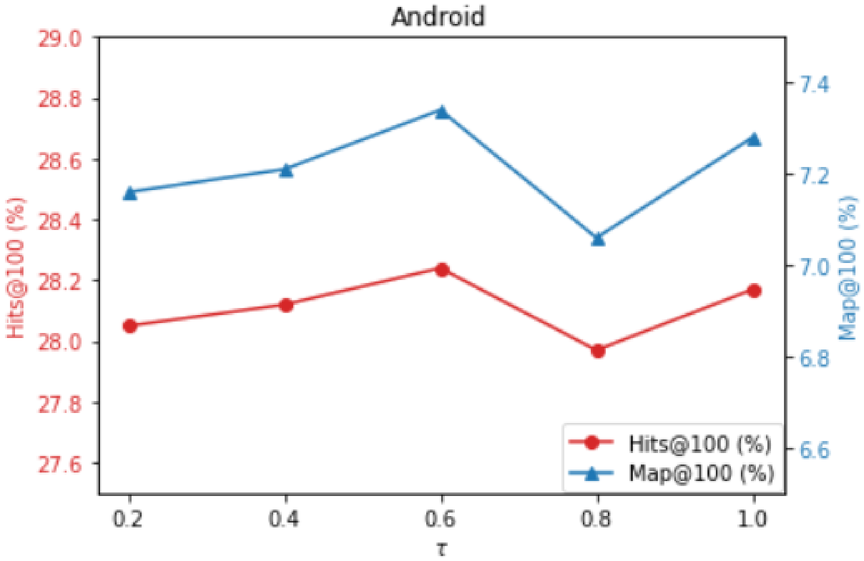


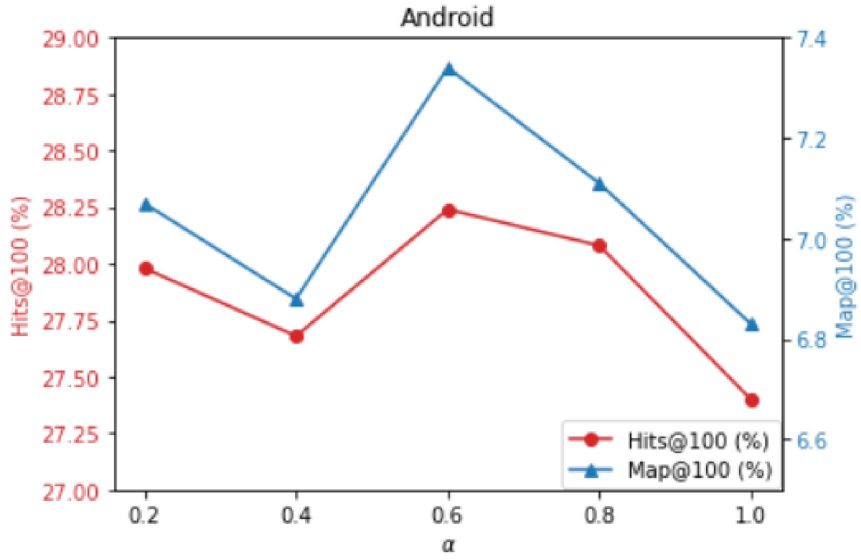


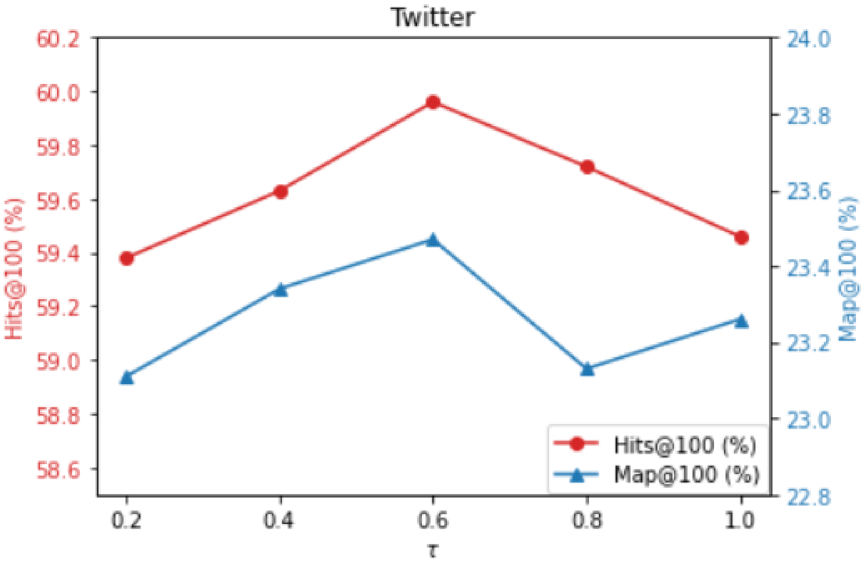
Experiments

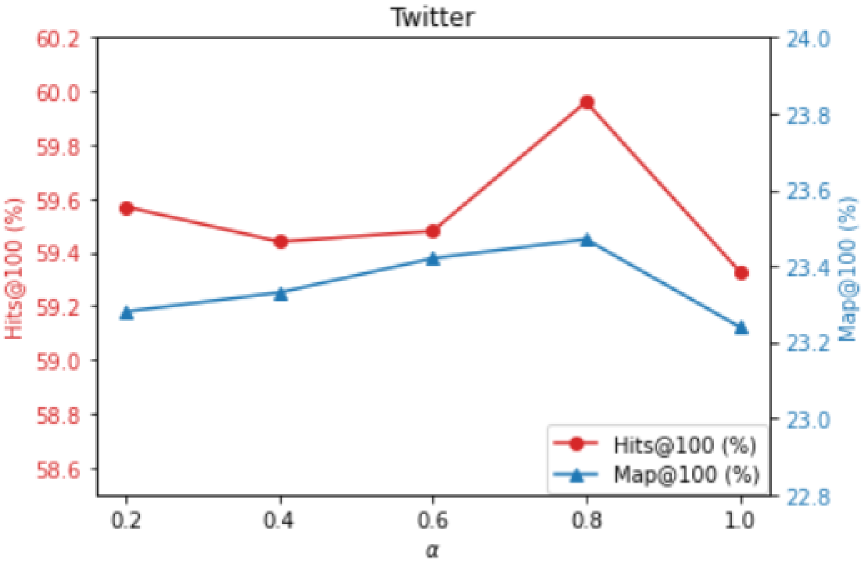
depth	android		twitter		Christianity		douban	
	hits@100	map@100	hits@100	map@100	hits@100	map@100	hits@100	map@100
l=0	27.85	7.06	58.18	22.37	55.17	18.09	43.56	12.64
l=1	**28.24**	**7.34**	59.53	23.11	**55.81**	**18.84**	44.24	13.41
l=2	28.17	7.32	**59.96**	**23.47**	55.42	18.42	**44.39**	**13.57**
l=3	28.06	7.21	59.68	23.25	55.22	18.23	44.31	13.45

Experiments









Experiments