RotDiff: A Hyperbolic Rotation Representation Model for Information Diffusion Prediction

Hongliang Qiao*
Harbin Institute of Technology
Shenzhen, China
21s151112@stu.hit.edu.cn

Shanshan Feng*
Wecar Technology Co., Ltd.
Shenzhen, China
victor_fengss@foxmail.com

Xutao Li†
Harbin Institute of Technology
Shenzhen, China
lixutao@hit.edu.cn

Huiwei Lin
Harbin Institute of Technology
Shenzhen, China
linhuiwei@stu.hit.edu.cn

Han Hu
Beijing Institute of Technology
Beijing, China
hhu@bit.edu.cn

Wei Wei
Huazhong University of Science and Technology
Wuhan, China
weiw@hust.edu.cn

Yunming Ye
Harbin Institute of Technology
Shenzhen, China
yeyunming@hit.edu.cn

https://github.com/PlaymakerQ/RotDiff

—— CIKM 2023

Reported by Yuyang Lai

1.Introduction

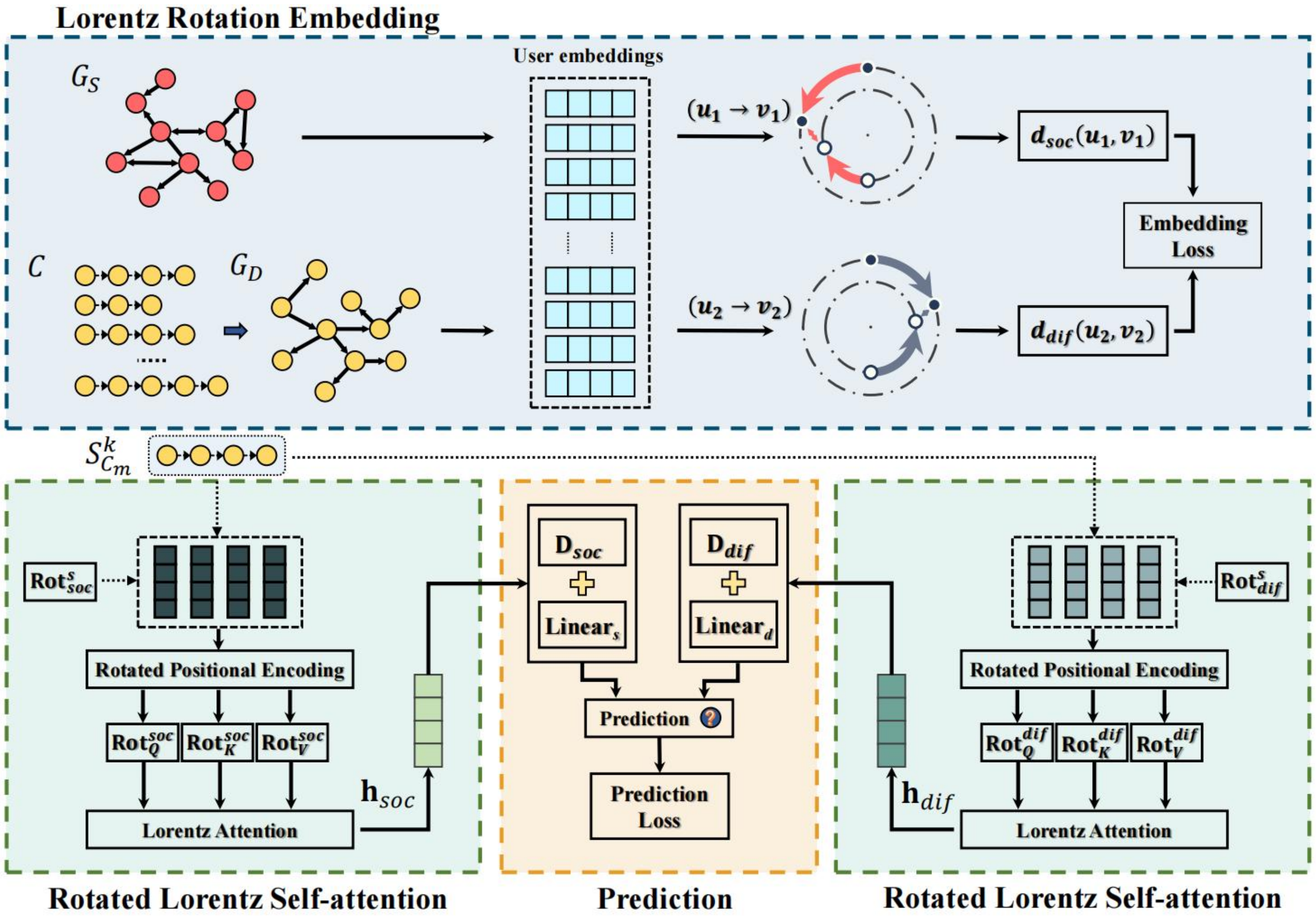
2.Method

3.Experiments

Introduction

Existing methods are mainly based on Euclidean space, which cannot well preserve the underlying **hierarchical structures** that could better reflect the strength of user influence.

Meanwhile, existing methods cannot accurately model the obvious **asymmetric features** of the diffusion process.


Lorentz Rotation Embedding
User embeddings
G_S
C
G_D
$(u_1 \rightarrow v_1)$
$(u_2 \rightarrow v_2)$
$d_{soc}(u_1, v_1)$
$d_{dif}(u_2, v_2)$
Embedding Loss
$S^k_{C_m}$
Rot^s_{soc}
Rot^s_{dif}
Rotated Positional Encoding
Rot^{soc}_Q Rot^{soc}_K Rot^{soc}_V
Rot^{dif}_Q Rot^{dif}_K Rot^{dif}_V
Lorentz Attention
h_{soc}
h_{dif}
D_{soc}
D_{dif}
$Linear_s$
$Linear_d$
Prediction
Prediction Loss
Rotated Lorentz Self-attention
Prediction
Rotated Lorentz Self-attention

Method

Problem Definition

$$C_m = \{(u_1^m, t_1^m), (u_2^m, t_2^m), \cdots, (u_{L_m}^m, t_{L_m}^m)\}$$

$$G_S = (V_S, E_S)$$

$$G_D = (V_D, E_D)$$

$$S_{C_m}^k = \{u_1, \cdots, u_k\}$$

Hyperbolic Geometry

$$\mathcal{L}_\gamma^d = (\mathbb{L}_\gamma^d, g_\mathbf{x}^\gamma)$$

$$\mathbb{L}_\gamma^d = \{\mathbf{x} \in \mathbb{R}^{d+1} : \langle \mathbf{x}, \mathbf{x} \rangle_\mathcal{L} = -\gamma\}$$

$$\mathbf{x} = (x_0, x_1, \cdots, x_d) \quad x_0 = \sqrt{\gamma + \sum_{i=1}^d x_i^2} > 0$$

$$\langle \mathbf{x}, \mathbf{y} \rangle_\mathcal{L} = -x_0 \cdot y_0 + \sum_{i=1}^{d} x_i \cdot y_i, \tag{1}$$

$$||\mathbf{x}||_\mathcal{L} = \sqrt{\langle \mathbf{x}, \mathbf{x} \rangle_\mathcal{L}}$$

$$\langle \mathbf{x}, \mathbf{y} \rangle_\mathcal{L} \leqslant -\gamma$$

$$g_\mathbf{x}^\gamma = \mathrm{diag}([-1, 1, 1, \cdots, 1])$$

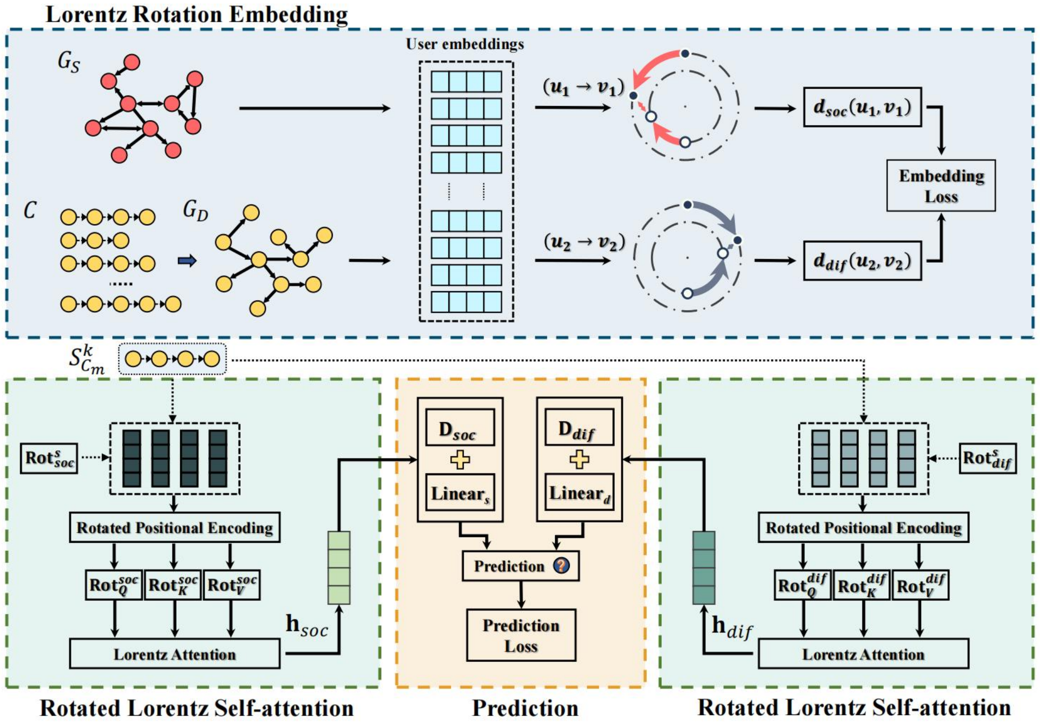
$$D_{SL}(\mathbf{x}, \mathbf{y}) = -2\gamma - 2\langle \mathbf{x}, \mathbf{y} \rangle_\mathcal{L}. \tag{2}$$

$$\mathbf{Rot}_\Theta = \begin{bmatrix} \mathbf{R}(\theta_{r,1}) & & & \\ & \mathbf{R}(\theta_{r,2}) & & \\ & & \cdots & \\ & & & \mathbf{R}(\theta_{r,d/2}) \end{bmatrix}, \tag{3}$$

$\mathbf{Rot} \in \mathbb{R}^{d \times d}$

$$\mathbf{R}(\theta_{r,i}) = \begin{bmatrix} \cos(\theta_{r,i}) & -\sin(\theta_{r,i}) \\ \sin(\theta_{r,i}) & \cos(\theta_{r,i}) \end{bmatrix}. \tag{4}$$

Method



Lorentz Rotation Embedding

$$f : \mathbf{z} = (x_1, x_2, \cdots, x_d) \rightarrow \mathbf{x_0} = (x_0, x_1, x_2, \cdots, x_d), \qquad (5)$$

$$x_o = \sqrt{\gamma + \sum_{i=1}^{d-1} x_i^2} = \sqrt{\gamma + ||\mathbf{z}||^2}$$

$(u \rightarrow v)$

$$\mathbf{x}_u^{socs} = \mathbf{Rot}^s_{soc}(\mathbf{x}_u), \quad \mathbf{x}_v^{soct} = \mathbf{Rot}^t_{soc}(\mathbf{x}_v). \qquad (6)$$

$$\mathbf{x}_u^{difs} = \mathbf{Rot}^s_{dif}(\mathbf{x}_u), \quad \mathbf{x}_v^{dift} = \mathbf{Rot}^t_{dif}(\mathbf{x}_v). \qquad (7)$$

$$S_{uv}^{soc} = -D_{SL}(\mathbf{x}_u^{socs}, \mathbf{x}_v^{soct}) + b_u^{soc} + b_v^{soc}, \quad S_{uv}^{dif} = -D_{SL}(\mathbf{x}_u^{difs}, \mathbf{x}_v^{dift}) + b_u^{dif} + b_v^{dif}, \qquad (8)$$

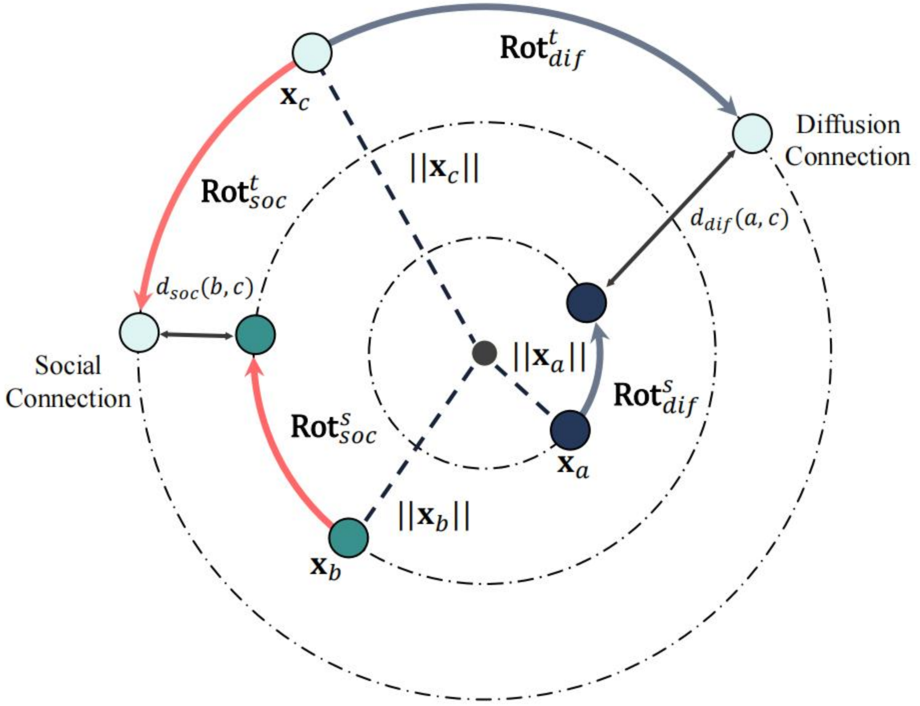
$$P(v|u) = \frac{e^{S_{uv}^{soc}}}{E(u)}, \qquad (9) \qquad E(u) = \sum_{i \in V_S} e^{S_{ui}^{soc}}$$

$$\log P(v|u) \approx \log \sigma(S_{uv}^{soc}) + \sum_{i \in N} \log \sigma(-S_{ui}^{soc}),$$
$$O_{soc} = \sum_{u \in V_S} \log P(C_u|u) = \sum_{u \in V_S} \sum_{v \in C_u} \log P(v|u), \qquad (10)$$

$$O_{dif} = \sum_{u \in V_D} \log P(D_u|u) = \sum_{u \in V_D} \sum_{v \in D_u} \log P(v|u), \qquad (11)$$

$$\mathcal{L}_{emb} = -\left(O_{soc} + O_{dif}\right). \qquad (12)$$

Method



Lorentz Rotation Embedding

$$f : \mathbf{z} = (x_1, x_2, \cdots, x_d) \rightarrow \mathbf{x_0} = (x_0, x_1, x_2, \cdots, x_d), \quad (5)$$

$$x_o = \sqrt{\gamma + \sum_{i=1}^{d-1} x_i^2} = \sqrt{\gamma + ||\mathbf{z}||^2}$$

$(u \rightarrow v)$

$$\mathbf{x}_u^{socs} = \mathbf{Rot}_{soc}^s(\mathbf{x}_u), \quad \mathbf{x}_v^{soct} = \mathbf{Rot}_{soc}^t(\mathbf{x}_v). \quad (6)$$

$$\mathbf{x}_u^{difs} = \mathbf{Rot}_{dif}^s(\mathbf{x}_u), \quad \mathbf{x}_v^{dift} = \mathbf{Rot}_{dif}^t(\mathbf{x}_v). \quad (7)$$

$$S_{uv}^{soc} = -D_{SL}(\mathbf{x}_u^{socs}, \mathbf{x}_v^{soct}) + b_u^{soc} + b_v^{soc}, \quad S_{uv}^{dif} = -D_{SL}(\mathbf{x}_u^{difs}, \mathbf{x}_v^{dift}) + b_u^{dif} + b_v^{dif}, \quad (8)$$

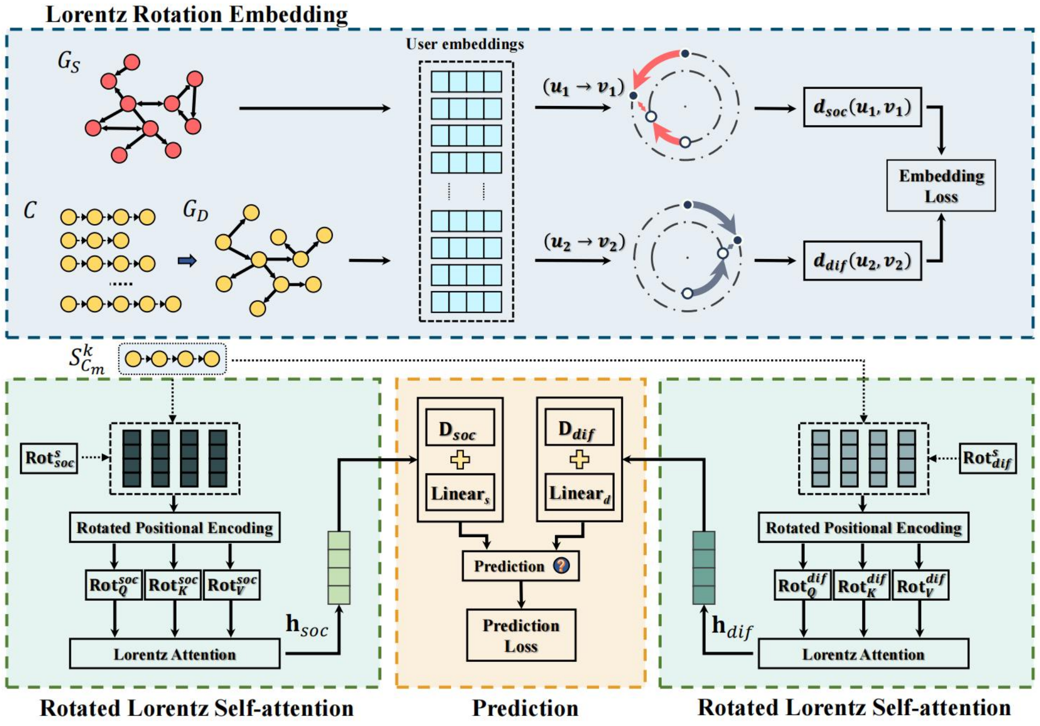
$$P(v|u) = \frac{e^{S_{uv}^{soc}}}{E(u)}, \quad (9) \qquad E(u) = \sum_{i \in V_S} e^{S_{ui}^{soc}}$$

$$\log P(v|u) \approx \log \sigma(S_{uv}^{soc}) + \sum_{i \in N} \log \sigma(-S_{ui}^{soc}), \quad O_{soc} = \sum_{u \in V_S} \log P(C_u|u) = \sum_{u \in V_S} \sum_{v \in C_u} \log P(v|u), \quad (10)$$

$$O_{dif} = \sum_{u \in V_D} \log P(D_u|u) = \sum_{u \in V_D} \sum_{v \in D_u} \log P(v|u), \quad (11)$$

$$\mathcal{L}_{emb} = -\left(O_{soc} + O_{dif}\right). \quad (12)$$

Method



Lorentz Rotation Embedding

$$f : \mathbf{z} = (x_1, x_2, \cdots, x_d) \rightarrow \mathbf{x_0} = (x_0, x_1, x_2, \cdots, x_d), \quad (5)$$

$$x_o = \sqrt{\gamma + \sum_{i=1}^{d-1} x_i^2} = \sqrt{\gamma + ||\mathbf{z}||^2}$$

$(u \rightarrow v)$

$$\mathbf{x}_u^{socs} = \mathbf{Rot}^s_{soc}(\mathbf{x}_u), \quad \mathbf{x}_v^{soct} = \mathbf{Rot}^t_{soc}(\mathbf{x}_v). \quad (6)$$

$$\mathbf{x}_u^{difs} = \mathbf{Rot}^s_{dif}(\mathbf{x}_u), \quad \mathbf{x}_v^{dift} = \mathbf{Rot}^t_{dif}(\mathbf{x}_v). \quad (7)$$

$$S_{uv}^{soc} = -D_{SL}(\mathbf{x}_u^{socs}, \mathbf{x}_v^{soct}) + b_u^{soc} + b_v^{soc}, \quad S_{uv}^{dif} = -D_{SL}(\mathbf{x}_u^{difs}, \mathbf{x}_v^{dift}) + b_u^{dif} + b_v^{dif}, \quad (8)$$

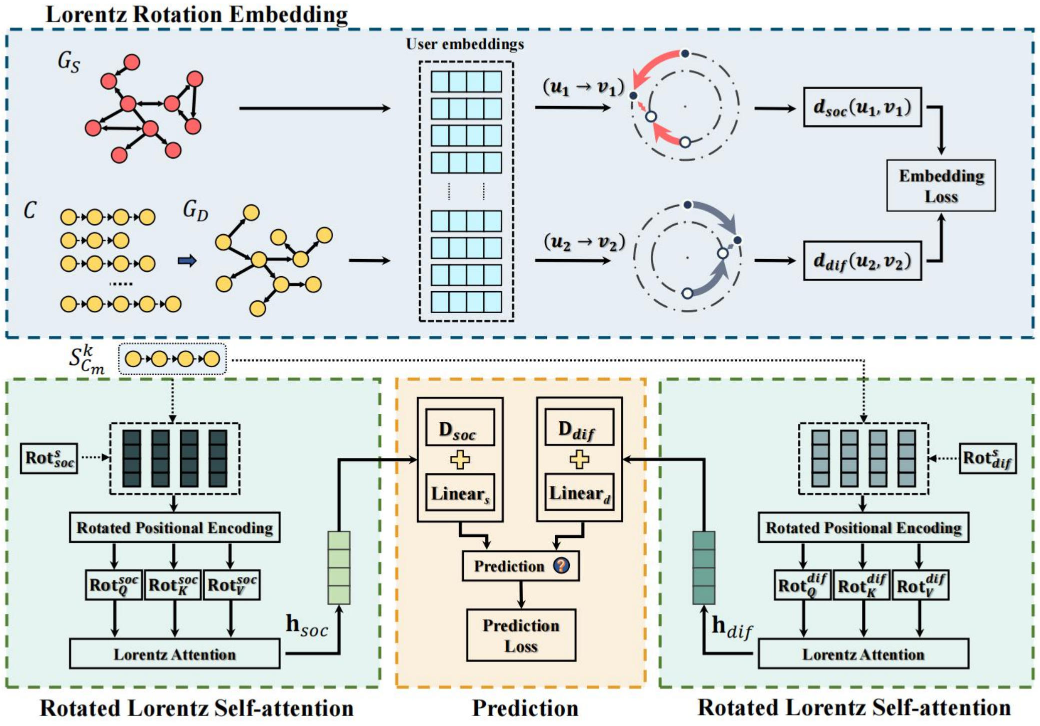
$$P(v|u) = \frac{e^{S_{uv}^{soc}}}{E(u)}, \quad (9) \qquad E(u) = \sum_{i \in V_S} e^{S_{ui}^{soc}}$$

$$\log P(v|u) \approx \log \sigma(S_{uv}^{soc}) + \sum_{i \in N} \log \sigma(-S_{ui}^{soc}),$$
$$O_{soc} = \sum_{u \in V_S} \log P(C_u|u) = \sum_{u \in V_S} \sum_{v \in C_u} \log P(v|u), \quad (10)$$

$$O_{dif} = \sum_{u \in V_D} \log P(D_u|u) = \sum_{u \in V_D} \sum_{v \in D_u} \log P(v|u), \quad (11)$$

$$\mathcal{L}_{emb} = -\left(O_{soc} + O_{dif}\right). \quad (12)$$

Method



$$\mathbf{y}_i = \frac{\sum_{j=1}^{|K|} \alpha_{ij} \mathbf{v}_j}{\sqrt{\gamma}\left\|\left\|\sum_{k=1}^{|K|} \alpha_{ik} \mathbf{v}_k\right\|_{\mathcal{L}}\right\|}, \quad (13)$$

$$\alpha_{ij} = \frac{\exp\left(\frac{-D_{SL}(\mathbf{q}_i, \mathbf{k}_j)}{\sqrt{d}}\right)}{\sum_{k=1}^{|K|} \exp\left(\frac{-D_{SL}(\mathbf{q}_i, \mathbf{k}_j)}{\sqrt{d}}\right)}. \quad (14)$$

$$\mathbf{h}_{soc} = \frac{\sum_{i=1}^{k} \alpha_i \mathbf{z}^{soc}_{V,i}}{\sqrt{\gamma}\left\|\left\|\sum_{i=1}^{k} \alpha_i \mathbf{z}^{soc}_{V,i}\right\|_{\mathcal{L}}\right\|},$$

$$\alpha_i = \frac{\exp\left(\frac{-D_{SL}(\mathbf{z}^{soc}_{Q,i}, \mathbf{z}^{soc}_{K,i})}{\sqrt{d}}\right)}{\sum_{j=1}^{k} \exp\left(\frac{-D_{SL}(\mathbf{z}^{soc}_{Q,i}, \mathbf{z}^{soc}_{K,j})}{\sqrt{d}}\right)}, \quad (15)$$

$$\mathbf{z}^{soc}_{Q,i} = \mathrm{Rot}_Q(\mathbf{x}^{soc_s}_{p,i}), \mathbf{z}^{soc}_{K,i} = \mathrm{Rot}_K(\mathbf{x}^{soc_s}_{p,i}) \text{ and } \mathbf{z}^{soc}_{V,i} = \mathrm{Rot}_V(\mathbf{x}^{soc_s}_{p,i})$$

$$\mathbf{x}^{soc_s}_{p,i} = \mathrm{Rot}^{pe}_i(\mathbf{x}^{soc_s}_i)$$

$$\mathbf{y}_{soc} = \mathbf{D}_{soc} + \mathbf{h}^T_{soc}\mathbf{W}_s + \mathbf{b}_s,$$
$$\mathbf{y}_{dif} = \mathbf{D}_{dif} + \mathbf{h}^T_{dif}\mathbf{W}_d + \mathbf{b}_d, \quad (16)$$

$$\mathbf{D}_{soc,i} = -D_{SL}(\mathbf{h}_{soc}, \mathbf{x}^{soc_t}_i) \qquad \mathbf{D}_{dif,i} = -D_{SL}(\mathbf{h}_{dif}, \mathbf{x}^{dif_t}_i)$$

$$\hat{\mathbf{y}} = \mathrm{Softmax}\left(\mathbf{y}_{soc} + \mathbf{y}_{dif} + \mathbf{M}^{pre}\right), \quad (17)$$

$$\mathcal{L}_{pre} = -\sum_{j=1}^{|S|}\sum_{i=1}^{|U_S|} \mathbf{y}_{j,i}\log(\hat{\mathbf{y}}_{j,i}), \quad (18)$$

$$\mathcal{L}_{total} = \mathcal{L}_{emb} + \mathcal{L}_{pre}. \quad (19)$$

Experiments

Table 1: Statistics of datasets used in our experiments.

Dataset	#Nodes	#Edges	#Cascades	#Ave Length
Android	9,958	48,573	679	41.05
Christianity	2,897	35,624	587	25.10
Memetracker	4,709	-	12,661	16.24
Twitter	12,627	309,631	3,442	32.60
Douban	12,232	396,580	3,475	21.76

Experiments

Table 2: The prediction results of Hits@k on five datasets.

Dataset	Android			Christianity			Memetraker			Twitter			Douban		
Hits@k	@10	@50	@100	@10	@50	@100	@10	@50	@100	@10	@50	@100	@10	@50	@100
NDM	0.0339	0.0953	0.1572	0.1651	0.3510	0.4553	0.2083	0.3663	0.4583	0.1934	0.2941	0.3573	0.1013	0.2123	0.3125
Inf-VAE	0.0673	0.1573	0.2179	0.1774	0.3960	0.5215	0.2124	0.4077	0.4934	0.1476	0.3178	0.4512	0.1116	0.2214	0.3468
FOREST	0.0700	0.1514	0.2237	0.2632	0.4909	0.6056	0.2963	0.4780	0.5786	0.2552	0.3850	0.4607	0.1868	0.3084	0.3857
DyHGCN	0.0842	0.1915	0.2679	0.2594	0.4976	0.6047	0.2952	0.4864	0.5848	0.2901	0.4688	0.5719	0.1987	0.3289	0.3942
HyperINF	0.0848	0.1553	0.2236	0.2700	0.4460	0.5165	0.2483	0.4634	0.5949	0.2692	0.4442	0.5648	0.1834	0.3321	0.4016
MS-HGAT	0.1049	0.1987	0.2747	0.2781	0.4814	0.5703	0.2843	0.4966	0.6047	0.2996	0.4654	0.5735	0.2065	0.3504	0.4136
H-diffu	0.0981	0.1860	0.2623	0.2746	0.5089	0.6004	0.2195	0.4499	0.5720	0.2707	0.4533	0.5636	0.1984	0.3479	0.4155
RotDiff	**0.1144**	**0.2304**	**0.3130**	**0.3237**	**0.5625**	**0.6674**	**0.3066**	**0.5170**	**0.6206**	**0.3590**	**0.5246**	**0.6121**	**0.2216**	**0.3823**	**0.4637**

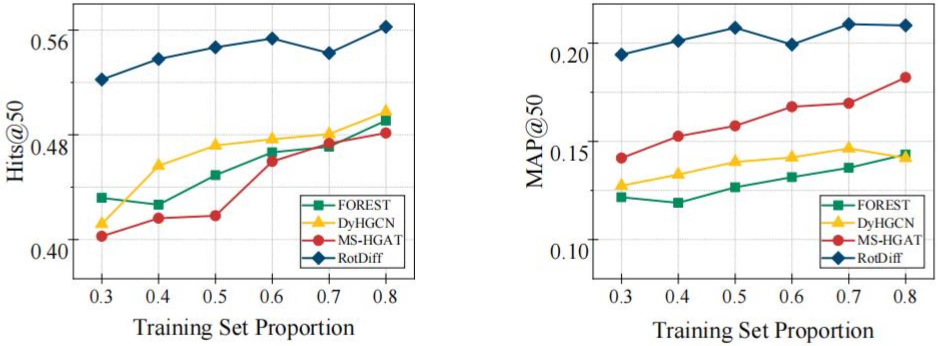
Table 3: The prediction results of MAP@k on five datasets.

Dataset	Android			Christianity			Memetraker			Twitter			Douban		
MAP@k	@10	@50	@100	@10	@50	@100	@10	@50	@100	@10	@50	@100	@10	@50	@100
NDM	0.0219	0.0244	0.0252	0.0676	0.0751	0.0765	0.1059	0.1131	0.1144	0.1296	0.1339	0.1348	0.0836	0.0879	0.0936
Inf-VAE	0.0426	0.0441	0.0482	0.1035	0.1194	0.1249	0.1345	0.1379	0.1446	0.1632	0.1725	0.1747	0.1044	0.1098	0.1142
FOREST	0.0381	0.0416	0.0426	0.1328	0.1433	0.1449	0.1553	0.1637	0.1751	0.1733	0.1790	0.1801	0.1086	0.1146	0.1183
DyHGCN	0.0458	0.0503	0.0514	0.1303	0.1415	0.1432	0.1611	0.1623	0.1725	0.1751	0.1832	0.1847	0.1048	0.1114	0.1148
HyperINF	0.0424	0.0461	0.0467	0.1629	0.1719	0.1732	0.1434	0.1545	0.1566	0.1679	0.1756	0.1774	0.1042	0.1139	0.1138
MS-HGAT	0.0633	0.0675	0.0685	0.1732	0.1825	0.1836	0.1542	0.1641	0.1657	0.1880	0.1951	0.1965	0.1122	0.1187	0.1198
H-diffu	0.0606	0.0643	0.0653	0.1689	0.1795	0.1808	0.1409	0.1514	0.1531	0.1777	0.1868	0.1885	0.1067	0.1017	0.1127
RotDiff	**0.0696**	**0.0745**	**0.0756**	**0.1981**	**0.2091**	**0.2105**	**0.1653**	**0.1691**	**0.1766**	**0.2406**	**0.2482**	**0.2495**	**0.1170**	**0.1254**	**0.1266**

Experiments

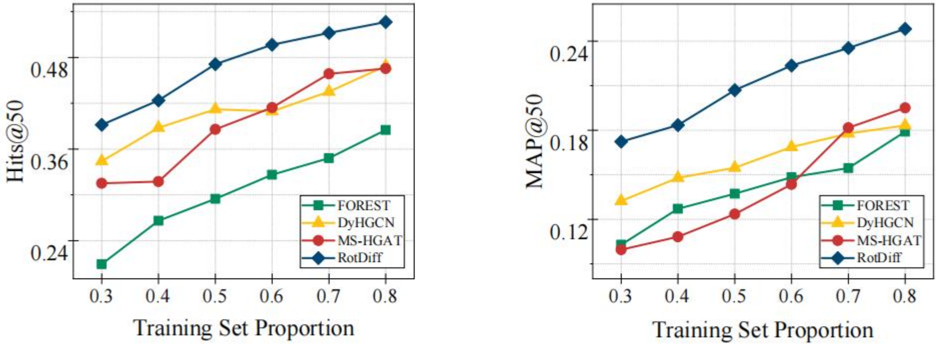
Table 4: The effect of curvature parameter γ.

Dataset	Android		Christianity	
Metrics (@50)	Hits	MAP	Hits	MAP
$\gamma = 0.1$	**0.2312**	0.0728	0.5402	0.2087
$\gamma = 0.3$	0.2296	0.0709	0.5513	0.2053
$\gamma = 0.6$	0.2258	0.0724	**0.5670**	0.2054
$\gamma = 1.0$	0.2304	**0.0745**	0.5625	**0.2091**
$\gamma = 1.5$	0.2289	0.0729	0.5603	0.2041
$\gamma = 2.0$	0.2120	0.0724	0.5491	0.2053



(a) Hits@50 (b) MAP@50

Figure 5: The effect of training set proportion on Christianity.



(a) Hits@50 (b) MAP@50

Figure 6: The effect of training set proportion on Twitter.

Experiments

Table 5: The results of ablation study.

Dataset	Christianity		Twitter	
Metrics (@50)	Hits	MAP	Hits	MAP
(1) w/o hyperbolic	0.5312	0.1928	0.5102	0.2287
(2) only soc-graph	0.5446	0.1993	0.5087	0.2300
(3) only dif-graph	0.5580	0.1974	0.5210	0.2319
(4) w/o Lo-rot-emb	0.5369	0.1955	0.4867	0.2137
(5) w/o all-att	0.5133	0.1845	0.4783	0.2032
(6) w/o Rot-in-att	0.5446	0.1924	0.5186	0.2295
RotDiff	**0.5625**	**0.2091**	**0.5246**	**0.2482**

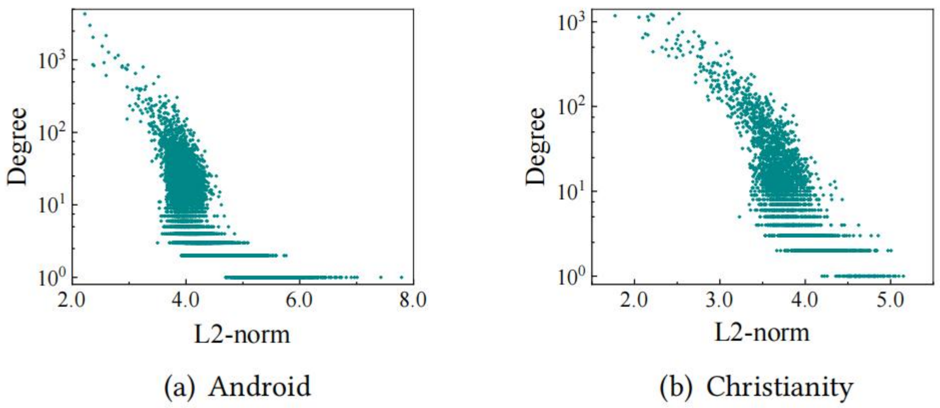


Figure 7: The correlation between the L2-norm of user embeddings and node degree.

Thank you!