Decoupling User Relationships Guides Information Diffusion Prediction (Student Abstract)

Wenxue Ye[1], Shichong Li[1], Zhangtao Cheng[1,2], Xovee Xu[1,2], Ting Zhong[1,2], Bei Hui[*] [1], Fan Zhou[1,2]

[1] University of Electronic Science and Technology of China, Chengdu, Sichuan 610054, China
[2] Kashi Institute of Electronics and Information Industry, Kashgar 844000, China
{wenxue.ye, lischong, zhangtao.cheng}@outlook.com, xovee@live.com, {zhongting, bhui, fan.zhou}@uestc.edu.cn

Abstract

Information diffusion prediction is a critical task for many social network applications. However, current methods are mainly limited by the following aspects: user relationships behind resharing behaviors are complex and entangled. To address these issues, we propose MHGFomer, a novel multi-channel hypergraph transformer framework, to better decouple complex user relations and obtain fine-grained user representations. First, we employ designed triangular motifs to decouple user relations into three different level hypergraphs. Second, a position-aware hypergraph transformer is used to refine user relation and obtain high-quality user representations. Extensive experiments conducted on two social datasets demonstrate that MHGFomer outperforms state-of-the-art diffusion models across several settings.

Introduction

Online user generated content (UGC) rapidly propagates to numerous users through posting and resharing behaviors. This process can be described as a *cascade* of user adoptions of the post. *Information diffusion prediction* (IDP) aims to identify the future potential users who will adopt the post, given previously adopted users. Existing works (Yuan et al. 2020; Sun et al. 2022) primarily utilize deep learning architectures (e.g., sequential models and graph models) to design an automatic framework to capture the structural and temporal cascade representations for IDP.

Challenges. Despite their successes, existing methods are still limited due to the following aspects: **User relationships behind re-sharing behaviors are complex and entangled.** They generally employ graph neural networks to only exploit the simple pairwise user relations from the social network. However, in real scenarios, user relationships often consist of ubiquitous high-order user relations. For instance, two users who have social connections and similar sharing preferences exist a stronger relations than those who are only socially connected, whereas common resharing information in the former is often ignored in previous diffusion models.

Present work. To solve the above issues, we propose *MHGFomer*, a novel **M**ulti-channel **H**yper**G**raph trans**Former** framework, to better decouple complex user relationships and obtain fine-grained user representations for IDP. First, in order to decouple complex user relations, we design particular triangular motifs with underlying semantics and construct three different levels of hypergraphs with specific triangular relations. Second, we design a position-aware hypergraph transformer layer to update and learn fine-grained user representations from multi-channel hypergraphs.

Methodology

Hypergraph Construction. To formulate the complex user relations, we first align the social network and distinct user cascade sequences, and then construct different level hypergraphs over this heterogeneous network. Inspired by Motif (Milo et al. 2002), we design three types of triangular motifs in our IDP task, i.e., $\mathcal{M}_1$-$\mathcal{M}_7$ describing user social relationships like "having a mutual follower", $\mathcal{M}_8$-$\mathcal{M}_9$ representing the compound relation like "followers retweeting the same UGC", and $\mathcal{M}_{10}$ denoting a additional compound relation like "users retweeting the same UGC without social connections". Therefore, we construct three types of hypergraphs via designed above triangular motifs, involving society-aware hypergraph ($\mathcal{G}_{\mathcal{S}}$), interest-aware hypergraph ($\mathcal{G}_{\mathcal{I}}$), and joint-aware hypergraph ($\mathcal{G}_{\mathcal{J}}$).

User Social Dependency Learning. To learn disentangled relation-specific user representations, we design a *relation-aware self-gating* (RSG) operation to initialize user embeddings. The RSG is defined as: $\boldsymbol{X}_s = f^s_{gate}(\boldsymbol{X}) = \boldsymbol{X} \odot \text{sigmoid}(\boldsymbol{X}\boldsymbol{W}_s + \boldsymbol{b}_s)$, where $\boldsymbol{W}_s \in \mathbb{R}^{d\times d}$, $\boldsymbol{b}_s \in \mathbb{R}^d$ are learnable parameters, $s \in \{\mathcal{S}, \mathcal{I}, \mathcal{J}\}$ denotes different user relations, $\odot$ represents the element-wise product, $\boldsymbol{X} \in \mathbb{R}^{N\times d}$ denotes the base user embedding matrix encoding users, and d is the adjustable latent dimensions.

To jointly preserve the sequential information and model different kinds of user relationships, we design a position-aware hypergraph transformer. First, we adopt a position encoding function $\mathbf{PE}(\cdot)$ to encode the user index in its corresponding cascade p_i. The position encoder is defined as: $\mathbf{PE}(pos, 2i) = sin(pos/10000^{2i/d})$, $\mathbf{PE}(pos, 2i+1) = cos(pos/10000^{2i/d})$, where d denotes the dimension of the position embedding. Subsequently, we obtain the initial user representations $\boldsymbol{z}^s_i = \boldsymbol{x}^s_i + \mathbf{PE}(u_i)$. Specifically, the hypergraph transformer consists of a two-step aggregation process, i.e., *node-to-hyperedge* and *hyperedge-to-node*,

[*]Corresponding author

Copyright © 2024, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

Model	Twitter		Weibo	
	H@100	M@100	H@100	M@100
SNIDSA	43.59	18.89	39.51	5.52
FOREST	50.12	22.36	52.55	8.63
Inf-VAE	46.42	21.82	38.05	6.58
DyHGCN	58.53	22.45	51.65	8.27
MS-HGAT	56.68	17.37	40.05	7.38
MHGFomer	**58.99**	**22.98**	**55.50**	**8.93**

Table 1: Performance comparisons on two datasets.

for refining user representations. In the process of node-to-hyperedge, given representations $\boldsymbol{Z}^s$ of hypergraph $\mathcal{G}_s$, an l-th hypergraph transformer layer calculates the j-th hyperedge representation as follows:

$$\boldsymbol{e}_j^{s,l} = \sum\nolimits_{n_k \in e_j^s} \left(\alpha_{jk} \mathbf{W}_z^s \boldsymbol{z}_k^{s,l-1}\right), \tag{1}$$

where $\mathbf{W}_z^s$ denotes learnable parameters and α_{jk} denotes the attention coefficient of node n_k in the hyperedge e_j^s. The attention scores are calculated as follows:

$$\alpha_{jk} = \frac{\exp\left(\sigma\left(\vec{\mathbf{a}}_{az}^T\left[\boldsymbol{z}_k^{s,l-1} \odot \bar{\boldsymbol{e}}_j^{s,l-1}\right]\right)\right)}{\sum_{v_\zeta \in e_j} \exp\left(\sigma\left(\vec{\mathbf{a}}_{az}^T\left[\boldsymbol{z}_\zeta^{s,l-1} \odot \bar{\boldsymbol{e}}_j^{s,l-1}\right]\right)\right)}, \tag{2}$$

where $\vec{\mathbf{a}}_{az} \in \mathbb{R}^d$ is a weight vector, $\bar{\boldsymbol{e}}_j^{s,l-1} = \{\boldsymbol{z}_\zeta^{s,l-1} | v_\zeta \in e_j^s\}$ is the average of the cluster, $\odot$ is the Hadamard product. σ is the LeakyReLU activation function. For the hyperedge-to-node, given learned hyperedge features $\boldsymbol{E}^{s,l}$, the second step is to update representations $\boldsymbol{Z}^{s,l-1}$:

$$\boldsymbol{z}_k^{s,l} = \sum\nolimits_{e_j \in n_k^s} \left(\beta_{kj} \mathbf{W}_e^s \boldsymbol{e}_j^{s,l}\right), \tag{3}$$

$$\beta_{kj} = \frac{\exp\left(\sigma\left(\vec{\mathbf{a}}_{ae}^T\left[\boldsymbol{z}_j^{s,l-1} \odot \boldsymbol{e}_k^{s,l}\right]\right)\right)}{\sum_{e_\zeta \in n_k^s} \exp\left(\sigma\left(\vec{\mathbf{a}}_{ae}^T\left[\boldsymbol{z}_k^{s,l-1} \odot \boldsymbol{e}_\zeta^{s,l}\right]\right)\right)}, \tag{4}$$

where $\boldsymbol{z}_k^{s,l}$ denotes updated features of node n_k^s and $\vec{\mathbf{a}}_{ae} \in \mathbb{R}^d$ is a weight vector. $\mathbf{W}_e^s$ is trainable parameters and β_{kj} represents the attention score of hyperedge e_j^s that connects to node n_k^s. Then, we combine the user features of L layers to avoid the over-smoothing: $\bar{\mathbf{X}}_s = \frac{1}{L+1}\sum_{l=0}^{L} \mathbf{X}_s^l$. Finally, the comprehensive relation-specific user representations can be computed as $\boldsymbol{P} = \sum_{s \in \{\mathcal{S},\mathcal{I},\mathcal{J}\}} \bar{\mathbf{X}}_s$.

Diffusion Prediction. For the final prediction, the probabilities $\hat{\mathbf{y}}_k \in \mathbb{R}^{N \times 1}$ for all users are calculated by: $\hat{\mathbf{y}}_k = \text{softmax}(\boldsymbol{Z}^* \boldsymbol{P}^\top + \mathbf{M}_{\text{mask}})$, where $\mathbf{M}_{\text{mask}}$ is used to mask users who have already been activated. We adopt the cross entropy loss $\mathcal{L}_{\text{cross}}$ as the objective to optimize the model.

Experiments

Datasets and Baselines. We conduct experiments on Weibo and Twitter datasets, and compare our **MHGFomer** with five baselines: **SNIDSA** (Wang, Chen, and Li 2018), **FOREST** (Yang et al. 2019), **Inf-VAE** (Sankar et al. 2020), **DyHGCN** (Yuan et al. 2020), **MS-HGAT** (Sun et al. 2022). We

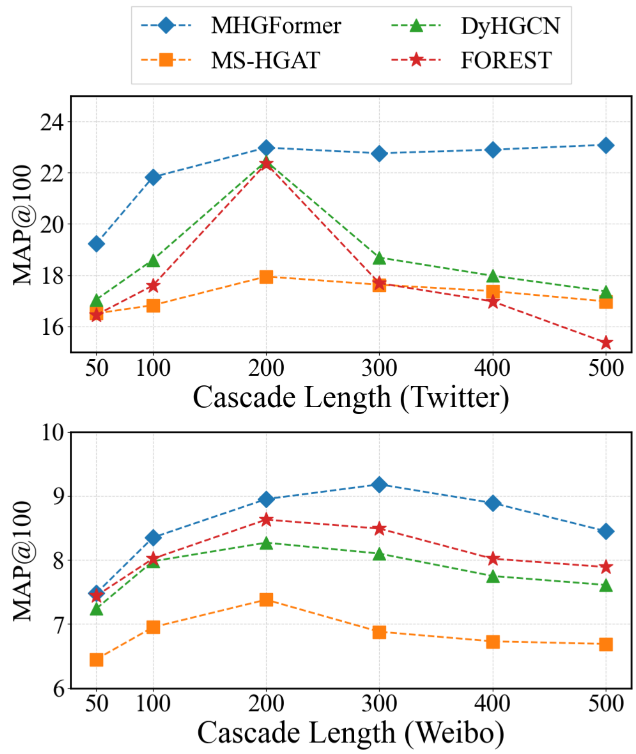

Figure 1: Performance with different cascade lengths.

employ two ranking metrics: MAP@K (M@K) and Hits@K (H@K), K = 100.

Main results. Table 1 reports overall comparison results. We can observe that our MHGFomer outperforms all baselines in terms of both ranking metrics. Specifically, on the Twitter dataset, MHGFomer outperforms the best baseline by 0.78% and 2.36% in terms of H@100 and M@100, respectively. This result verifies the effectiveness of decoupling complex user relations and obtaining fine-grained user representations.

Decoupling Analysis. We conducted an experiment on Twitter and Weibo datasets to investigate the impact of decoupling user relationships on the cascade length. Specifically, we set a range of cascade lengths from 50 to 500 for verification. Fig. 1 shows the result in terms of MAP@100, different from the three baselines (FOREST, DyHGCN, and MS-HGAT), the performance of our proposed model MHGFomer decreases relatively slowly as the cascade length increases.

Acknowledgements

This work was supported in part by National Natural Science Foundation of China (Grant No.62176043, No.62072077, No. 62273071) and Kashgar Science and Technology Bureau (Grant No.KS2023025).

References

Milo, R.; Shen-Orr, S.; Itzkovitz, S.; Kashtan, N.; Chklovskii, D.; and Alon, U. 2002. Network motifs: simple building blocks of complex networks. *Science*, 298(5594): 824–827.

Sankar, A.; Zhang, X.; Krishnan, A.; and Han, J. 2020. Inf-vae: A variational autoencoder framework to integrate homophily and influence in diffusion prediction. In *WSDM*, 510–518.

Sun, L.; Rao, Y.; Zhang, X.; Lan, Y.; and Yu, S. 2022. MS-HGAT: Memory-enhanced Sequential Hypergraph Attention Network for Information Diffusion Prediction. In *AAAI*.

Wang, Z.; Chen, C.; and Li, W. 2018. A sequential neural information diffusion model with structure attention. In *CIKM*, 1795–1798.

Yang, C.; Tang, J.; Sun, M.; Cui, G.; and Liu, Z. 2019. Multi-scale Information Diffusion Prediction with Reinforced Recurrent Networks. In *IJCAI*, 4033–4039.

Yuan, C.; Li, J.; Zhou, W.; Lu, Y.; Zhang, X.; and Hu, S. 2020. DyHGCN: A dynamic heterogeneous graph convolutional network to learn users' dynamic preferences for information diffusion prediction. In *ECML-PKDD*, 347–363.